MANUAL

NEUSREL 2.4

Universal Structure Modeling Software

Matlab Based Analysis Software

August 1, 2009, Cologne Germany

Dr. Frank Buckler, NEUSREL Causal Analytics, Jungbergerstr. 7, 51105 Cologne, Germany
+49 163 282 55 37, Buckler@neusrel.com , www.neusrel.com

Overview

Getting Started

Matlab Software

- NEUSREL requires a base licence of Matlab R12 or later
- Please contact MathWorks for purchase options: http://www.mathworks.com

System Requirements

Operating System:

- Windows® XP (Service Pack 1, 2 or 3)
- Windows Server 2003 (Service Pack 1 or 2, R2)
- Windows Vista™ (Service Pack 1)

Processors:

- Intel® Pentium (Pentium 4 and above)
- Intel Celeron
- Intel Xeon
- Intel Core
- AMD Athlon 64**
- AMD Opteron
- AMD Sempron

Disk Space: 510 MB* (MATLAB® only)

RAM: 512 MB (1024 MB recommended)

Installation of NEUSREL

- Extract NEUSREL.zip anywhere onto your harddrive. For instance to: “C:\NEUSREL\”
- Set an additional Matlab path (Matlab menu: File > Set Path > Add Folder > …)

Run NEUSREL

- Start a project by filling the Excel Template >> See next chapter
- Import project data into Matlab: >> See next chapter
- Run analysis by typing NEUSREL in the Matlab command window and push “Return”

Start a Project

1. You fill the Excel-template file with data, options and project settings

<u>**Step1:** Fill sheet “Project”</u>

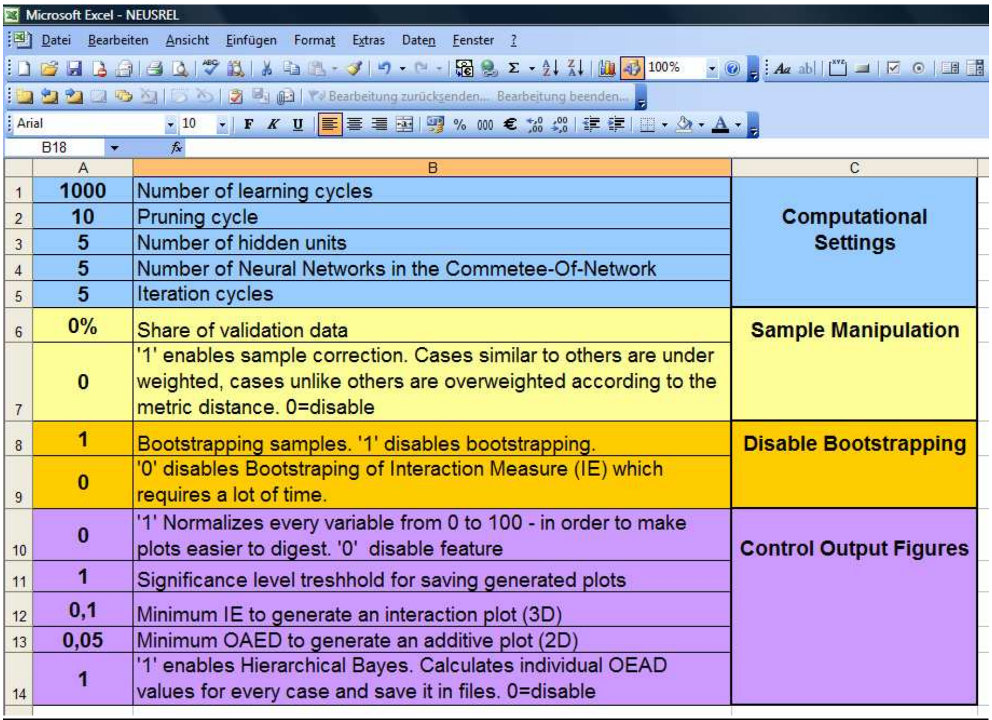
	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	6	CSI		1=Invert Scale, 0=disable	Missings	Reflexiv=1 Formative=2	Quality	Expectati on	Value	Satisfacti on	Loyalty	Complain ts	###	0
2	Quality	1	7	1	99	1	0	0	0	0.5	0.5	0.5	0.5	
3	Expectation	8	8	1	99	1	0.5	0	0	0.5	0.5	0.5	0.5	
4	Value	9	10	1	99	1	0	0	0	0.5	0.5	0.5	0	
5	Satisfaction	11	13	1	99	1	0	0	0	0	0.5	0.5	0	
6	Loyalty	14	15	1	99	1	0	0	0	0	0	0	0	
7	Complaints	16	16	1	99	1	0	0	0	0	0.5	0	0	

Cell	Cell Colour	Content
A1	Orange	Enter number of latent variables. The “6” here is an example.
B1	Olive-colored	Enter project name. “CSI” is an example for a name. Do not use blanks at all!
A2-A...	Yellow	Name of latent variables (LV) . The order should match the order within data matrix (in Excel-sheet “data”). The names in the picture are examples.
B2-B…	Green	Number of column within data matrix (in Excel-sheet “data”) where the <u>**first**</u> manifest variable (MV) that belongs to the LV name, mentioned in the same row.
C2-C…	Green	Number of column within data matrix (in Excel-sheet “data”) where the <u>**last**</u> manifest variable (MV) that belongs to the LV name, mentioned in the same row. LVs are computed out of MVs, that are found in the columns from first to last MV. That means **corresponding MVs must be in this order within the data matrix** (in Excel-sheet “data”)**.** In the example screen shot you see for Quality that MVs in columns 1 to 7 is defined. That means in sheet “data” all MVs in columns A to G belong to Quality.
D2-D…	Light-Green	Placing a “1” in **inverts the scale** of the manifest variables (MV) behind this LV. You might wish to do this in order to improve readability of plots. If e.g. satisfaction gets higher, you might which to have higher values. In some scales in fact 1 means high

		and e.g. 5 low.
E2-E…	Turquoise	Enter here the numerical code you used in your data matrix in order to mark **missing values** (no answers). If you have no missing values anymore just enter a number which is not used within the data matrix. NEUSREL uses the **20-Nearest-Nabour approach** to substitute missing values. This is superior to a simple mean imputation.
F2-F…	Grey	If your measuring model is reflexive use "1", if formative use 2. "Reflexive" means that MVs are effected by the latent variable (example: customer satisfaction). "Formative" means that latent variable is constituted and caused by MVs (example: index of reputation elements) Please consult scientific literature if you are not sure which to choose.
Matrix starting in G2	Pink	The number of rows and columns in the pink matrix equals the number of LVs (defined cell A1) In this pink matrix you define which cause-effect relation you like to allow and which not. <u>Causing LVs are found in rows</u> (labelled in yellow cells) <u>Effected/influenced LVs are found in columns</u>. A "0" in cell I2 means that "Quality" is not influencing "Expectation" and will therefore be excluded. A "0,5" in cell J3 means that "expectation" will be considered when analysing the causes of "satisfaction". "0,5" is the preferred default value which indicates that you give the relation a priori a 50% chance of being true. If you have strong indication to set the value anywhere else below 1 (100%), feel free to do so. Please refer to literature of Bayessian statistics to find ways to systematically measure this apriori likelihood, e.g. betting games are method to retrieve apriori likelihood figures. Note: Within NEUSREL you are not restricted to stick only with theoretical justified pathes. You are only required to exclude pathes that are from a theoretical standpoint *not at all* justifiable. Usually you will at least exclude some pathes in order to **define the direction of effect**. When there are no u-shape nonlinearities, it is not possible to draw from (non-longitudinal) data causal conclusion.

NOTE: Leave all not used cells in the sheets clear. E.g. Don't leave a "Name LV15" when you do not have a 15th LV. Accordingly leave the matrix starting at I2 only in the size it need to be as described. Otherwise NEUSREL might not work.

<u>Step2:</u> <u>Fill sheet "Options"</u>



	A	B	C
1	1000	Number of learning cycles	Computational Settings
2	10	Pruning cycle	
3	5	Number of hidden units	
4	5	Number of Neural Networks in the Commetee-Of-Network	
5	5	Iteration cycles	
6	0%	Share of validation data	Sample Manipulation
7	0	'1' enables sample correction. Cases similar to others are under weighted, cases unlike others are overweighted according to the metric distance. 0=disable	
8	1	Bootstrapping samples. '1' disables bootstrapping.	Disable Bootstrapping
9	0	'0' disables Bootstraping of Interaction Measure (IE) which requires a lot of time.	
10	0	'1' Normalizes every variable from 0 to 100 - in order to make plots easier to digest. '0' disable feature	Control Output Figures
11	1	Significance level treshhold for saving generated plots	
12	0,1	Minimum IE to generate an interaction plot (3D)	
13	0,05	Minimum OAED to generate an additive plot (2D)	
14	1	'1' enables Hierarchical Bayes. Calculates individual OEAD values for every case and save it in files. 0=disable	

- For most studies we conducted so far, it was not necessary to change the defaults A1-A6 in order to improve results significantly.
- You might consider changing the values in order to reduce calculation time.

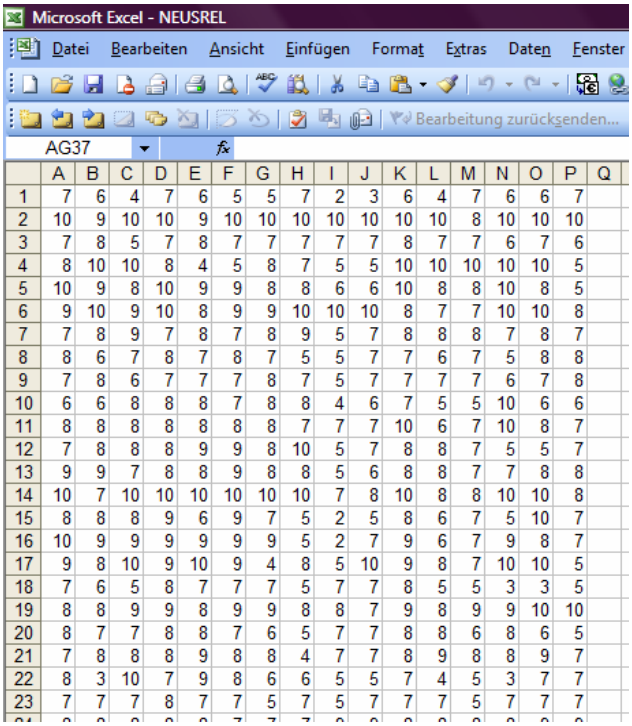
Cell	Preset value	Description
A1	1000	**Number of learning cycles** of neural networks Tip: Decrease default only when you like to save computation time. But this will likely severely effect the quality of estimation.
A2	10	**Pruning Frequency**: Frequency of learning cycles at which hidden units will be subject for pruning. Tip: Increasing it for saving time. But this will likely severely effect the quality of estimation.

A3	10	**Number of hidden units**. This value defines the maximum complexity of nonlinearities and interactions NEUSREL will be able to capture. The method automatically kills hidden units when not needed. Tip: Decrease it when you are sure that little nonlinearities and interactions exists or when you need to save computation time. Increase only if very complex nonlinearities and interactions are expected.
A4	5	Number of neural networks simultaneously trained (**Committee Of Networks**) Tip: Decrease to save computation time. Increase to support generalization performance.
A5	3	**Number of iterative cycles** with altering estimation of structural and measurement model. Tip: Increase if you are like to increase accuracy and you have a lot of time. Decrease if you trust more in your measurement models then in your structural model. Even a value of 1 is justifiable if you are confident that your measuring models are true.
A6	0%	**Validation data: Portion** of the dataset which will be not subject for learning but for model validation. Tip: Increase to test generalization performance. But be cautious since this decreases the number of cases for model specification and therefore automatically reduces the quality of estimation. This is especially an option when operating with large data sets.
A7	0	All values different from 1 represent a standard analysis. A **value of 1** represents the **Sample Correction Modus**. Cases similar (i.e. close) to others will get under-weighted and cases unlike (i.e. far to) others will get over-weighted according to the average metric distance to all other cases. Use this modus if you are interested in the exact form of relations along the whole domain of values and if you want higher precision in domain regions where data are rather rare. This is on the expense of R2 on the sample at hand. Therefore, do NOT use this modus, if you like to derive relations that best fit the data generated by the existing sample method (e.g. the specific survey design)
A8	200	**Number of Bootstrapping-Samples** Tip: Start first computations with "1" since bootstrapping is very time consuming. 100 Bootstrapping-Samples require the time of 100 iterations

		(A7)
A9	0	**Disables Bootstraping of Interaction Measure (IE)** Bootstrapping on the IE measure requires a lot of time (often several days). Disable this with a Zero value in this cell.
A10	0	All values different from 1 represent a standard analysis. A **value of 1** represents the **Normalization Modus**. It normalizes every variable to a scale of 0 to 100. Within the plots the new scale can be interpreted as percentage of the variable at hand.
A11	1	**Log only significant graphs** All generated plots are saved as Bitmap-Files in the current path. With this value you can define that only plots for relations that have a minimum significance level are saved. With 1 every plot is saved. E.g. with 0.10 every relationship with significance level (likelihood of failure) from 0% to 10% is saved.
A12	0.05	**Generate interaction plot only above this minimum IE** Generate plots only for high impact interactions. Set the value to 0 in order to disable omission of plots. Or use rather high values in order **speed up your calculation** time. With many variables (LVs) this can reduce your waiting time by 5, 10 or even 20.
A13	0.05	**Generate additive plots only above this minimum OEAD** Generate plots only for high impact effects. Set the value to 0 in order to disable omission of plots.
A14	1	**Generate individual OEAD values per case.** '1' enables the **Hierarchical Bayes** feature. Values are saved in the current path as a wk1-file. This format can be read by Excel. For every endogen variable a separate file is generated.

NOTE: Leave all other cells clear.

Step 3: Fill sheet “Data”



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	7	6	4	7	6	5	5	7	2	3	6	4	7	6	6	7	
2	10	9	10	10	9	10	10	10	10	10	10	10	8	10	10	10	
3	7	8	5	7	8	7	7	7	7	7	8	7	7	6	7	6	
4	8	10	10	8	4	5	8	7	5	5	10	10	10	10	10	5	
5	10	9	8	10	9	9	8	8	6	6	10	8	8	10	8	5	
6	9	10	9	10	8	9	9	10	10	10	8	7	7	10	10	8	
7	7	8	9	7	8	7	8	9	5	7	8	8	8	7	8	7	
8	8	6	7	8	7	8	7	5	5	7	7	6	7	5	8	8	
9	7	8	6	7	7	7	8	7	5	7	7	7	7	6	7	8	
10	6	6	8	8	8	7	8	8	4	6	7	5	5	10	6	6	
11	8	8	8	8	8	8	8	7	7	7	10	6	7	10	8	7	
12	7	8	8	8	9	9	8	10	5	7	8	8	7	5	5	7	
13	9	9	7	8	8	9	8	8	5	6	8	8	7	7	8	8	
14	10	7	10	10	10	10	10	10	7	8	10	8	8	10	10	8	
15	8	8	8	9	6	9	7	5	2	5	8	6	7	5	10	7	
16	10	9	9	9	9	9	9	5	2	7	9	6	7	9	8	7	
17	9	8	10	9	10	9	4	8	5	10	9	8	7	10	10	5	
18	7	6	5	8	7	7	7	5	7	7	8	5	5	3	3	5	
19	8	8	9	9	8	9	9	8	8	7	9	8	9	9	10	10	
20	8	7	7	8	8	7	6	5	7	7	8	8	6	8	6	5	
21	7	8	8	8	9	8	8	4	7	7	8	9	8	8	9	7	
22	8	3	10	7	9	8	6	6	5	5	7	4	5	3	7	7	
23	7	7	7	8	7	7	5	7	5	7	7	7	5	7	7	7	

- Copy & paste your data (e.g. from SPSS) into the sheet “Data”.
- Make sure the data is still formatted as numbers and not as strings or text.
- Rows are the cases
- Columns are the manifest variables (MV).
- Keep in mind that the MVs that belong to a LV must be in neighbour columns.
- All variables have to be numerical.
- Do not put variable names in here.
- All missing values have to be coded with a dedicated value. No cells within the rectangular matrix should be blank.

2. Save project Excel file…

… as a xls-file. Use your project name (or any other description) as file name.

3. Import project Excel file

- Import project data into Matlab:
    - File > Import Data
    - Select project xls file
    - Input wizard opens: Press the “Finish” button
- When you are a user of a Matlab **student version**, the above does NOT result in correct imports. Instead you need the following procedure, which is optional for all other users:
    - Name your xls-file as follows: “NEUSREL.xls”. And place it somewhere in the defined path/directory
    - Enter in Matlab command window: xlsload and press Enter
- The latter option is more convenient, with the only disadvantage that you can not choose the file-name.
- Another advantage is that the function xlsload automatically clears the memory. **If you run different datasets** in a row and import data manually, you have to clear otherwise the memory beforehand (Matlab-Command: clear all)

4. Run NEUSREL

- Run NEUSREL by typing NEUSREL into the Matlab commando window and confirm with Enter.

Or **Run PLS:**

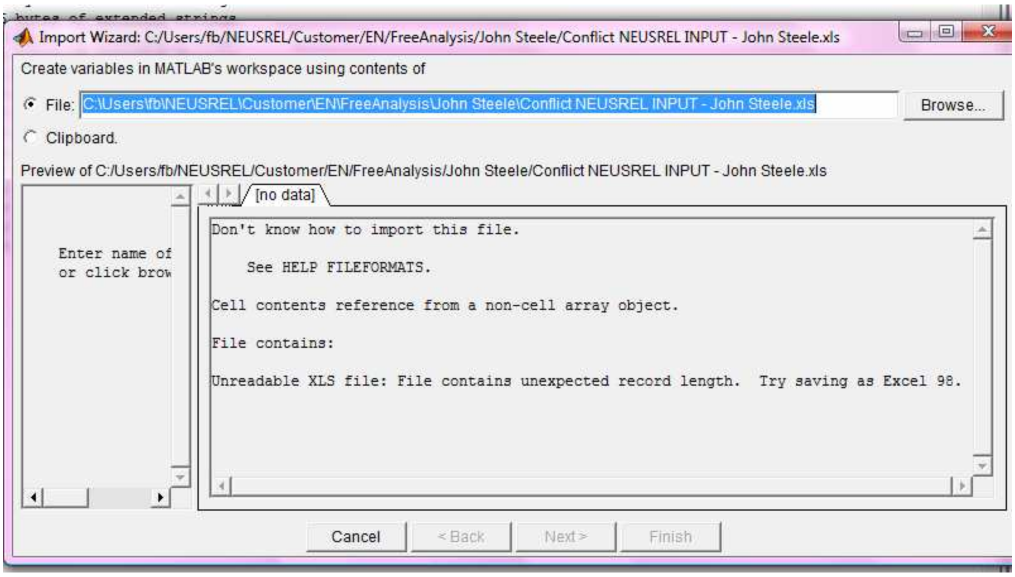
For comparison reasons user might wish to run alternatively a standard linear PLS analysis.

- Run a PLS analysis by typing PLS into the Matlab commando window and confirm with Enter.

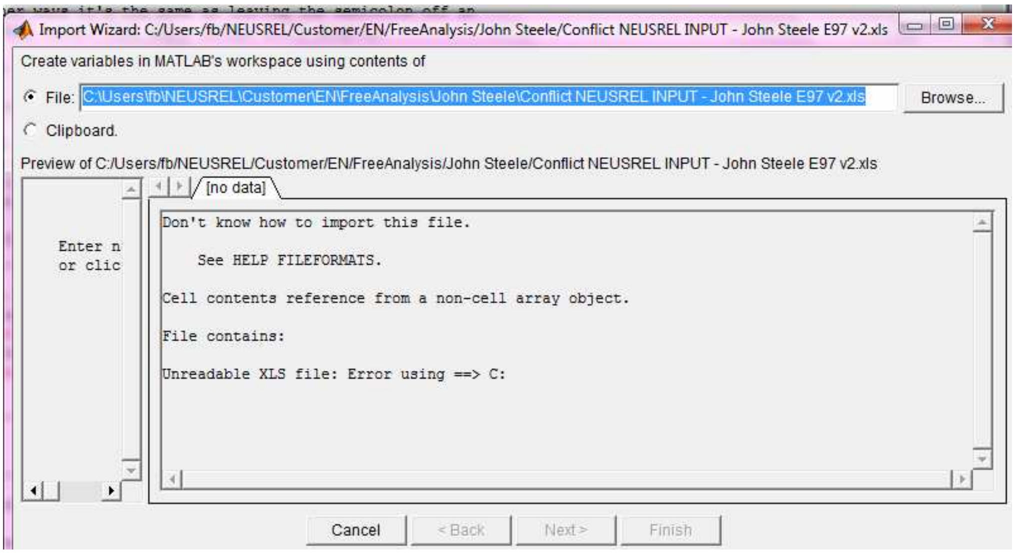
Error / Failures Messages

Can’t import data, what to do?

If you get an error massage as:



Or



You should save the same file as Excel 2003 or lower format.

What does this warning mean?

Error 1:

```
??? Error using ==> imwrite
Could not open file for writing.  Check directory or file permissions.
…
```

The label for one of the latent variables you defined in the Excel template file is not allowed, e.g. “/”. Check Variable Names and start again.

Error 2:

```
?? Error using ==> eig
NaN or Inf prevents convergence.
…
```

Your data contains NaNs or other not-a-number cells. Please review your Excel template file.

Error 3

```
Warning: Divide by zero.
> In c:\users\fb\neusrel\software\neusrel\bootstat.m at line 26
  In c:\users\fb\neusrel\software\neusrel\SN3.m at line 150
  In c:\users\fb\neusrel\software\neusrel\NEUSREL.m at line 34
```

When significance becomes infinitively high such warning appears due to numerical reasons. But for you it’s good news since your model seams to be quite good. The calculation is not aborted.

Error 4

```
??? Error using ==> diary
CSI_log.txt: Cannot open file: permission denied.
```

You have the logfile open. That’s why Matlab can not write into it. Close the logfile and type in the Matlab command window: logresults Then press Enter/Return.

Error 5

```
??? Undefined function or method 'Project' for input arguments of type 'double'.
```

You are probably using a student version (or similar) of MATLAB and you tried to import your data manually. Please name the input xls file "NEUSREL.xls" and use the "xlsload" command to load your data, as described above.

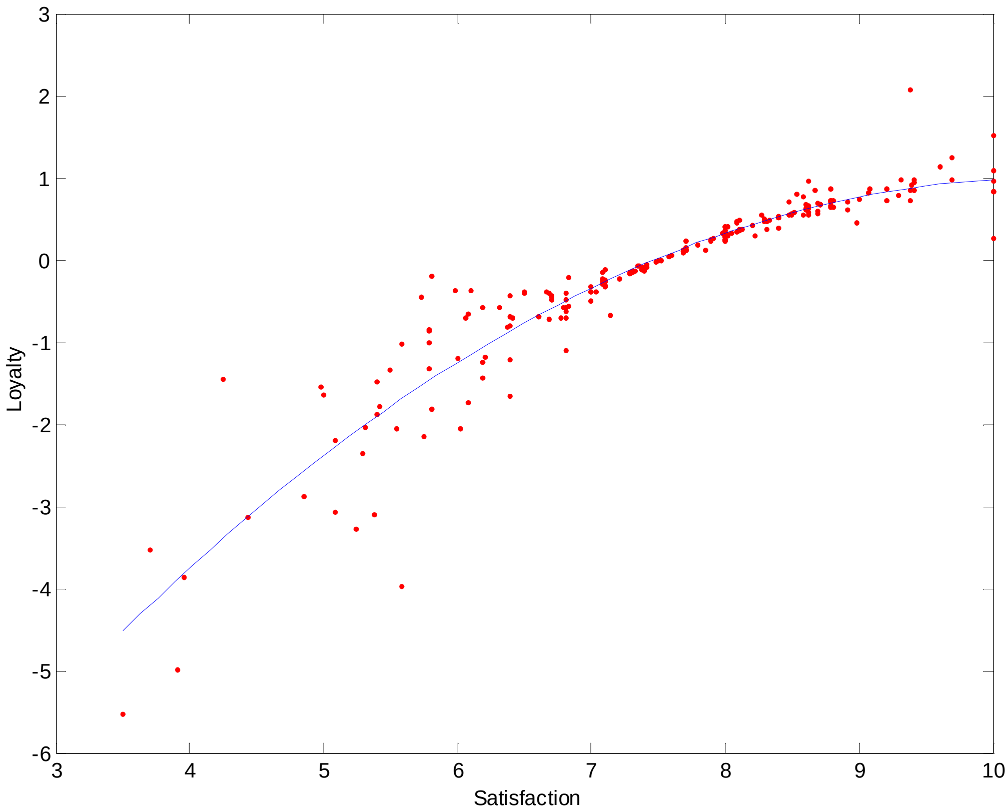
Interpretation of Results

Note: All graphs are saved in the active path as GIF. Default is: ...\Matlab\work

Interpreting 2D Plots: The Additive Plots

General Interpretation

The following plot is showing an example you will see in an extra Matlab figure window. You can save the plot either by saving as *.fig file, which can then only be viewed with Matlab. The only advantage of this format is that you can manipulate it easily within the Matlab window. You can save it also in other formats as JPG or EPS. The latter is scalable and has the best quality for later usage e.g. in Microsoft Word.



W

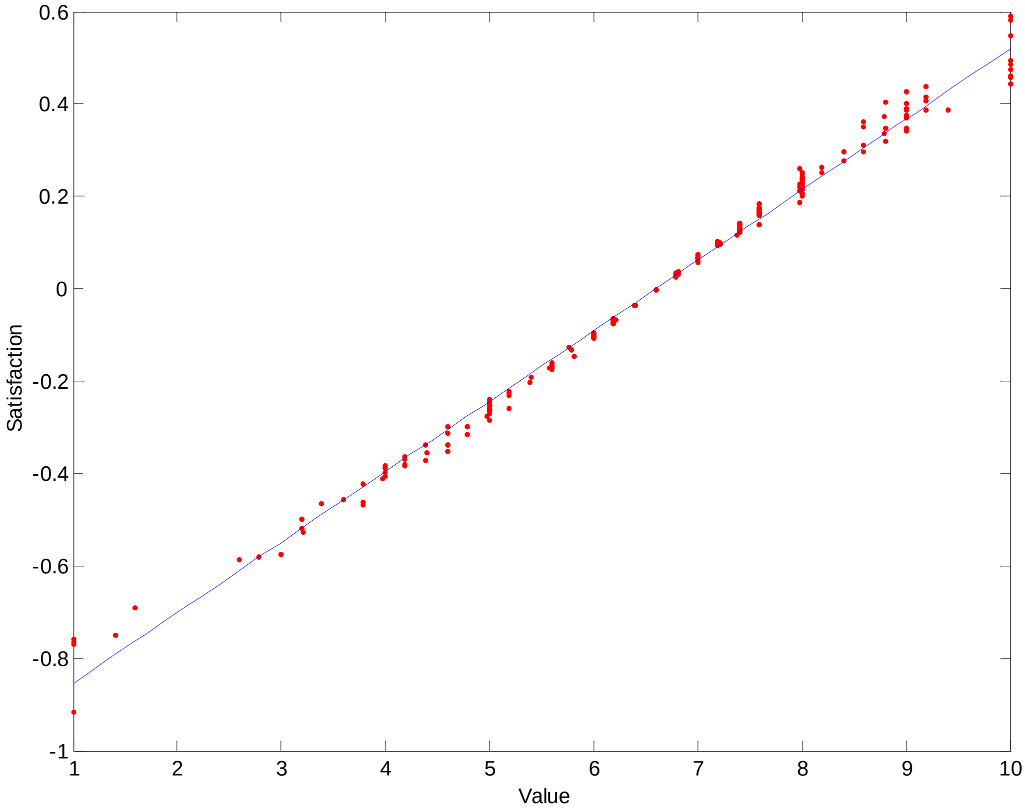
hat do you see on this plot?

- On the x-axis you see a causing latent variable (here Satisfaction). On y-axis you see the effected latent variable (Loyalty). The name that you defined in the Excel-Sheet (cells A2-A…) you will see here.
- All cases are represented by red dots.
- The blue line represents the estimated additive function from x on y.
- The scale 3 to 10 is the bandwidth of data of the latent variable "Satisfaction": Survey data normally show integer values, but the combination of integer items in the measurement results in "sub-integer" values in latent variables. Furthermore the substitution of missing values generates non-integer value.
- The y-axis scales always around zero. It shows the VARIATION in the y-variable caused by the x-variable. In this example, increasing "Satisfaction" from 4 to 6 would cause an incremental increase of "Loyalty" of 2,5 on average on the respective scale of "Loyalty" (which was in this case 1 to 10). In contrast an increase from 8 to 10 in "Satisfaction" causes only approx. 0.5 increase in "Loyalty".
- Overall interpretation of this example: **You see a positive degressive relationship.** Meaning that increasing lower x-values have higher positive effects on y than higher x-values. This effect is also measure by the Leverage Factor, which is described in the next chapter.

Other Examples

<u>Linearity</u>

Especially when you analyse well known theoretical models, where structural and measurement as well as the scales where "trimmed" over decades of scientific research you are likely to find for most relations linear relationship. If not the classical SEM method would not have generated acceptable results and consequently the measurement model would not have been excepted or used either.

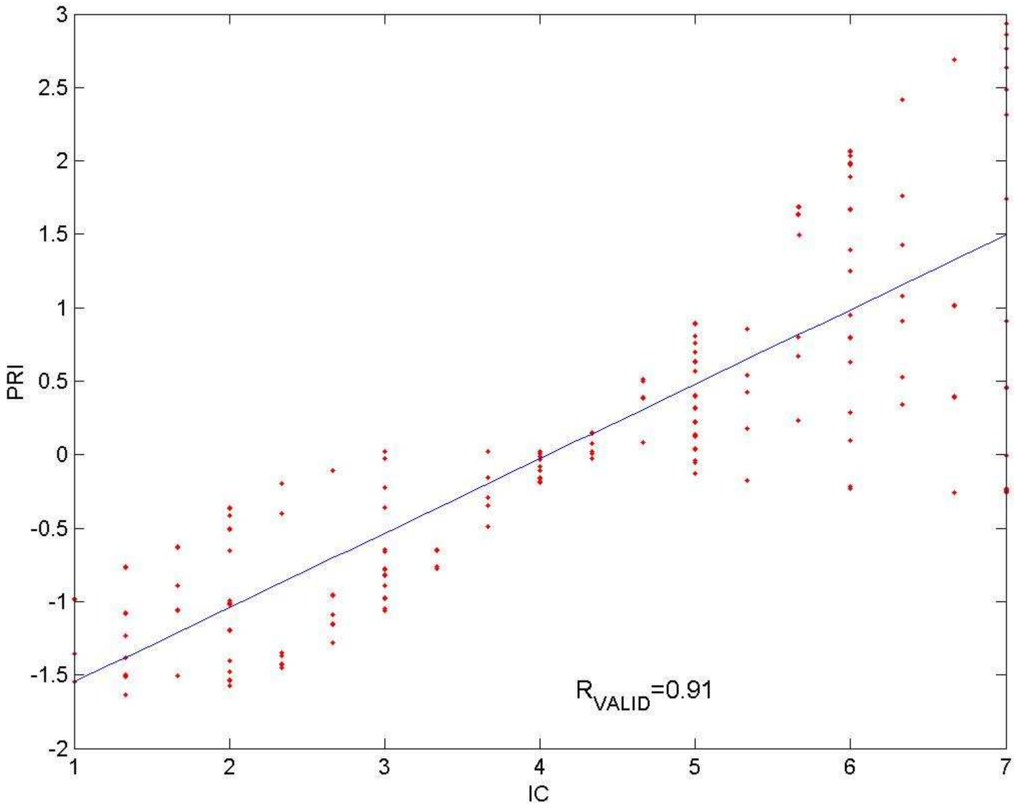


Here you see the perceived “Value” influencing the customer “Satisfaction”. Increasing the perceived “Value” by 9 causes the satisfaction to rise by 1,3.

Here you can see some variation (red) around the linear function (blue) which might lead you to think this is a slidely nonlinear relation. However, a cross-validation algorithm has decided that the linear function provides a more generalizable assumption compared to a quadratic function.

Interactions

You will also experience plots like the following.



Drawing from previous experiences, researchers tend to interpret a linear relationship with some uncertainty. This is NOT exactly true.

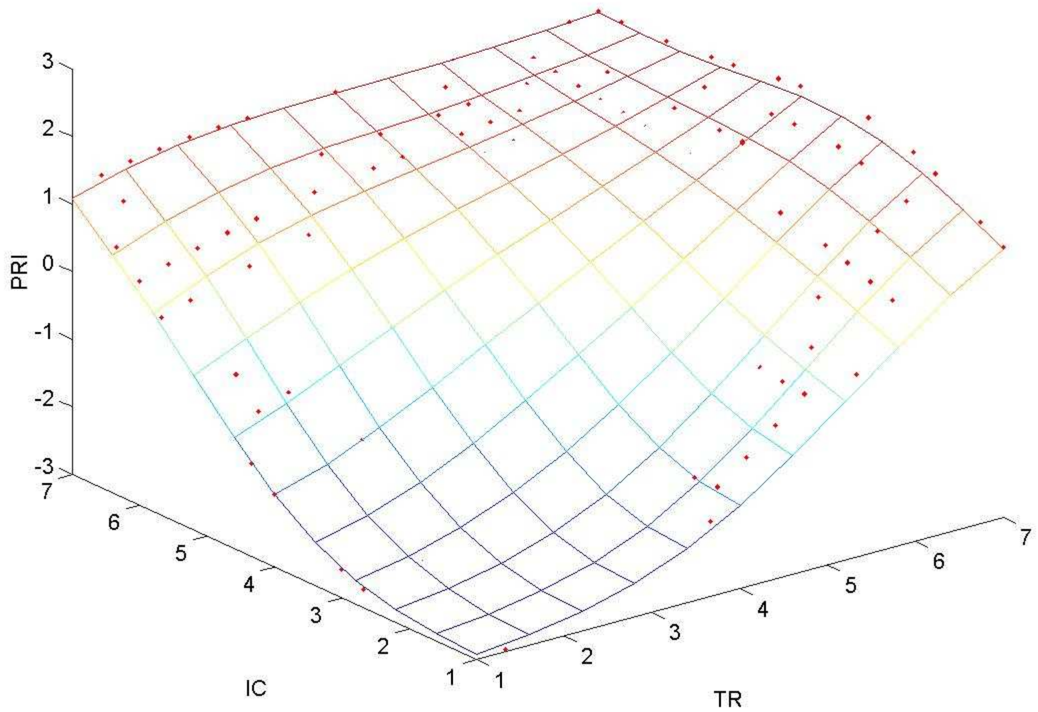
All variation in the Additive Plots around the blue line is caused by INTERACTIONS and NOT by noise/error/residuals. Taking this into account, we can conclude:

- IC influenced PRI heavily by interacting with an other variable
- The influence direction is positive and at least close to linearity – although exact linearity can not be concluded with certainty.

How this particular interaction with the other variable looks like can be seen in the following chapter.

General Interpretation

The following plot shows a typical interaction plot. NEUSREL generate plots only when a minimum interaction effect is given.



- Again the vertical axis is the effected (explained) latent variable, showing differences caused by the causes around zero.
- The interaction function is represented by a colour-coded surface. The colour indicate the height on the scale of the effected (explained) latent variable
- All cases can be seen as red dots, although roughly half of cases are hidden below the surface. Matlab allows rotating the graph, as you like. So you can take a look below the surface.

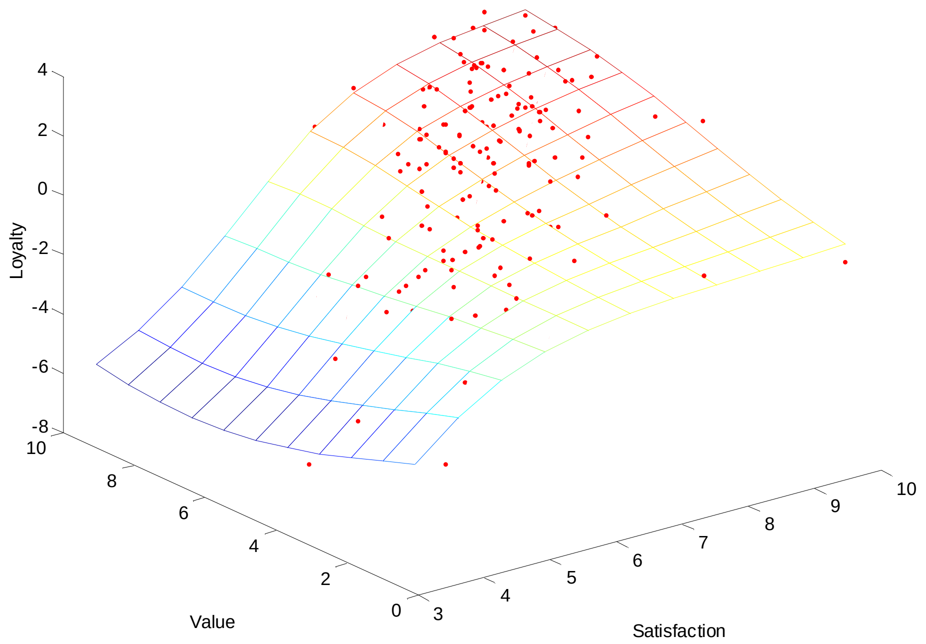
Most users help the following exercise to understand the concept of the plots:
Using this rotation function you could rotate the plot so that you retrieve a 2D plot - “IC” is at the x-axis and PRI at y-axis. What you get by doing this is similar to the Additive Plot (from previous section).

Interpretation of this example plot

- When the other latent variable is low, IC as well as TR shows positive, slidly progressive influence on PRI.
- When the other latent variable is high, IC as well as TR show minor influence on PRI
- This type of interactions can be named as “compensating”. Meaning either you apply one or the other cause to induce an effect. Applying both causes will not result in major changes in results compared to applying only one cause.

Other Examples

“Conditionary interaction” or “Amplifying Moderator”.



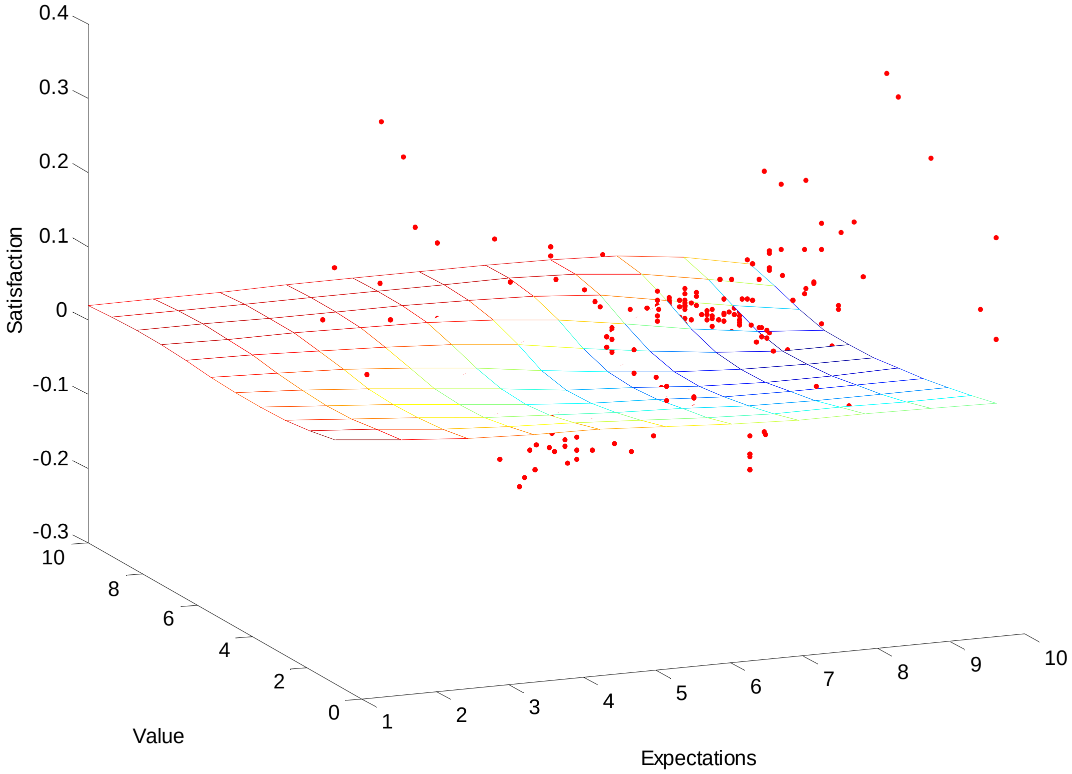
Interpretation:

- “Value” alone has only a linear positive effect on “Loyalty”, when “Satisfaction” is high.
- “Satisfaction” alone has a positive degressive effect on “Loyalty”, which get strong when “Value” is high
- Strong “Loyalty” can only be achieved with high “Satisfaction” AND high “Value”
- In that sense each cause amplifies the effect of the other. It can be seen as a conditionary interaction in the sense that maximises only at the condition that both causes show high values.

3-way interactions

The 3D plots show only two-way interactions. If a third cause is involved in the interaction, it can be seen as a big variation around the surface. An example can be seen in the following plot. This plotted interaction of “Value” and “Expectation” on

“Satisfaction” seems to be very low as can be read on the vertical scale. In contrast the cases (red dots) vary heavily around the surface.

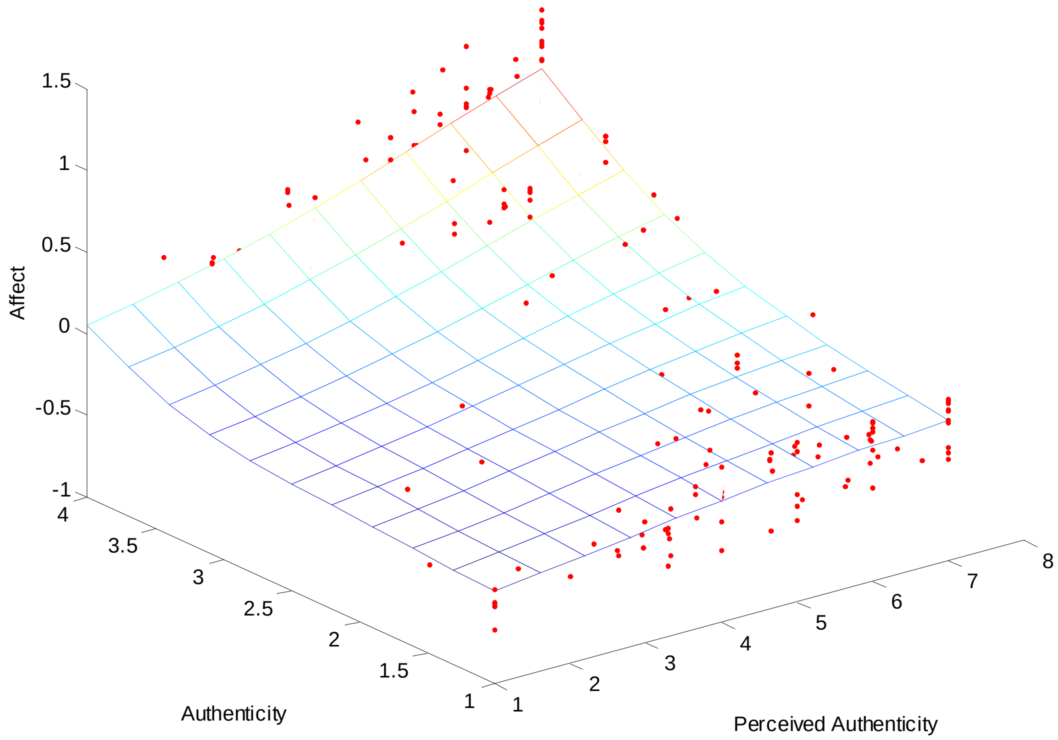


Conditional and 3-way Interactions

The following example shows both, a conditional interaction and some sign of a third interacting variable.

Conditional interaction: As you see neither “Authenticity” nor “Perceived Authenticity” have alone a significant effect on “Affect”. Only if both elements are high, a significant effect on “Affect” can be encountered.

3-way Interaction: Again there a notably variation of cases around the surface. This indicates that a third cause has influence on the cause-effect-relationship of “Authenticity” and “Perceived Authenticity” on “Affect”.



Interpreting Coefficients

All coefficients can be found in the log-textfile named according to your definition of "Projectname" in cell B1 in the Excel input file: **"*projectname*_log.txt"**

The textfile is saved in the active path. Default is: ...\Matlab\work\

Logfile starts like this example:

```
================================
CSI
================================
27-Jul-2009
>> Calculation time in hours:_0.13878
>> Number of cases for model estimation:_250
Latent Variables
    'Quality'
    'Expectation'
    'Value'
    'Satisfaction'
    'Loyalty'
    'Complaints'

Apriori Matrix
        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0         0.5         0.5         0.5
2_Expec     0.5         0.5         0.5         0.5
3_Value       0         0.5         0.5         0.5
4_Satis       0           0         0.5         0.5
5_Loyal       0           0           0           0
6_Compl       0           0         0.5           0
```

First it states the project name, the needed calculation time, the used training cases and the names of LV. Furthermore you see the A-priori Matrix you've defined. You can expect coefficients only for values above 0, since all others had been excluded.

R2 - Total variance of an endogenous latent variable explained by all other model constructs

This measure is known from standard multivariate analysis. It measures how much of the variance of a LV can be explained (using a multivariate estimation method) by all other LVs. It is only computed for LVs that are defined to be influenced by other LVs. R2 scales from 0 (equals "no explanation") to 1 (equals "every variation in data can be explained by the LV within the model").

The values can be found in the log-file in the order of LVs according to your definition in cell A2-A…

```
================================
R2
        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
         0.19        0.66        0.53        0.34
R2 valid
  >> No validation data defined
R2 for linear structural model - similar PLS
        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
         0.19        0.65        0.45        0.31
================================
```

In this example “Quality” was the first LV and gets explained by NEUSREL with 0.19. The fives latent variable “Loyalty” get explained by NEUSREL with 0.53. Below you find R2 values for the case if not Artificial Neural Networks had been used but Multivariate Linear Regression (like in PLS). You find in this example similar R2 except for “Loyalty”. The reason is that NEUSREL has found nonlinearities only in the causes of “Loyalty”. The plot in the beginning of previous chapter showed the effect.

Goodness Of Fit - GoF

GoF is an overall coefficient to measure how well the data can explain each other. It takes all R2 into account to measure the explanation of the structural model. AND it takes into account how well the manifest variables explain the latent variables (R2s). All together result in the GoF measure. Again, the measure scales from 0 (no fit at all) to 1 (perfect fit, no errors/residuals neither in measurement nor in structural model).

It is important to combine the fitness of structural and measurement model, since one can improve one side by weakening the other. SEM models (Lisrel, Amos, etc.) favour to fit structural models on the expense of measurement models. They produce high R2’s but lower GoF’s.

```
================================
GOF -Goodness Of Fit
    0.5758    0.5912    0.5907
================================
```

Here you see the progress of GoF over the iterations. This example has only 3 iterations. Some think about calculating a lot of iterations and choosing the best. This is by purpose not supported by NEUSREL, since this is an effort of overfitting. Furthermore in our experience the models GoF’s make 80% improvements in the second iteration and sometimes even decreases later for a while. And of cause, a model is not just true because the GoF is high. It is true based on your true assumptions and a high GoF. If you change assumptions in the process you fool yourself and others. So choosing e.g. one iteration only, might be a conscious act of trusting the measurement model over your data and you structurally set limitations (A-priori Matrix).

Factor Score

The resulting factor scores for every MV are also printed.

```
FactorScores
    0.7990    0.6528    0.7877    0.7571    0.7648    0.7677    0.7753
    1.0000         0         0         0         0         0         0
    0.9515    0.8858         0         0         0         0         0
    0.8018    0.8535    0.8429         0         0         0         0
    0.8327    0.9165         0         0         0         0         0
    1.0000         0         0         0         0         0         0
```

In the rows you find factor scores belonging to the respective latent variable. So row 1 is “Quality” and so forth. In the columns you find the scores in the order defined in you data set (in the excel-sheet “data”)

Measurement model reliability measures

The validity of every latent variable (LV) is expressed by reliability measures as Chrombachs Alpha, Average Explained Variance (AEV) or Composite Reliability:.

```
================================
1. Chrombachs Alpha | 2. AEV Average Explained Variance | 3.Composite Reliability
    0.8770    0.5741    0.7595
         0         0         0
    0.8236    0.2414    0.1069
    0.7792    0.2915    0.1975
    0.7027    0.2174    0.0916
         0         0         0
```

Overall Explained Absolute Deviation OEAD

Overall Explained Absolute Deviation (OEAD) is the share of variance of a latent variable i’s which is explained by latent variable j in the structural model. It scales from 0 (equal to “no influence) to about the value of R2 of the LV (equals “LV i is by 100% influenced by LV j”). The measure does not say anything about the direction or property of the relationship. This enables us to compare linear with nonlinear and interactive effects.

```
================================
OEAD - Overall Explained Absolute Deviation
        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0         0.53          0        0.15
2_Expec    0.19            0          0           0
3_Value       0         0.11       0.02        0.01
4_Satis       0            0       0.38        0.17
5_Loyal       0            0          0           0
6_Compl       0            0       0.09           0
................................
OEAD t-values
        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali     NaN         7.48       0.71        2.06
2_Expec    5.47         1.16       1.11        1.16
3_Value     NaN         3.34       1.64        1.27
4_Satis     NaN          NaN        6.5        3.47
5_Loyal     NaN          NaN        NaN         NaN
6_Compl     NaN          NaN       2.29         NaN
................................
OEAD significance level according to t-values
        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0         0.01        0.2        0.03
2_Expec    0.01         0.15       0.15        0.15
3_Value       0         0.01        0.1        0.15
4_Satis       0            0       0.01        0.01
5_Loyal       0            0          0           0
================================
```

Again the rows represent the causing and the columns the effected latent variables. In row 4 (Satisfaction) and Column 5 (Loyalty) you find a value of 0.38, which is quite a strong influence on the R2=0.53 explained variable. That means that nearly two thirds of “Loyalty” is explained somehow by “Satisfaction”. Significance level in the same cell shows a confidence level of 1% and the respective t-value of 6.58. “NaN” stands for “Not a Number” and is a placeholder for “I did not calculate a value for that cell” – since these relations where excluded by you.

Degree Of Freedom DF

With DF we describe the level of nonlinearity of a given additive effect from LV i on j.

- DF=0 no effect analysed
- DF=1 linear effect (One-Degree-Polynom-Regression $y=a*x+const$)
- DF=2 quadratic effect (Two-Degree-Polynom-Regression $y=b*x^2+a*x+const$)
- DF=3 cubic effect (Three-Degree-Polynom-Regression $y=c*x^3+b*x^2+a*x+const$)
- DF>3 … analogue scheme

```
================================
DF - Degree of Freedom = Existents of Nonlinear Relationships
        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0           1           2           1
2_Expec       1           2           1           0
3_Value       0           1           2           2
4_Satis       0           0           2           1
5_Loyal       0           0           0           0
6_Compl       0           0           2           0
```

In this example you find a „0“ for all excluded paths and a “1” for all to be found linear. The relation “Satisfaction on Loyalty” in row 4 and column 5 has DF=2. This means you need a quadratic form for building the positive degressive effect we showed in previous chapter. All other nonlinear values of DF>1 are not very significant, as you can read in OEAD significance level table.

Linear path coefficients LPC

For every linear path (cause-effect-relation with DF=1) NEUSREL computes a standardized linear path coefficient. This can be compared to those computed by other causal analysis methods as PLS, LISREL or AMOS. In fact, results of NEUSREL are quite similar to PLS, when the relations are linear and additive (=non-interactive).

```
================================
LPC - Linear Path Coefficients
        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0        0.67           0        0.25
2_Expec    0.44           0        0.01       -0.05
3_Value       0        0.19           0           0
4_Satis       0           0           0        0.31
5_Loyal       0           0           0           0
6_Compl       0           0           0           0
................................
LPC t-values
        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali     NaN         7.5        0.65           2
2_Expec    5.54        0.86        0.89        1.19
3_Value     NaN        3.14        1.46        0.88
4_Satis     NaN         NaN        4.52        3.28
5_Loyal     NaN         NaN         NaN         NaN
6_Compl     NaN         NaN        1.75         NaN
................................
LPC significance level according to t-values
        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0        0.01        0.25        0.05
2_Expec    0.01         0.2         0.2        0.15
3_Value       0        0.01         0.1         0.2
4_Satis       0           0        0.01        0.01
5_Loyal       0           0           0           0
6_Compl       0           0        0.05           0
```

The significance level for LPC may seem to appear overstated when comparing with PLS and OEAD values. We suggest to always refer to the OEAD significance level values as the more general figure, when judging about significance.

Polynomial Coefficients of Nonlinear Effects

For all nonlinear additive plots the polynomial coefficients are included in the logfile. polynomial regression has the structure as follows:

$y= a_0 + x \cdot a_1 + x^2 \cdot a_2 + x^3 \cdot a_3 \ldots \cdot + x^n \cdot a_n$

The printed coefficients “a” are printed in the order of : a_0 a_1 a_2 a_3 ... a_n

```
================================
Polynomial Coefficients of Nonlinear Effects
--------------------------------------------
................................
Cause: Expectation    Effect: Satisfaction
    -> Coefficients: 0.0045169   -0.050158     0.12123    0.054318    0.033915           0
    -> Leverage-Factor: -1.9612
................................
Cause: Quality    Effect: Loyalty
    -> Coefficients: 0.018196    -0.28594      1.1309           0           0           0
    -> Leverage-Factor: -0.43544
................................
Cause: Value    Effect: Loyalty
    -> Coefficients: 0.040688    -0.35211     0.61671           0           0           0
    -> Leverage-Factor: -6.1292
................................
Cause: Satisfaction    Effect: Loyalty
    -> Coefficients: -0.115631      2.45318     -11.8938           0            0            0
    -> Leverage-Factor: 0.33079
```

What means -> Leverage Factor: ?

For second-order nonlinearities we experience either progressive or degressive functions. Degressive means that a change in the effected variable is higher for low values of the causing variable than for high values. The opposite is true for progressive functions – the effect of high values is higher as with low values. The Leverage-Factor is quantifying this. It is the quotient of a linear path coefficient calculated only for values *higher than average* divided through the linear path coefficient calculated only for values *lower than average*. Consequently a Leverage Factor **higher than 1 indicates a progressive relationship** and lower than 1 a degressive relationship. A value of 2 means that a change in higher than average values causes effects of the double extent than for lower then average values. A negative Leverage Factor express that function at hand is of U- or inverted-U-shape.

Interaction Effect IE

The strength of an interactive effect is measured by IE. It measures how much of the explained variance of a latent variable i can be explained by interaction effects caused by latent variables j and k. The coefficient scales from 0 (equals “no interaction effect of j and k”) to about the value of R2 of the LV (equals “all variance can be explained exclusively by interacting LV j and k and no additive effects are present”).

The output may spread several pages as the following example. The first matrix shows the interaction of the first LV (here: Quality) with the LV in the rows, when influencing the LVs in the columns. The highest values we find in the tables for Satisfaction and Value. With 0.13 Satisfaction and Value interact the most when effecting Loyalty. The value 0.13 you find in both tables (Satisfaction: row 3, column 5 resp. Value: row 4, column 5). The values of both twin cells may slidly deviate from each other.

```
IE - Interaction Effects
    'Quality'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0           0           0           0
2_Expec       0        0.14           0        0.04
3_Value       0        0.12        0.08           0
4_Satis       0           0         0.1        0.03
5_Loyal       0           0           0           0
6_Compl       0           0        0.07           0
    'Expectation'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0        0.14           0        0.04
2_Expec       0           0           0           0
3_Value       0        0.02        0.02        0.03
4_Satis       0           0        0.12        0.08
5_Loyal       0           0           0           0
6_Compl       0           0        0.05           0
    'Value'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0        0.12        0.08           0
2_Expec       0        0.03        0.02           0
3_Value       0           0           0           0
4_Satis       0           0        0.13           0
5_Loyal       0           0           0           0
6_Compl       0           0        0.11           0
    'Satisfaction'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0           0        0.09           0
2_Expec       0           0        0.11           0
3_Value       0           0        0.13           0
4_Satis       0           0           0           0
5_Loyal       0           0           0           0
6_Compl       0           0        0.04           0
    'Loyalty'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0           0           0           0
2_Expec       0           0           0           0
```

```
3_Value       0           0           0           0
4_Satis       0           0           0           0
5_Loyal       0           0           0           0
6_Compl       0           0           0           0
    'Complaints'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0           0        0.08           0
2_Expec       0           0        0.06           0
3_Value       0           0         0.1           0
4_Satis       0           0        0.04           0
5_Loyal       0           0           0           0
6_Compl       0           0           0           0
................................
Leverage Factors thru IE
    'Quality'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint NaN
1_Quali         0             0             0             0
2_Expec         0          0.78         16.89           3.2
3_Value         0          0.72         -0.54             0
4_Satis         0             0         -0.54             0
5_Loyal         0             0             0             0
6_Compl         0             0       -168.76             0
    'Expectation'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali        0          6.37        91.95         -0.88
2_Expec        0             0            0             0
3_Value        0         15.74        -5.92          1.63
4_Satis        0             0        -5.16         -0.82
5_Loyal        0             0            0             0
6_Compl        0             0        91.64             0
    'Value'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0        1.94       -2.38           0
2_Expec       0        3.95        0.38           0
3_Value       0           0           0           0
4_Satis       0           0       -0.82           0
5_Loyal       0           0           0           0
6_Compl       0           0        1.82           0
    'Satisfaction'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0           0        0.27           0
2_Expec       0           0         0.3           0
3_Value       0           0        0.25           0
4_Satis       0           0           0           0
5_Loyal       0           0           0           0
6_Compl       0           0        0.22           0
    'Loyalty'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0           0           0           0
2_Expec       0           0           0           0
3_Value       0           0           0           0
4_Satis       0           0           0           0
5_Loyal       0           0           0           0
6_Compl       0           0           0           0
    'Complaints'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali        0             0         27.5             0
2_Expec        0             0        63.67             0
3_Value        0             0       -26.45             0
4_Satis        0             0         0.06             0
5_Loyal        0             0            0             0
6_Compl        0             0            0             0
```

What means -> Leverage Factor thru IE: ?

For interactions we experience often moderating effects in the sense that the effect strength of causes heavily depends on another moderating variable. A moderator can suppress or amplify the effect of a cause. In order to make this effect more tangible to users the Leverage-Factor is quantifying this effect. It is the quotient of a linear path coefficient (of a cause on an effect) calculated only for cases with *higher than average* values of the moderator variable divided through the linear path coefficient calculated only for cases with *lower than average* values of the moderator variable. Consequently a Leverage Factor **higher than 1 indicates an amplifying effect of the moderator** in the relation at hand. A value of 2 means that the effect is double when *higher* than average values of the moderator are given compared to the effect when *lower* than average values of the moderator are given. A negative Leverage Factor express that the moderator reverses the effect of the cause.

After that t-values and significance level is shown, if you have choosen this calculation within the Excel input file. The calculation may take a while. Every figure requires to calculate a new Neural Network for every Bootstrapping sample. The following results needed 19 hours to calculate on a 2GHz, Centrino Prozessor Laptop with only 50 Bootstrapping samples (and 7 hours on a Desktop-PC with Intel(R) Core(TM)2 Duo CPU E8500 @ 3.16GHz).

```
...............................
IE t-values
    'Quality'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali     NaN         NaN         NaN         NaN
2_Expec     NaN        0.14        0.14        1.34
3_Value     NaN        0.14        0.14        1.37
4_Satis     NaN         NaN        0.14        1.56
5_Loyal     NaN         NaN         NaN         NaN
6_Compl     NaN         NaN        0.14         NaN
    'Expectation'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali     NaN        0.14        0.14        1.34
2_Expec     NaN         NaN         NaN         NaN
3_Value     NaN        0.14        0.14        1.49
4_Satis     NaN         NaN        0.14        1.34
5_Loyal     NaN         NaN         NaN         NaN
6_Compl     NaN         NaN        0.14         NaN
    'Value'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali     NaN        0.14        1.41         NaN
2_Expec     NaN        0.14        1.31         NaN
```

```
3_Value     NaN         NaN          NaN         NaN
4_Satis     NaN         NaN         1.54         NaN
5_Loyal     NaN         NaN          NaN         NaN
6_Compl     NaN         NaN         1.63         NaN
    'Satisfaction'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali     NaN         NaN         1.23        1.54
2_Expec     NaN         NaN         1.26        1.31
3_Value     NaN         NaN         1.55        1.28
4_Satis     NaN         NaN          NaN         NaN
5_Loyal     NaN         NaN          NaN         NaN
6_Compl     NaN         NaN         1.51         NaN
    'Loyalty'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali     NaN         NaN          NaN         NaN
2_Expec     NaN         NaN          NaN         NaN
3_Value     NaN         NaN          NaN         NaN
4_Satis     NaN         NaN          NaN         NaN
5_Loyal     NaN         NaN          NaN         NaN
6_Compl     NaN         NaN          NaN         NaN
    'Complaints'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali     NaN         NaN         1.35         NaN
2_Expec     NaN         NaN         1.38         NaN
3_Value     NaN         NaN         1.63         NaN
4_Satis     NaN         NaN         1.51         NaN
5_Loyal     NaN         NaN          NaN         NaN
6_Compl     NaN         NaN          NaN         NaN
................................
IE significance level according to t-values
    'Quality'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0           0           0           0
2_Expec       0        0.35        0.35         0.1
3_Value       0        0.35        0.35         0.1
4_Satis       0           0        0.35         0.1
5_Loyal       0           0           0           0
6_Compl       0           0        0.35           0
    'Expectation'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0        0.35        0.35         0.1
2_Expec       0           0           0           0
3_Value       0        0.35        0.35         0.1
4_Satis       0           0        0.35         0.1
5_Loyal       0           0           0           0
6_Compl       0           0        0.35           0
    'Value'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0        0.35         0.1           0
2_Expec       0        0.35         0.1           0
3_Value       0           0           0           0
4_Satis       0           0        0.05           0
5_Loyal       0           0           0           0
6_Compl       0           0         0.1           0
    'Satisfaction'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0           0        0.15         0.1
2_Expec       0           0        0.15         0.1
3_Value       0           0        0.05         0.15
4_Satis       0           0           0           0
5_Loyal       0           0           0           0
6_Compl       0           0         0.1           0
    'Loyalty'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0           0           0           0
```

```
2_Expec       0           0           0           0
3_Value       0           0           0           0
4_Satis       0           0           0           0
5_Loyal       0           0           0           0
6_Compl       0           0           0           0
    'Complaints'

        1_Quality   4_Satisfact 5_Loyalty   6_Complaint
1_Quali       0           0         0.1           0
2_Expec       0           0         0.1           0
3_Value       0           0         0.1           0
4_Satis       0           0         0.1           0
5_Loyal       0           0           0           0
6_Compl       0           0           0           0
```

In this example there are only two IE measures with a significance level of 5%: Satisfaction and Value on Loyalty.

Module: Second Order

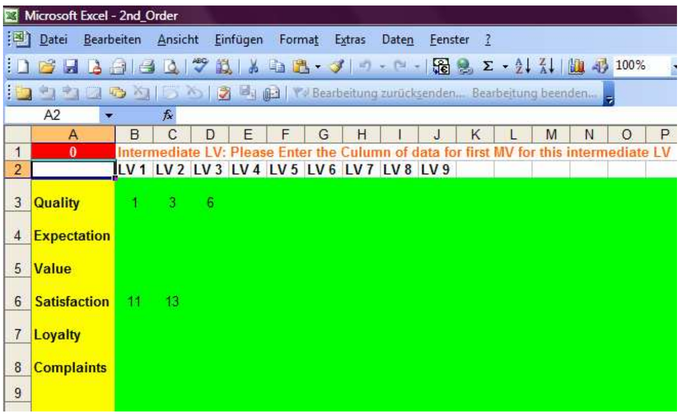
Since version 2.3, NEUSREL allows also to consider 2nd order constructs. These are latent variables that are only defined thru latent variables (of lower generalisation status). Meaning, groups of MVs are compressed to several LV's of first order, which are still not part of the structure model. They get compressed to the LV of 2nd order which is part of the structure model.

NEUSREL treats this problem as follows. First a NEUSREL model is trained neglecting LVs of 1st order and assuming MVs define the 2nd LV. Hence the first run is as usual.

After having run a NEUSREL analysis, the definition of the 2nd measurement models get specified. 1st order LVs are computed using PCA (principle component analysis). Then factor loadings are deducted either using multiple regression or bivariate regression, depending on the path direction (formative or reflexive).

How to set up a 2nd order calculation?

In the NEUSREL directory you find an Excel file called "2nd_Order.xls".



Enable for second order calculation by setting the red cell (A1) to '1' and set to '0' to disable.

In the yellow column (A) you find the same LVs as defined in the standard input file in column (A). The only thing you have to do in this sheet is to fill in the green cells and here the rows in which you find the 2nd order LVs. The number you put in here is the number of the first MV belonging to LV of 1st order defined in this column. For example you find above the 2nd order LV “Quality”. As we had define in standard input file MV 1 to 7 belong to Quality. Here in row 3 and columns B to D you see that the first 1st order LV is defined by MV 1 to 2, the second LV of 1st order is defined by MV 3 to 5 and the last by MV 6 and 7.

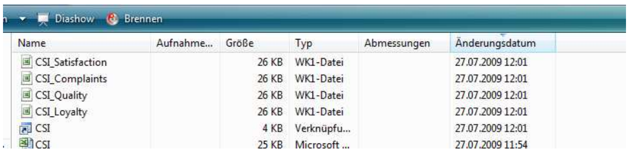
The output in the logfile looks as follows

```
================================
Second Order Factor Scores
--------------------------------
    'Quality'

First Order Scores:
    0.7385    0.9557         0
    0.8809    0.8144    0.7979
    0.8097    0.9137         0

Second Order Scores:
    0.8152
    0.8978
    0.8994

--------------------------------
    'Satisfaction'

First Order Scores:
    0.7337    0.9678
    1.0000         0

Second Order Scores:
    0.9414
    0.9199

--------------------------------
```

You can read here the logfile output for every 2nd order LV (like in this example ‘Quality’ and ‘Satisfaction’). Below “First Order Scores:” you find in every row the Factor Scores that specify the respective LV of first order. E.g., hence in first row you find the two Factor Scores for MV 1 und 2. Below “Second Order Scores:” you find the Factor Scores that define the 2nd order LV. Hence the first value is for the first LV 1st order.

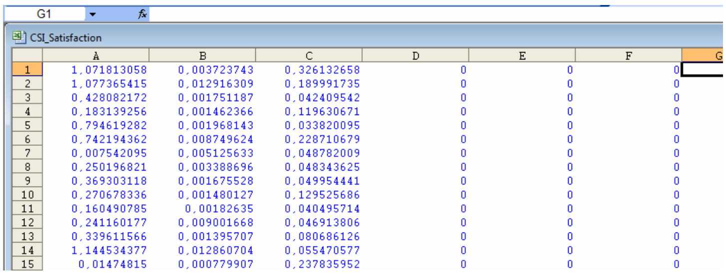
Module: Hierarchical Bayes

Since version 2.4, NEUSREL allows for calculation of case-specific path strength OEAD values. The values are saved in your current directory:



Name	Aufnahme...	Größe	Typ	Abmessungen	Änderungsdatum
CSI_Satisfaction		26 KB	WK1-Datei		27.07.2009 12:01
CSI_Complaints		26 KB	WK1-Datei		27.07.2009 12:01
CSI_Quality		26 KB	WK1-Datei		27.07.2009 12:01
CSI_Loyalty		26 KB	WK1-Datei		27.07.2009 12:01
CSI		4 KB	Verknüpfu...		27.07.2009 12:01
CSI		25 KB	Microsoft ...		27.07.2009 11:54

For every endogen LV a separate data matrix is save using the WK1-format, that can be read by Excel. A row contents the OEAD values of a case. The first row is for the first case in the same order as copied in the input xls-file. The same is true for the columns. The first column corresponds to the first LV defined in the input xls-file. Hence, in the screenshot below you see in Column A all OEAD values of the path, that describes the effect of Quality on Satisfaction.



CSI_Satisfaction

	A	B	C	D	E	F	G
1	1,071813058	0,003723743	0,326132658	0	0	0	
2	1,077365415	0,012916309	0,189991735	0	0	0	
3	0,428082172	0,001751187	0,042409542	0	0	0	
4	0,183139256	0,001462366	0,119630671	0	0	0	
5	0,794619282	0,001968143	0,033820095	0	0	0	
6	0,742194362	0,008749624	0,228710679	0	0	0	
7	0,007542095	0,005125633	0,048782009	0	0	0	
8	0,250196821	0,003388696	0,048343625	0	0	0	
9	0,369303118	0,001675528	0,049954441	0	0	0	
10	0,270678336	0,001480127	0,129525686	0	0	0	
11	0,160490785	0,00182635	0,040495714	0	0	0	
12	0,241160177	0,009001668	0,046913806	0	0	0	
13	0,339611566	0,001395707	0,080686126	0	0	0	
14	1,144534377	0,012860704	0,055470577	0	0	0	
15	0,01474815	0,000779907	0,237835952	0	0	0	

This data allows you for further process analysis in Excel or other statistical software as SPSS.

Questions and Answers

How long does calculation last?

The calculation time is dependent by many factors:

1. Your hardware speed (linear)
2. Number of cases (linear)
3. Number of hidden units (quadratic)
4. Number of Committee-Of-Network members (linear)
5. Number of Bootstrapping samples (linear)
6. Number of dependent/endogene/influenced/effected latent variables. (linear)
7. Number of independent/causing/exogene/influencing latent variables (linear)
8. The threashhold by that interaction plots should be calculated. With many variables (LVs) this calculations consumes the majority of time (quadratic)
9. If you choose to calculate bootstrapping samples for the interaction measure IE. (Requires as well huge amount of calculation time)

If you double each of the factors 2 to 7, you increase computation time at least by 2^6=64. That's why it is hard to make a general judgment about computation time.

From our experience most models are estimated without bootstrapping within several minutes and below one hour. When applying bootstrapping, calculation can last several hours (or even days when significance for IE is needed). In extreme cases (10.000 cases, bootstrapping incl. IE, a large model) the model might take few weeks to finish.

How many cases do I need?

Its minimum sample size requirements are basically the same as those for PLS when the true relations are linear and additive (i.e., 5–10 times the largest number of incoming paths for a latent model variable or formative items for a construct; Bentler and Chou 1987). Although nonlinearity and interactive effects require larger samples, our experiments give us the confidence to state that NEUSREL can locate nonlinearity and interactions with sample sizes of less than 250 cases. More detailed general recommendation are not possible since sample size requirement heavily depend on the complexity of the very specific problem.

Do we have “dependent” and “independent” variables?

The dependency you define by your own in the apriori matrix in sheet “Project” starting with cell F2. The rows are the “independent” variables and the columns the “dependent”. Nevertheless with NEUSREL every variable can be both.

How to get access to the case values of latent variables?

You will find the file **ExampleFilename.xls** in the directory you have defined in the path set up. In each row you find the cases of a latent variable. Accordingly, the first row contains the values of the first latent variable defined from you in the Excel Input Template ... and so forth.

Which bootstrap results reflect parameter significance?

Every significance measure is result of bootstrapping. Refer to "significance level according to t-values" as the standard why of analysing bootstrapping results.

How can I access the standard deviation of bootstrapping measures?

Type **Logstd** and the bootstrapping standard deviation values for LPCs (Linear Path Coefficients) and OEADs will be printed.

How can I close all these Matlab figures (plot windows)?

Just type in the Matlab Command Window the following command and press Enter:

```
>> close all
```

Not all equations are featured in the log-file in "Polynomial Coefficients of Nonlinear Effects"?

Here, only effects that have higher degree of freedom than 1 are listed here. All other effects have a linear path coefficient in LPC section of log-file.

All graphs are produced for all combination regardless of significance?

Prevent graphs for non-significant paths by entering a lower significance threshold in cell "A8" in sheet "Options" of the Excel input file.

Is there a way to have the text output export as is into a text editor or Word? When it is copied and pasted into Word the columns etc output is messy.

In this manual your find:
"All coefficients can be found in the log-textfile named according to your definition of "Projectname" in cell B1 in the Excel input file: "projectname_log.txt"
The textfile is saved in the active path. Default is: ...\Matlab\work\

How does NEUSREL handles missing values?

NEUSREL uses the **20-Nearest-Nabour approach** to substitute missing values. This is superior to a simple mean imputation.

R2 for linear models are higher then with NEUSREL. Why is it?

Below you find the results for two dataset that showed higher R2 for linear models then NEUSREL. To prove what is happening we run the datasets with Bayesian Neural Networks (as applied in NEUSREL) and with standard "Frequentistic" Neural Networks (FNN). FNN are based in classical Multi Layer Perceptron and only poorly encompasses issues as overfitting. One might expect to have highest R2 with Neural Networks since they have most free parameters.

```
Dataset 1
BNN
R2
        1_Communic  2_Alignment 3_ITContrib 4_AvailAlte 5_ITCapacit 6_Success
         0.40           0           0           0           0        0.07
FNN
R2
        1_Communic  2_Alignment 3_ITContrib 4_AvailAlte 5_ITCapacit 6_Success
         0.54           0           0           0           0        0.55

R2 for linear structural model - similar PLS
        1_Communic  2_Alignment 3_ITContrib 4_AvailAlte 5_ITCapacit 6_Success
         0.39           0           0           0           0        0.25
```

```
Dataset2

BNN
R2
        1_Communic  2_Alignment 3_ITContrib 4_AvailAlte 5_ITCapacit 6_Success
         0.38           0           0           0           0        0.15
FNN
R2
        1_Communic  2_Alignment 3_ITContrib 4_AvailAlte 5_ITCapacit 6_Success
         0.44           0           0           0           0        0.25

R2 for linear structural model - similar PLS
        1_Communic  2_Alignment 3_ITContrib 4_AvailAlte 5_ITCapacit 6_Success
         0.38           0           0           0           0        0.12
```

FNN always "outperform" linear models, but this is most probably overfitting, which is in contrast prevented with BNN.

Methodical Introduction

A large body of research in leading marketing journals uses structural equation modeling (SEM) to estimate structural models with three or more latent variables simultaneously. The two most frequently applied SEM methods are covariance-based structural equation modeling (CVSEM), made popular through Jöreskog and Sörbom's (1999) LISREL program, and the component-based partial least squares (PLS) approach originated by Wold (1989). More than 180 articles published in the *Journal of Marketing*, *Journal of Marketing Research*, and *Journal of the Academy of Marketing Science* applied SEM between 1995 and 2005, with 89% of these publications using CVSEM and 11% PLS. Today, SEM can be considered a methodological paradigm in its own right within the marketing discipline—a dominant logic for defining and addressing complex research problems.

Although both CVSEM and PLS offer widely recognized, powerful methods for testing a specific model, the early stage of theory development in marketing often prevents researchers from excluding the possibility that alternative models might represent the true relations among model constructs more effectively (Rust and Schmittlein 1985). Comparing a proposed model with certain alternative models is certainly helpful (Bollen 1989; Cudeck and Browne 1983), but it does not rule out the existence of still other models that are equally or even more likely than the proposed one. Similarly, the widely available software for CVSEM (e.g., LISREL, AMOS, EQS) and PLS (e.g., PLS Graph, SmartPLS) remains limited to linear relations among constructs and does not test for interactions among model variables that have not been theoretically proposed by the researchers. Accordingly, alternative models with theoretically unproposed nonlinear paths and interactions systematically get overlooked by CVSEM and PLS methods.[1] This exclusion is important because both nonlinear effects (Agustin and Singh 2005;

[1] We use the term "theoretically unproposed" to acknowledge that nonlinear relations and interactions that are known in advance could be considered when testing a structural marketing model with CVSEM and PLS. We focus on those interactions and nonlinear relations that are *not* hypothesized by a researcher and, therefore, are not part of the theoretical model.

Oliva, Oliver, and MacMillan 1992) and interactions (Chandrashekaran et al. 1999; Kohli, Shervani, and Challagalla 1998; Nowlis and Simonson 1996; Rao, Qu, and Rueckert 1999) are common phenomena in marketing.

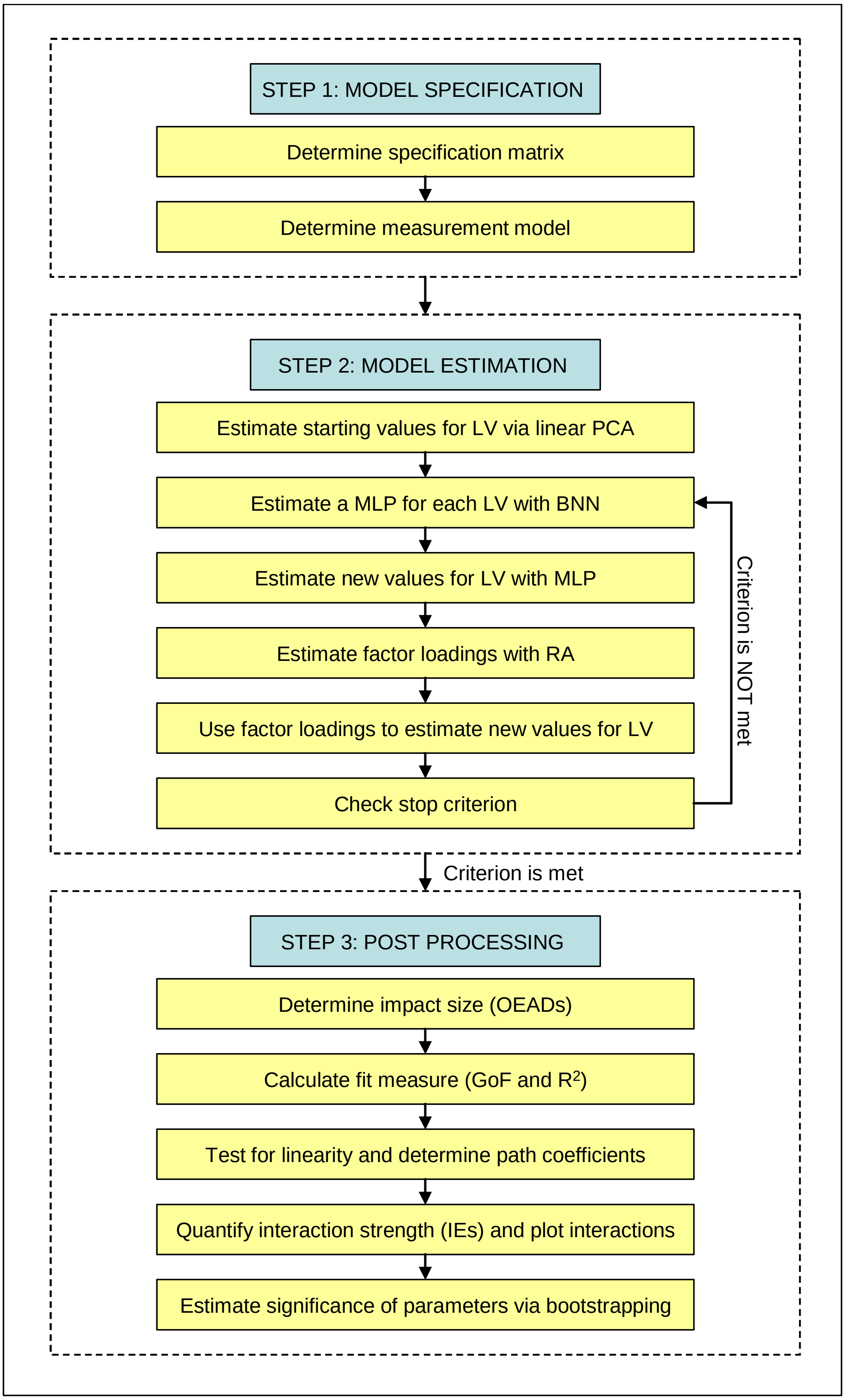
Overcoming these limitations of traditional SEM methods requires applying a more exploratory approach that is not limited to testing a small (and often arbitrarily chosen) number of structural models but rather provides insight into any possible relationship among the variables of a structural model. Ideally, such an exploratory approach takes into account the potential existence of theoretically unproposed nonlinear relations and interactions among model constructs, in addition to unhypothesized model paths. This manuscript introduces Universal Structure Modeling (USM) as a method that enables researchers to apply such an exploratory approach to SEM and thus helps them identify different kinds of “hidden” structures instead of testing a limited set of rival model structures. Specifically, the USM approach combines the iterative component-based approach of PLS with a neural network involving a multilayer perceptron architecture and points researchers toward theoretically unproposed (1) paths among model constructs, (2) nonlinear relations between model constructs, and (3) interactions among model constructs. As a result, USM enables researchers to improve their theoretical models and rule out the existence of superior alternatives.

Methodological Foundation

Universal Structure Modeling builds on the iterative PLS approach for testing structural models but substitutes its linear least squares regression element with a universal regression method, namely, a neural network.[2] Thus, USM solves the black box problem inherent to universal regression through its combined use of methods that measure the strength of model paths and procedures that quantify and visualize

[2] Universal regression methods, given sufficient parameterization, can model any kind of function (Rojas 1996). Examples of these regression algorithms include, in addition to neural networks, projection pursuit regression, Gaussian processes, and multivariate adaptive regression splines. We have chosen neural networks because of their high flexibility and appropriateness for marketing issues demonstrated in earlier work (e.g., West, Brockett, and Golden 1997).

nonlinear and interactive effects among model constructs. Whereas PLS and CVSEM both limit model estimation to *a priori* hypothesized paths, USM represents a more exploratory approach that also tests for hidden model structures, namely, theoretically unproposed paths, nonlinearity, and interaction effects. In Figure 1, we overview the different phases and steps of the USM parameter estimation process, each of which we discuss in detail.



STEP 1: MODEL SPECIFICATION
Determine specification matrix
Determine measurement model
STEP 2: MODEL ESTIMATION
Estimate starting values for LV via linear PCA
Estimate a MLP for each LV with BNN
Estimate new values for LV with MLP
Estimate factor loadings with RA
Use factor loadings to estimate new values for LV
Check stop criterion
Criterion is NOT met
Criterion is met
STEP 3: POST PROCESSING
Determine impact size (OEADs)
Calculate fit measure (GoF and R^2)
Test for linearity and determine path coefficients
Quantify interaction strength (IEs) and plot interactions
Estimate significance of parameters via bootstrapping

Step 1: Model Specification

As with CVSEM and PLS, a USM model consists of a structural (or inner) model that contains several latent variables and their interrelations, as well as a measurement (or outer) model that links each latent variable to a set of manifest measurement variables. The initial step of a USM analysis involves creating a structural model specification matrix *S* that indicates the relations among the latent variables of the structural model to be excluded from the estimation process. Generally, because USM represents an exploratory approach to structural model estimation, the model includes all possible relationships between model variables. Only those relationships that are *known* to be wrong (e.g., a path from phenomenon B to phenomenon A when A took place before B) should be excluded by assigning values of 0 in the model specification matrix. If cross-sectional data appear in the model, researchers must decide *a priori* about the direction of model paths and exclude reverse effects.

In USM, the endogenous latent variables *y* are defined as functions of one or more other latent variables *y* that can be exogenous or endogenous in the structural model. Formally, the estimator $\hat{y}^j$ of the endogenous latent variable y^j is defined as the output of a multilayer perceptron (MLP) architecture, as shown in Equation 1:[3]

(1) $$\hat{y}^j = f_{Act2}\left(\sum_{h=1}^{H} w_h \cdot f_{Act1}\left(\sum_{i=1}^{I} w_{ih} \cdot S_i^j \cdot y^i + b_{1h}\right) + b_2\right),$$

where f_{Act1} is the logistic sigmoid activation function $f_{Act1}(Term) = \frac{1}{1+e^{-Term}}$ of the hidden neural units, and f_{Act2} is the linear activation function of the output neural unit.

[3] Regarding parameterization, we started all calculations in this research with ten hidden units and one hidden unit layer. This is an arbitrary choice which we chose based on our experience with the method. Please note that the choice of the number of hidden units is not a crucial question in the Bayesian framework, as the ASP mechanism automatically decreases the number of hidden units to the required level. Consequentially, in our empirical studies reported in this paper, ten units led to literally the same results as 20 units, so we chose 10 units to save calculation time.

Specifically, f_{Act2} is a unity function required in MLP networks if the dependent variable is metric (Bishop 1995). In turn, H is the number of hidden neural units, I is the number of latent input variables y, w are the weights, and b are the bias weights. In addition, S_i^j is the *a priori* likelihood that a variable i influences another variable j.[4] S_i^j is set to 1 for all variables that affect j in the model specification matrix and 0 for other variables. The sigmoid activation function f_{Act1} is approximately linear for a certain range of values, namely, very small weight parameters w_{ih} in Equation 1 (Ripley 1996).

In USM, the measurement model defines a latent variable y^i (or, more precisely, its estimator $\hat{y}^i$) as a linear combination of its indicators:

(2) $$\hat{y}^i = \sum_{m=1}^{M_i} f_m \cdot x_m + f_0 ,$$

where x are the values of M_i measurement variables that determine $\hat{y}^i$, f_m are factor loadings, and f_0 is the constant term of the function. Although the USM approach in general allows measurement model relations to be nonlinear, the nonlinear representation of constructs by a set of items further would increase the method's complexity and impede comparisons of the USM structural model results with other methods. Accordingly, we limit nonlinear relations to the structural model in USM in this manuscript.

Step 2: Model Estimation

As with PLS, structural and measurement models get estimated simultaneously in USM through an iterative estimation process. For USM, the estimation process starts with values for the latent variables derived from linear principal component analysis

[4] Our discussion of USM focuses on metric output units, which is the traditional scale format for structural models. Although it is possible to run USM with nonmetric (i.e., dichotomous) output units, this application is beyond the scope of this manuscript.

(instead of arbitrary values, as in PLS, mainly to save time) and estimates the paths between the latent variables using a neural network involving the MLP architecture (Minsky and Papert 1988; Ripley 1996). We use the evidence framework for MLP architectures introduced by MacKay (1992) for parameter estimation, which ensures effective weights and input pruning to detect irrelevant paths and prevent overfitting. Specifically, USM estimates the neural network by minimizing the error function *E* for each endogenous variable *i* of the structural model, with *E* being the overall error of the respective variable's neural network:

(3) $$E_i = \beta \cdot \sum_{n=1}^{N} (\hat{y}^i_{t-1,n} - \hat{y}^i_{t,n})^2 + \sum_{h=1}^{H} \alpha_{t,h} \cdot \sum_{p=1}^{P} w^2_{ph} \cdot ,$$

where *n* is an index for the individual cases, *N* is the number of cases included in the estimation, and *p* is an index for the weights *w*. In addition, $\hat{y}^i_t$ is the conditional estimate of the latent variable *i* in the current estimation round *t*, as calculated from the structural model by the neural network, and $\hat{y}^i_{t-1}$ is the estimate of the previous iteration for this latent variable, derived from the measurement model. Finally, α_h and β are hyperparameters that limit the space of possible solutions (i.e., degrees of freedom) and prevent overfitting of the model. Practically, parameters that do not contribute substantially to explaining the dependent variable's variance are removed from the estimation process (MacKay 1995). The hyperparameters α_h and β get computed in every learning iteration, as described in Equations 4 and 5:

(4) $$\alpha_h = \frac{\gamma}{2\sum_{n=1}^{N} w^2_{nh}}$$, and

(5) $$\beta = \frac{N - \gamma}{2\sum_{n=1}^{N} (y^i_n - \hat{y}^i_n)^2},$$

where *N* is the number of cases and $\gamma = \sum_{p=1}^{P} \frac{\lambda_p}{\lambda_p + \alpha_{LI-1}}$. Finally, λ_p are the eigenvalues of the Hessian matrix of the error function (see equation 3), and α_{LI-1} is the hyperparameter α from the previous learning iteration (*LI*).

In contrast to weighted decay and ridge regression, the evidence framework is not based on heuristics but on a systematic statistical approach with an inherent logic (for a detailed discussion, see Bishop 1995, p. 385ff.). We use the RPROP algorithm, suggested by Riedmiller and Braun (1993), to minimize the overall error *E*. RPROP is a variation of the basic back-propagation algorithm that changes the network parameters in the direction in which the overall error *E* declines (also referred to as "negative gradient"). We employ a boosting (i.e., Committee-of-Networks) approach to generate stable results with 30 replications (Bishop 1995)[5] by averaging 30 estimates of $\hat{y}^i$.[6]

The output of the neural network results in improved scores for the latent variables. These new scores then provide input for calculating the weights of the measurement model, which we need to generate the next round of latent variable scores. The concrete procedure in this step differs between reflective and formative measurement scales (Fornell and Cha 1994; Jarvis, MacKenzie, and Podsakoff 2003). In the case of reflective items, the new latent construct estimates emerge from a series of bivariate regression analyses (with observed scores as dependent variables), whereas we use multivariate regression analysis for formative scales (with the latent construct score as dependent variable). In both cases, the weighted regression coefficients transform into new estimates for the latent variable $\hat{y}_t^i$, which substitute for the previous latent variable estimations $\hat{y}_{t-1}^i$. The process of iteratively calculating inner and outer model estimates continues until the differences between the latent variable

[5] A weighted mean of the output of all Committee-of-Network solutions performs well on validation data (Bishop 1995, p. 364). We employ boosting since backpropagating MLPs start with random weights and never stop with exactly the same solution, as they perform a search on an error surface with many local optima. This issue may be of theoretical, but hardly of practical importance.

[6] Please note that, conceptually, USM is not limited to the use of the MLP architecture; it also could run with alternative approaches such as Gaussian processes.

scores calculated by the inner model and those by the outer model are minimal. Specifically, we stop the estimation process when the divisor of the absolute change in the latent variable scores summed across all model constructs and the sum of latent variable scores falls below 1%.

The iteration process aims to minimize residual variance (instead of maximizing a theoretically derived function, as is the case with CVSEM), and the different kinds of residual variables to be minimized (for endogenous latent and measurement variables) are partitioned into estimable subsets. In other words, one part of the parameters is held fixed (and is assumed to be known), whereas the other part gets estimated. This step describes basically the same procedure as PLS, so we can assume the iterative process converges, though convergence has not been formally proven (Fornell and Cha 1994).

Step 3: Post-Processing

After determining the final estimates for the latent variables, the next step is to investigate the strength, significance, and shape of the relations among the latent constructs of the inner model. Therefore, we calculate variance explanation parameters, coefficients of determination, the model's goodness of fit, path coefficients (for linear relations), and interaction effects.

Overall Explained Absolute Deviation. Path coefficients (measures of the strength of the relation between two latent variables) can be calculated only when the relationship between two variables is linear (note that path coefficients describe the additive influence of one variable on another). Therefore, we require a more general criterion for the strength of construct interrelations. We draw on Zimmermann (1994) and introduce the Overall Explained Absolute Deviation (OEAD) as a measure of latent variable *i*'s share of variance which is explained by latent variable *j* in the structural model. We formally define OEAD in Equation 6:

(6) $$OEAD_j^i = \frac{\sum_{n=1}^{N} \left| \frac{\hat{y}_n^i - f_{n\,NN}^{\,i}(y^1,..,\bar{y}^j,...,y^I)}{\hat{y}_n^i - \bar{\hat{y}}_n^i} \right|}{N},$$

where $f^{i}_{NN}(y^1,..,\bar{y}^j,...,y^I)$ is the outcome of the neural network function (Equation 1) when the mean of y^j (termed $\bar{y}^j$) serves as an input variable. In other words, as the numerator of Equation 6 shows, we calculate for each case how the estimates of a latent variable *i* change if we exclude the latent variable *j* from the model. This is done by fixing the input of variable *j* to its mean value, or $\bar{y}^j$. We normalize the effect by dividing it by the number of cases. Because the effect size of a latent variable for any other variable does not depend on the direction of this effect, we use the absolute value (the direction must be taken from the graphic depiction of the respective effect). We calculate effect sizes for each case, and OEAD represents the mean of the case-specific values. Because we can calculate OEAD for all kinds of relationships in structural models, including linear ones, we are able to compare USM structural model results with results from CVSEM and PLS. If a relationship between two latent variables is linear, USM also reports path coefficients by isolating the additive portion of a relationship. We discuss this feature formally in the context of interaction effects.

Coefficient of determination and goodness of fit. In addition to OEAD as a measure of relationship strength between two latent variables, USM provides R^2 as the total variance of an endogenous latent variable explained by all other model constructs. Furthermore, because USM does not aim to fit empirical and theoretical covariance matrices and therefore does not allow for a formal test of the “appropriateness” of a proposed model, we draw on the goodness-of-fit (GoF) criterion suggested by Tenenhaus and colleagues (2005) to compare the overall model fit of USM with CVSEM and PLS:

(7) $$GoF = \sqrt{\left(\frac{1}{M}\sum_{i=1}^{I} M_i \cdot communality_i\right) \cdot \overline{R^2}},$$

where *I* is the number of latent constructs in the model, *M* is the total number of measurement variables in the model, and M_i is the number of measurement variables for the construct *i*. Communality refers to the regression coefficient between an item and its latent variable, and $\overline{R^2}$ is the mean explained variance of all endogenous latent variables of the structural model.

Nonlinear shapes, path coefficients, and interaction effects. To identify an additive (potentially nonlinear) effect of the latent variable *j* on y^i, we calculate an *a*-score for each individual case *n* (Plate 1998):[7]

(8) $$a^i_j = f^i{}_{NN}(y^1,...y^j,...,y^n) - f^i{}_{NN}(y^1,...\bar{y}^j,...,y^I),$$

where a^i_j is the change in y^i caused by the additive effect of y^j, f_{NN} again is the neural network function (see Equation 1), and y^1 to y^n are the latent input variables of the structural model. The second term on the right side of Equation 8 represents the network output (i.e., $\hat{y}^i$) when variable *j* is absent (we use the mean value over all cases, $\bar{y}^j$, to fix *j* at an arbitrary value). The difference between the first and second terms on the right side represents the change in y^i that y^j provides in an additive manner.

We test the linearity of this additive effect of y^j on y^i for each relationship by estimating a series of polynomial regressions of y^j on a^i_j (Hastie and Tibshirani 1990). When doing that, we increase the number of parameters of the polynomial regression stepwise, an approach referred to as a "growing algorithm" (Bishop 1995, p. 353). If only linear effects are considered (i.e., one degree of freedom), polynomial regression is equivalent to linear regression. We assume that a relationship between two latent constructs is linear if a regression model with two degrees of freedom (i.e., quadratic model) shows lower prediction performance for validation data than does the linear model. This comparison relies on a jack-knifing cross-validation procedure (Bishop 1995). If we find a linear relationship, we take the standardized regression parameter *β* from the polynomial regression with one degree of freedom and interpret it as a path coefficient, similar to PLS and CVSEM.[8]

---

[7] As it is not possible to interpret the weights from neural network estimations directly (e.g., Kumar, Rao, and Soni 1995, p. 262), additional techniques have to be used.

[8] Because USM uses standardized raw data for its calculations, the USM path coefficients are, by definition, standardized parameters ranging from +1 to –1.

To measure the interaction effect of two independent latent variables y^j and y^k on y^i, we calculate a *z*-score for each individual case *n* (Plate 1998):

(9) $$z^i_{jk} = f^i{}_{NN}(y^1,...y^j,...,y^k,...,y^I) - f^i{}_{NN}(y^1,...\bar{y}^j,...,\bar{y}^k,...,y^I),$$

where *z* is the change in y^i for an individual case caused by the additive and the interactive effect of y^j and y^k, and $\bar{y}^j$ and $\bar{y}^k$ represent the mean values of y^j and y^k, respectively. The idea behind this equation is that the difference in latent variable *i* when both latent variables *j* and *k* simultaneously are fixed to the mean value represents the interactive effect (Plate 1998). The strength of the interactive effect IE^i_{jk} then can be calculated according to Equation 10:

(10) $$IE^i_{jk} = \frac{\sum_{n=1}^{N} \left| \frac{\hat{z}^i_{jk} - \hat{a}_j - \hat{a}_k}{\hat{y}^i - \overline{\hat{y}^i}} \right|}{N},$$

where $\hat{a}$ are the additive scores of a polynomial regression of *y* on *a* (as introduced in equation 8), and $\hat{z}$ is the outcome of a universal regression with the two latent variables *j* and *k* as regressors on z^i_{jk}. In other words, the interaction effect IE^i_{jk} is expressed as the portion of variable *i*'s explained variance that can be attributed to the interaction between y^j and y^k, with IE^i_{jk} = 1 meaning that the explained variance of *i* is caused fully by this interaction. Because IE only provides a measure of interaction strength, not the kind of interaction effect at work, we visualize the interaction effect by creating a three-dimensional scatter plot, in which y^j and y^k plot on the *x*-axis and the *y*-axis, respectively, and z^i_{jk} on the *z*-axis. The data points are complemented by the surface derived from the $\hat{z}^i_{jk}$ estimates. We limit our discussion to two-way interactions in this manuscript, but in principle, USM can also handle interactions among more than two variables. The main challenge for such multivariate interactions would be the higher-dimensional graphical representation that is needed to interpret the interaction effect (Soukup and Davidson 2002).

Significance of parameters. Finally, as with PLS and in contrast to CVSEM, USM does not pose any distributional assumptions on the data, which prevents us from testing the significance of its parameters against any kind of statistical distribution. We therefore test the model parameters' statistical significance through a bootstrapping routine (Mooney and Duval 1993). Specifically, we conduct such tests for all OEADs and IEs, as well as for the path coefficients and the factor loadings of the measurement model.

REFERENCES

Agustin, C., J. Singh. 2005. Curvilinear effects of consumer loyalty determinants in relational exchanges. J. Marketing Res. 42(February) 96-108.

Bagozzi, R. P., Y. Yi. 1994. Advanced Topics in Structural Equation Models. R. P. Bagozzi, ed. Advanced Methods of Marketing Research. Cambridge, MA: Blackwell, 1-51.

Baron, R. M., D. A. Kenny. 1986. The Moderator-Mediator Variable Distinction in Social Psychological Research: Conceptual, Strategic, and Statistical Considerations. J. Personality and Social Psychology. 51(6) 1173-1182.

Bentler, P. M., C.-P. Chou. 1987. Practical Issues in Structural Modeling. Sociological Methods and Research 16(1) 78-117.

Bishop, C. M. 1995. Neural Networks for Pattern Recognition. Oxford, UK: Oxford University Press.

Bollen, K. A. 1989. Structural Equations with Latent Variables. New York: Wiley.

Chandrashekaran, M., R. Mehta, R. Chandrashekaran, R. Grewal. 1999. Market Motives, Distinctive Capabilities, and Domestic Inertia: A Hybrid Model of Innovation Generation. J Marketing Res. 36(February) 95-112.

Chin, W. W. 2001. PLS-Graph User's Guide, Version 3.0. Houston, TX: Soft Modeling Inc.

Chin, W. W. 2004. Partial Least Squares & PLS-Graph: Multi-Group Analysis with PLS. http://disc-nt.cba.uh.edu/chin/plsfaq.htm. [July 28, 2006].

Cooil, B., R. S. Winer, D. L. Rados. 1987. Cross-Validation for Prediction. J Marketing Res. 24(August) 271-279.

Cudeck, R., M. W. Browne. 1983. Cross-Validation of Covariance Structures. Multivariate Behavioral Research. 18(April) 147-167.

De Wulf, K., G. Odekerken-Schröder, D. Iacobucci. 2001. Investments in Consumer Relationships: A Cross-Country and Cross-Industry Exploration. J Marketing. 65(October) 33-50.

Fornell, C. 1987. A Second Generation in Multivariate Analysis: Classification of Methods and Implications for Marketing Research. Michael Houston, ed. Review of Marketing. Chicago, IL: American Marketing Association, 407-450.

Fornell, C. 1989. The Blending of Theoretical and Empirical Knowledge in Structural Equations with Unobservables. Herman Wold, ed. Theoretical Empirism, New York: Paragon.

Fornell, C. 1992. A National Customer Satisfaction Barometer: The Swedish Experience. J Marketing. 56(January) 6-21.

Fornell, C., F. L. Bookstein. 1982. Two Structural Equation Models: LISREL and PLS Applied to Consumer Exit-Voice Theory. J Marketing Res. 19(November) 440-452.

Fornell, C., J. Cha. 1994. Partial Least Squares. Richard P. Bagozzi, ed. Advanced Methods of Marketing Research. Cambridge, MA: Blackwell, 52-78.

Fornell, C., M. D. Johnson, E. W. Anderson, J. Cha, B. E. Bryant. 1996. The American Customer Satisfaction Index: Nature, Purpose, and Findings. J Marketing. 60(October) 7-18.

Gentle, J. E. 2003. Random Number Generation and Monte Carlo Methods. 2d ed. New York: Springer.

Hastie, T. J., R. Tibshirani. 1990. Generalized Additive Models. Boca Raton, FL: Chapman & Hall/CRC.

Hennig-Thurau, T., A. Klee. 1997. The Impact of Customer Satisfaction and Relationship Quality on Customer Retention: A Critical Reassessment and Model Development. Psychology & Marketing. 14(8) 737–64.

Jarvis, C. B., S. B. MacKenzie, P. M. Podsakoff. 2003. A Critical Review of Construct Indicators and Measurement Model Misspecification in Marketing and Consumer Research. J Consumer Res., 30(September) 199-218.

Jöreskog, K. G. 1971. Simultaneous Factor Analysis in Several Populations. Psychometrika. 36 409-426.

Jöreskog, K. G., D. Sörbom. 1993. LISREL 8: Structural Equation Modeling with the SIMPLIS Command Language. Lincolnwood, IL: Scientific Software International.

Jöreskog, K. G., D. Sörbom. 1999., LISREL 8: User's Reference Guide. Lincolnwood, IL: SSI Inc.

Journal of Marketing. 2006. Most-Cited JM Articles. http://www.marketingpower.com/content31973.php [April 21, 2006].

Kenny, D. A., C. M. Judd. 1984. Estimating the Nonlinear and Interactive Effects of Latent Variables. Psychological Bulletin. 96(1) 201-210

Kohli, A. K., T. A. Shervani, G. N. Challagalla. 1998. Learning and Performance Orientation of Salespeople: The Role of Supervisors. J Marketing Res. 35(May) 263-274.

Kumar, A., V. R. Rao, H. Soni. 1995. An Empirical Comparison of Neural Network and Logistic Regression Models. Mark. Letters. 6(4) 251-263

MacCallum, R. C. 1986. Specification Searches in Covariance Structure Modeling. Psychological Bulletin. 100(1) 130-143.

MacKay, D. J. C. 1992. A Practical Bayesian Framework for Backpropagation Networks. Neural Computation. 4(May) 448-472.

MacKay, D. J. C. 1995. Bayesian Non-Linear Modelling for the 1993 Energy Prediction Competition. Glenn R. Heidbreder, ed. Maximum Entropy and Bayesian Methods. Santa Barbara, CA: Kluwer, 221-234.

Minsky, M. L., S. A. Papert. 1988. Perceptrons. Boston, MA: MIT Press.

Mooney, C. Z., R. D. Duval. 1993. Bootstrapping. A Nonparametric Approach to Statistical Inference. Newbury Park, CA: Sage.

Nowlis, S. M., I. Simonson. 1996. The Effect of New Product Features on Brand Choice. J Marketing Res. 33(February) 36-46.

Oliva, T. A., R. L. Oliver, I. C. MacMillan. 1992. A Catastrophe Model for Developing Service Satisfaction Strategies. J Marketing. 56(July) 83-96.

Plate, T. 1998. Controlling the Hyper-Parameter Search in MacKay's Bayesian Neural Network Framework. G. Orr, K.-R. Müller, R. Caruana, eds. Neural Networks: Tricks of the Trade. Berlin: Springer, 93-112.

Rao, A. R, L. Qu, R. W. Ruekert. 1999. Signaling Unobservable Product Quality Through a Brand Ally. J Marketing Res. 36(May), 258-268.

Riedmiller, M., H. Braun. 1993. A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm. Proceedings of the International Conference on Neural Networks. San Francisco, CA: IEEE, 586–591.

Ringle, C. M., S. Wende, A. Will. 2005. SmartPLS. http://www.smartpls.de, Hamburg: University of Hamburg.

Ripley, B. D. 1996. Pattern Recognition and Neural Networks. Cambridge, UK: Cambridge University Press.

Rojas, R. 1996. Neural Networks: A Systematic Introduction. Berlin: Springer.

Rust, R. T., D. C. Schmittlein. 1985. A Bayesian Cross-Validated Likelihood Method for Comparing Alternative Specifications of Quantitative Models. Marketing Sci. 4(Winter) 20-40.

Soukup, T., I. Davidson. 2002. Visual Data Mining: Techniques and Tools for Data Visualization and Mining. New York: Wiley.

Tenenhaus, M., V. E. Vinzi, Y.-M. Chatelin, C. Lauro. 2005. PLS Path Modeling. Computational Statistics & Data Analysis. 48 159-205.

West, P. M., P. L. Brockett, L. L. Golden. 1997. A Comparative Analysis of Neural

Networks and Statistical Methods for Predicting Consumer Choice. Marketing Sci. 16(4) 201-206.

Wold, H. 1989. Introduction to the Second Generation of Multivariate Analysis. H. Wold, ed. Theoretical Empirism. New York: Paragon House, VII-XL.

Zeithaml, V., L. L. Berry, A. Parasuraman. 1996. The Behavioral Consequences of Service Quality. J Marketing, 60(April) 31-46.

Zimmermann, H. G. 1994. Neuronale Netze als Entscheidungskalkül. H. Rehkugler, H. G. Zimmermann, eds. Neuronale Netze in der Ökonomie. Munich: Vahlen, 1-87.