

# **NN-Tool 2024**

Stand Oktober 2023  
Autor: Frank Bärmann

# Inhalt

<b>1. Einführung</b>	<b>4</b>
<b>2. Installation</b>	<b>5</b>
<b>3. Datenaufbereitung</b>	<b>7</b>
<b>4. Netztraining</b>	<b>15</b>
<b>5. Klassifikatoren und Lernparameter</b>	<b>25</b>
<b>6. Optimierung der Modellgüte – Crossvalidation und Input-Optimierungslauf</b>	<b>33</b>
<b>7. Messdatenüberwachung – Data Reconciliation und Predictive Maintenance</b>	<b>41</b>
<b>8. Zeitreihen</b>	<b>52</b>
<b>9. Anwendungsmodul Mischpult</b>	<b>58</b>
<b>10. Komplexe Optimierungsprobleme / Rezepturoptimierungen</b>	<b>62</b>
<b>Anhang 1: Versuchsplanung</b>	<b>73</b>
<b>Anhang 2: Automatische Dokumentation</b>	<b>76</b>
<b>Anhang 3: Batchdatei</b>	<b>77</b>
<b>Anhang 4: Spezielle Funktionen zur Datenanalyse und Vorverarbeitung</b>	<b>78</b>
<b>Anhang 5: Ausreisserliste</b>	<b>87</b>
<b>Anhang 6: Validationsets</b>	<b>88</b>
<b>Anhang 7: Nichtlineare Skalierungstransformationen</b>	<b>89</b>
<b>Anhang 8: Integration von Vorwissen / Input-Output-Zuordnung</b>	<b>93</b>
<b>Anhang 9: Registerkarte Lernparameter / Crossvalidation</b>	<b>94</b>
<b>Anhang 10: Dynamische Simulationen</b>	<b>96</b>

<b>Anhang 11: Online-Anwendungen</b>	<b>99</b>
<b>Anhang 12: Closed Loop Simulation</b>	<b>105</b>
<b>Anhang 13: Datenfile filtern</b>	<b>108</b>
<b>Anhang 14: Einsatz Neuronaler Netze unter Excel / NN-Tool Excel Add-In</b>	<b>110</b>
<b>Anhang 15: Hauptkomponentenanalyse / PCATool</b>	<b>116</b>
<b>Anhang 16: Multi-Netz-Anwendungen - MNA</b>	<b>117</b>
<b>Anhang 17: UserModules</b>	<b>124</b>
<b>Anhang 18: NN-Tool Netzstruktur</b>	<b>132</b>
<b>Anhang 19: NN-Tool Produktübersicht</b>	<b>134</b>
<b>Anhang 20: Aufbau der 'COB'-Datei:</b>	<b>136</b>
<b>Anhang 21: GNU LESSER GENERAL PUBLIC LICENSE</b>	<b>140</b>
<b>Anhang 22: Installations- und Laufzeitfehler</b>	<b>149</b>

## 1. Einführung

NN-Tool besteht aus einem Simulator zur Erstellung neuronaler Netzmodelle und aus einem Anwendungsmodul zur Auswertung der erstellten Modelle (Prognose bzw. Optimierung). Nach Bereitstellung der Daten können die Modelle in kürzester Zeit erstellt werden, da NN-Tool die bei neuronalen Netzen (NN) erforderlichen Skalierungstransformationen sowie die Ermittlung der vollständigen Datensätze selbständig vornimmt. Die entstehenden NNe sind vom Standardtyp (Feedforwardnetz mit einem Hiddenlayer), der sich in allen praktisch relevanten Aufgabenstellungen als optimal erwiesen hat <sup>1</sup>. Nach der Modellerstellung hat der Anwender die Möglichkeit auf der Basis des erstellten Modells Prognose- und Optimierungsrechnungen durchzuführen.

Die Anwendung besteht aus den drei Hauptschritten:

- Datenaufbereitung
- Netztraining
- Netzauswertung zu Prognose- und Optimierungszwecken

Im vorliegenden Handbuch wird der Einsatz von NN-Tool an mehreren Beispielen diskutiert. Die entsprechenden Beispieldaten liegen in den im Verzeichnis „**Dokumente\NN-Tool**“ abgelegten Dateien TEST.xls, Class6.xls, DataRec\_Demo.xlsx sowie ZEIT.xls als Excel-Files vor. Dieses Verzeichnis wird beim ersten Start von NN-Tool erzeugt.

---

<sup>1</sup> Diese experimentellen Ergebnisse wurden in jüngster Zeit durch die Arbeit von Barron auch theoretisch untermauert. Als Lernalgorithmus wird jedoch nicht Backpropagation verwendet, sondern aus Performancegründen wird ein Algorithmus eingesetzt, der in den folgenden Veröffentlichungen näher beschrieben ist:

Bärnmann, Biegler-König: On a Class of Efficient Learning Algorithms for Neural Networks, Neural Networks **5** (1992) pp. 139-144

Martino, Fanelli, Protasi: Computational Experiences of New Direct Methods for the On-line Training of MLP-Networks with Binary Outputs ICANN 93 - Sorrento, pp. 627-630



## 2. Installation

Bitte beachten Sie unbedingt die Hinweise auf dem beigegeführten Beiblatt.

Legen Sie die CD ein und starten Sie das Programm "Setup.exe". Befolgen Sie die Anweisungen des Installationsprogramms.

**Besonders wichtig:** Überprüfen Sie dann **die Zahleneinstellungen Ihres Betriebssystems**: Gehen Sie in die Systemsteuerung. Wählen Sie in der Systemsteuerung die Regions- und Sprachoptionen (bzw. „Datums-, Uhrzeit- oder Zahlenformat ändern“) aus.  
**Ablauf unter Windows 10:** Windows-Startbutton / Einstellungen / Zeit und Sprache/ Region / Weitere Datums-, Uhrzeit- und Ländereinstellungen / Datums- Uhrzeit oder Zahlenformat ändern / Weitere Einstellungen:

The screenshot shows the 'Format anpassen' dialog box with the 'Zahlen' tab selected. The 'Beispiel' section shows 'Positiv: 123.456.789,00' and 'Negativ: -123.456.789,00'. The settings are as follows:

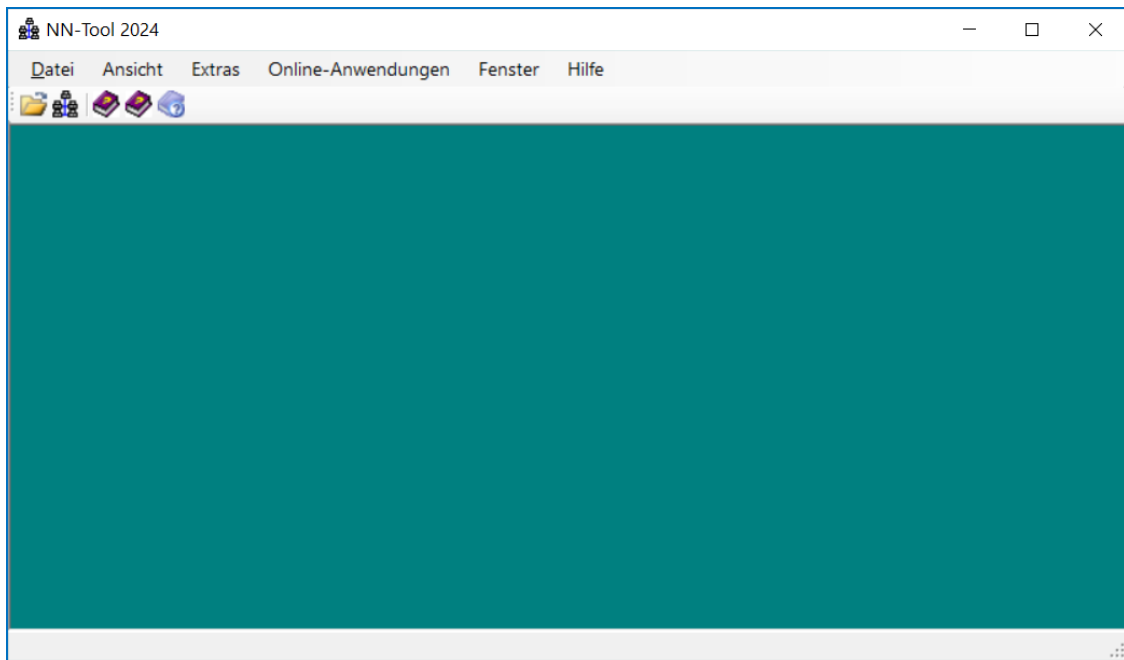
Setting	Value
Dezimaltrennzeichen:	,
Anzahl der Dezimalstellen:	2
Symbol für Zifferngruppierung:	.
Zifferngruppierung:	123.456.789
Negatives Vorzeichen:	-
Format für negative Zahlen:	-1,1
Führende Nullen anzeigen:	0,7
Listentrennzeichen:	;
Maßsystem:	Metrisch
Standardziffer:	0123456789
Ziffernersatz:	Nie

At the bottom, there is a 'Zurücksetzen' button and three buttons: 'OK', 'Abbrechen', and 'Übernehmen'.

**Wichtig:** als Dezimaltrennzeichen ist das zum jeweiligen Gebietsschema gehörende Dezimaltrennzeichen einzustellen. Für das Gebietsschema **Deutsch (Deutschland)** ist das **Komma** einzustellen, für die Gebietsschemata Englisch (Großbritannien) und Englisch

(USA) gilt der Dezimalpunkt. Das Symbol für Zifferngruppierung ist ebenfalls nach dem Gebietsschema einzustellen. Diese Einstellungen sind die **Windows Standardeinstellungen**. **Ohne diese Einstellungen kann und wird NN-Tool nicht korrekt arbeiten.** Dies trifft im Übrigen auf eine ganze Reihe von Anwendungen zu. Wenn Sie in einer deutschen Windows-Version aus irgendeinem Grund gezwungen sind mit dem Dezimalpunkt zu arbeiten, sollten Sie **immer das gesamte Gebietsschema z.B. auf Englisch (USA) umstellen!**

Starten Sie nun NN-Tool. Es erscheint das folgende Fenster (NN-Tool-Hauptfenster):



**Hinweis: Unter Extras/Optionen („Extras/Options“) lässt sich die Sprache der Benutzeroberfläche einstellen!**

NN-Tool zeigt immer nur die zum aktuellen Bearbeitungsstand sinnvollen Optionen als Menüpunkte an. Die genaue Abarbeitung wird nun an der Beispieldatei „Test.xls“ erläutert.

### 3. Datenaufbereitung

NN-Tool setzt voraus, dass sich die Daten in einer **Datei mit der Extension .pat** (für Pattern) befinden. Jeder Datensatz muss in einer Zeile stehen, verschiedene Parameter sind durch Semikola zu trennen. Der erste Datensatz besteht aus den Namen der Parameter. **Als Dezimaltrennzeichen ist das zu den jeweiligen Regionaleinstellungen gehörende Dezimaltrennzeichen (d.h. bei deutschen Regionaleinstellungen das Komma „,”) zu verwenden.**

**Grundsätzlich stellt NN-Tool zwei verschiedene Möglichkeiten bereit um Daten zu importieren bzw. eine korrekte Datendatei (Pat-Datei) zu erzeugen:**

1. **Direkter Zugriff auf eine Excel-Mappe:** statt einer Pat-Datei lädt NN-Tool zunächst eine Excel-Anwendung, der User wählt ggf. ein Tabellenblatt aus (bei mehreren Blättern in der Excel-Mappe) und NN-Tool erzeugt und importiert die Datendatei.
2. **Laden einer .csv-Datei:** Das sogenannte csv-Format („character separated values“) ist ein weit verbreitetes Datenformat für Tabellen. csv-Dateien können aus einer Vielzahl von Anwendungen erzeugt werden (z.B. Excel, Access etc.).

Die beiden Methoden werden im Folgenden vorgestellt. Dabei wird vorausgesetzt, dass die Daten zunächst in einem Excel-Tabellenblatt vorliegen. Dabei können die Excel-Mechanismen zum Datenimport verwendet werden. Aus dieser Tabelle wird nun die .pat-Datei erzeugt.

#### 1. **Direkter Zugriff auf eine Excel-Mappe**

Die einfachste Methode eine korrekte Datendatei zu erzeugen besteht darin, die Daten zunächst in ein Excel-Spreadsheet zu laden. Dabei können die Excel-Mechanismen zum Datenimport verwendet werden. Anschließend kann aus NN-Tool heraus auf die Excel-Mappe zugegriffen und die entsprechende NN-Tool Datendatei (.pat-Datei) generiert werden.

**Starten Sie nun Excel** und öffnen Sie die Datei **Test.xls** im Verzeichnis „**Dokumente\NN-Tool**“. Die Datei sieht unter Excel wie folgt aus:

Die Datei **Test.xls** beschreibt eine Rezeptur-Eigenschaftsaufgabe. Es sind an 500 Rezepturen, bestehend aus je 6 Komponenten (Komp1 bis Komp6), je zwei physikalische Eigenschaften gemessen worden (Eig1 und Eig2). Zusätzlich ist als erster Parameter eine laufende Versuchsnummer vergeben worden.

Anhand dieses Beispiels wird im Folgenden die Handhabung von NN-Tool erläutert.

Eine Anwendung (d.h. ein neuronales Netz mit allem was dazugehört: Daten, Gewichte etc.) wird von NN-Tool unter einem Anwendungsnamen angesprochen. **Sämtliche zugeordneten Dateien erhalten den Anwendungsnamen als Dateinamen.** Sie unterscheiden sich also nur durch ihre Extension. Zu Beginn einer Anwendung muss nur die Datendatei **<Anwendungsname>.pat** vorhanden sein, bzw. sie muss z.B. aus Excel oder einer anderen Anwendung (Access, Matlab, ...) erzeugt werden.

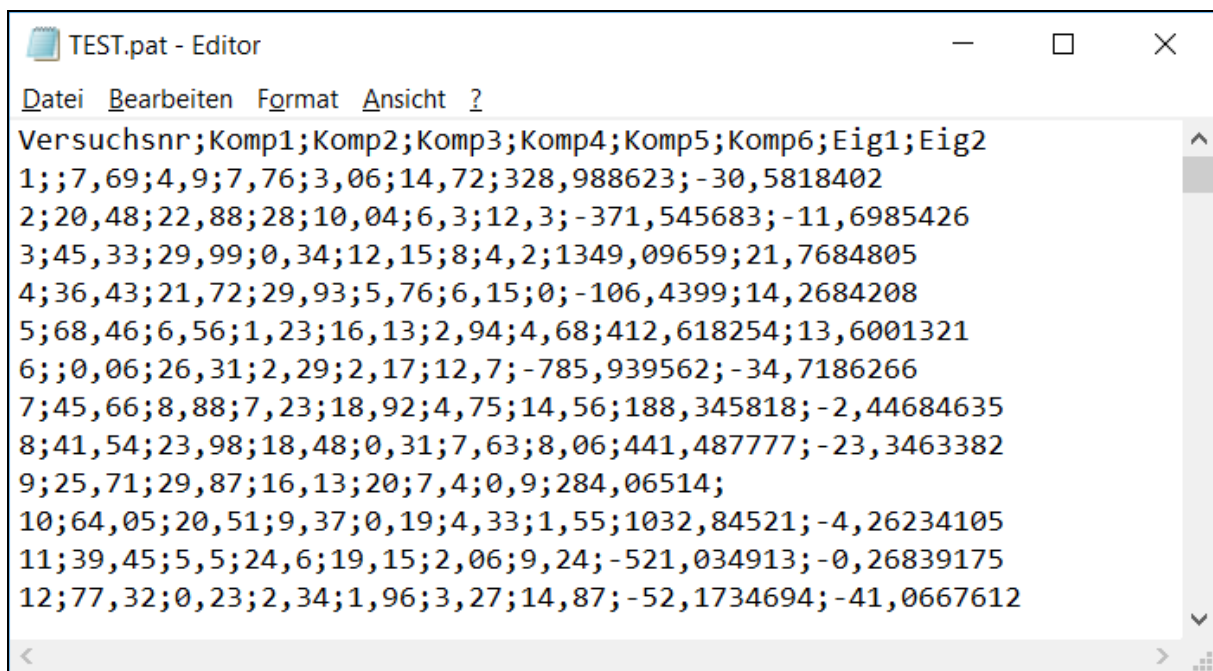
Wie Sie in der Spalte Komp1 sehen, sind die Daten teilweise unvollständig. NN-Tool kann unvollständige Datensätze geeignet verarbeiten, vorausgesetzt, dass die entsprechenden Werte **wirklich leer** sind und keine Blanks (Leerstellen) oder Tabulatoren etc. enthalten. Blanks werden wie alle übrigen nichtnumerischen Werte von NN-Tool als sogenannte **Klassifikatoren** interpretiert und dürfen daher in numerischen Parametern nicht auftreten (evtl. mit einem Editor entfernen). Klassifikatoren sind Parameter, die genau einen Wert (Instanz) aus einer endlichen Auswahl annehmen, z.B. der Klassifikator "Oberflächenfehler" mit den Instanzen "keine", "Kratzer", "Riefen", "Dellen" etc. Auf die Verarbeitung von Klassifikatoren wird im zweiten Beispiel eingegangen.

Schließen Sie nun die Excel-Mappe, kehren Sie zu NN-Tool zurück (bzw. starten Sie NN-Tool) und gehen Sie bitte wie folgt vor:

- Wählen Sie im NN-Tool Hauptfenster **Datei/Datenfile öffnen** oder drücken Sie den entsprechenden Button (der Button hat den Erläuterungstext "Datenfile öffnen").

- Setzen Sie im „Öffnen“-Dialog als Dateityp „**Excel-Dateien (\*.xls, \*.xlsx)**“.
- Wählen Sie die Excel-Mappe mit den Daten. Beachten Sie, dass die Daten in Excel den gleichen Aufbau haben wie im oben angegebenen Beispiel, d.h. in der ersten Zeile der Excel-Tabelle stehen die Parameternamen, unmittelbar gefolgt von den Datensätzen (Datensätze zeilenweise angeordnet, keine Leerzeilen). Falls die Excel-Mappe nur ein Datenblatt enthält, wird daraus die .pat-Datei erzeugt und sofort in NN-Tool importiert. Als **Name wird der Tabellenblattname** (und nicht etwa der Excel-Dateiname!) mit der Erweiterung .pat für Patterndatei festgelegt.
- Falls die Excel-Mappe mehrere Tabellenblätter enthält, folgt ein entsprechender Hinweis und es erscheint ein Auswahlfenster, in welchem Sie das richtige Blatt auswählen können. Dazu werden Ihnen die jeweils ersten 20 Zeilen der verschiedenen Tabellenblätter als zusätzliche Information angezeigt.
- Wählen Sie das gewünschte Blatt und drücken Sie „**Tabelle laden**“. Aus der Tabelle (natürlich mit sämtlichen Datensätzen, nicht nur den ersten 20) wird ein .pat-File erzeugt und importiert. Die Daten werden im Fenster Datenblatt angezeigt.

Da die NN-Tool Datendatei auch direkt erstellt werden kann (z.B. aus Datenbanken), wird im Folgenden kurz auf den Aufbau der Datei eingegangen. Die zuvor erzeugte Datendatei Test.pat hat den folgenden Aufbau (entspricht .csv):



```

TEST.pat - Editor
Datei Bearbeiten Format Ansicht ?
Versuchsnr;Komp1;Komp2;Komp3;Komp4;Komp5;Komp6;Eig1;Eig2
1;;7,69;4,9;7,76;3,06;14,72;328,988623;-30,5818402
2;20,48;22,88;28;10,04;6,3;12,3;-371,545683;-11,6985426
3;45,33;29,99;0,34;12,15;8;4,2;1349,09659;21,7684805
4;36,43;21,72;29,93;5,76;6,15;0;-106,4399;14,2684208
5;68,46;6,56;1,23;16,13;2,94;4,68;412,618254;13,6001321
6;;0,06;26,31;2,29;2,17;12,7;-785,939562;-34,7186266
7;45,66;8,88;7,23;18,92;4,75;14,56;188,345818;-2,44684635
8;41,54;23,98;18,48;0,31;7,63;8,06;441,487777;-23,3463382
9;25,71;29,87;16,13;20;7,4;0,9;284,06514;
10;64,05;20,51;9,37;0,19;4,33;1,55;1032,84521;-4,26234105
11;39,45;5,5;24,6;19,15;2,06;9,24;-521,034913;-0,26839175
12;77,32;0,23;2,34;1,96;3,27;14,87;-52,1734694;-41,0667612

```

Jede Zeile in diesem Beispiel enthält 9 Parameter, die durch 8 Semikola getrennt sind. In der ersten Zeile stehen (**immer!**) die Parameternamen. Für NN-Tool zulässige Parameternamen dürfen aus Buchstaben, Zahlen, runden Klammern ( ), dem Bindestrich - und dem Unterstrich \_ bestehen. Andere Satz- bzw. Sonderzeichen können bei der Nutzung der NN-Tool Anwendungsmodule zu Problemen führen. **Auf keinen Fall sollten eckige Klammern [], Punkt . und Semikolon ; verwendet werden.** Auf die Überschriftenzeile muss unmittelbar die erste Datenzeile folgen. In den Datenzeilen 1 und 6 ist der zweite Parameter nicht mit einem Wert belegt, in der 9. Zeile fehlt der letzte Parameter. NN-Tool kann derartige unvollständige Datensätze verarbeiten, solange nur sichergestellt ist, dass in jeder Zeile die gleiche Anzahl von Semikola steht. Bei maximal n Parametern müssen **in jeder Zeile** (n-1)

Semikola stehen. Zusätzliche Blanks (Leerstellen) dürfen in numerischen Parametern nicht auftreten (evtl. mit einem Editor entfernen), da sie als Klassifikatoren interpretiert werden. Bitte untersuchen Sie bei direkt erstellten Eingabedateien vor der Anwendung von NN-Tool Ihre Daten mit Hilfe eines Editors. **Fehlerhafte Dateiformate sind der bei weitem häufigste Anwendungsfehler.**

Wie zuvor erwähnt, erzeugt NN-Tool aus der Excel-Mappe die .pat-Datei, öffnet diese und bestimmt zunächst die Maximalzahl der in den Datensätzen auftretenden Parameter sowie die Zahl der vorhandenen Datensätze. Diese Werte werden dem Anwender in der Statusleiste am unteren Bildrand angezeigt. Der Applikationsname wird in der NN-Tool-Überschriftenleiste eingeblendet. Zusätzlich werden die geladenen Datensätze im Fenster **Datenblatt** angezeigt:

Satznr.	Versuchsnr.	Komp1	Komp2	Komp3	Komp4	Komp5	Komp6	Eig1	Eig2
1	1		7,69	4,9	7,76	3,06	14,72	328,988623	-30,5818402
2	2	20,48	22,88	28	10,04	6,3	12,3	-371,545683	-11,6985426
3	3	45,33	29,99	0,34	12,15	8	4,2	1349,09659	21,7684805
4	4	36,43	21,72	29,93	5,76	6,15	0	-106,4399	14,2684208
5	5	68,46	6,56	1,23	16,13	2,94	4,68	412,618254	13,6001321
6	6		0,06	26,31	2,29	2,17	12,7	-785,939562	-34,7186266
7	7	45,66	8,88	7,23	18,92	4,75	14,56	188,345818	-2,44684635
8	8	41,54	23,98	18,48	0,31	7,63	8,06	441,487777	-23,3463382
9	9	25,71	29,87	16,13	20	7,4	0,9	284,06514	
10	10	64,05	20,51	9,37	0,19	4,33	1,55	1032,84521	-4,26234105
11	11	39,45	5,5	24,6	19,15	2,06	9,24	-521,034913	-0,26839175
12	12	77,32	0,23	2,34	1,96	3,27	14,87	-52,1734694	-41,0667612
13	13	25,54	10,11	29,51	16,54	6,53	11,77	-627,065878	6,94362805
14	14	58,1	25	0,02	5,28	7,98	3,62	1451,91529	4,07128422
15	15	22,71	29,64	29,02	12,67	5,9	0,06	-197,553799	30,6034643

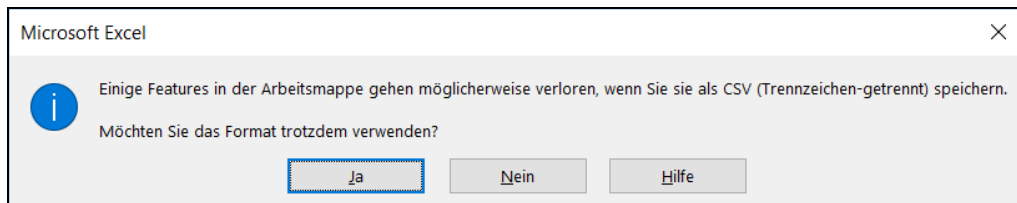
In unserem Beispiel (Anwendung Test mit der Datendatei Test.pat im Verzeichnis „Dokumente\NN-Tool“) besteht ein Datensatz aus 9 Parametern (auch wenn nicht in jedem Satz alle vollständig sind). Insgesamt sind 500 Datensätze vorhanden (die Zeile mit den Parameternamen wird nicht mitgezählt). **Dieses Fenster sollte zu einer Kontrolle des korrekten Datenimports genutzt werden!**

**Falls der Import in der angegebenen Weise nicht möglich war, wird nun zunächst die andere Option vorgestellt:**

## 2. Laden einer .csv-Datei

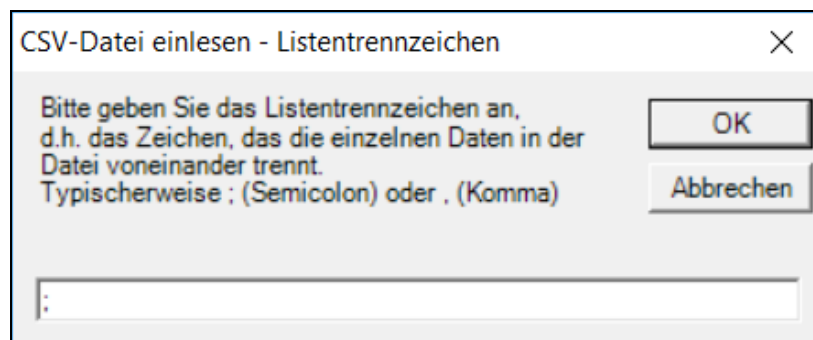
Falls ein direkter Zugriff von NN-Tool auf Excel nicht möglich ist oder die Daten gar nicht in Excel (sondern in einer anderen Anwendung) vorliegen, können Sie die Daten wie folgt importieren (hier für Excel beschrieben, in anderen Anwendungen ist die Vorgehensweise analog):

- Falls die zuvor genannte Option scheitert (d.h. ein direkter Zugriff auf Excel ist nicht möglich), muss die Konvertierung der Excel-Daten mit den Excel-Methoden selbst vorgenommen werden. Dazu muss die Excel-Tabelle mittels „Speichern unter“ mit dem Dateityp „**csv**“ („**character separated values**“) abgespeichert werden. Schließen Sie anschließend Excel. Falls dabei ein Hinweis der Art kommt:



dann drücken Sie „Ja“, d.h. Datei im CSV-Format speichern.

- Es wird nun eine Datei vom .csv-Format erzeugt, in der die Daten mittels eines Listentrennzeichens voneinander getrennt werden. Dieses Listentrennzeichen hängt (leider!) von den Windows-Einstellungen ab, genauer von den sogenannten Regionaleinstellungen. Bei deutschen Regionaleinstellungen handelt es sich um das Semikolon, bei englischen um das Komma. Eine .csv-Datendatei kann von NN-Tool im Menüpunkt „**Datenfile öffnen**“ eingelesen werden. Dazu ist dann beim Öffnen der Dateityp auf „**CSV (Trennzeichen getrennt)**“ umzustellen um die Datei auszuwählen. Anschließend wird nach dem oben genannten Listentrennzeichen gefragt.



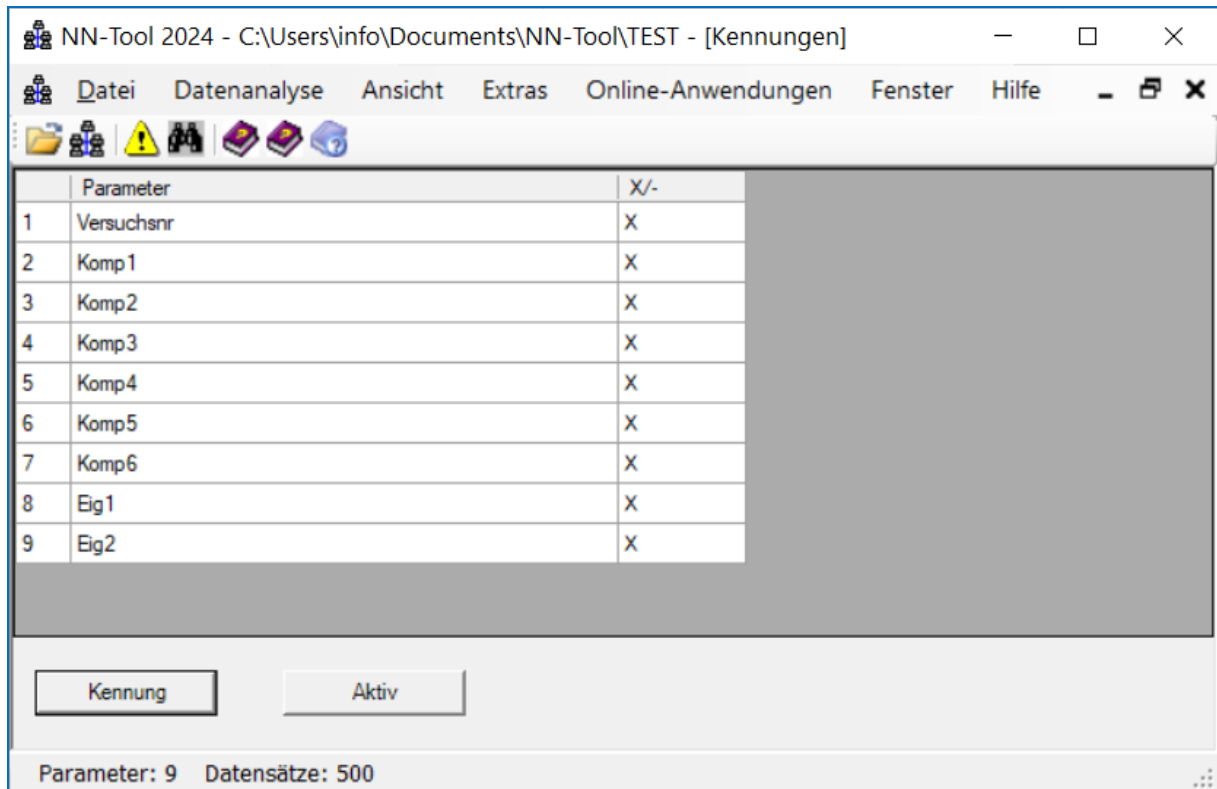
- Falls Ihre .csv-Datei das Semikolon verwendet (Standardeinstellung bei deutschen Windowsversionen), können sie sofort auf „OK“ drücken. Andernfalls müssen Sie das Komma (oder ggf. ein noch anderes Listentrennzeichen) eingeben. Es darf nur genau ein Zeichen eingegeben werden. Damit importiert NN-Tool die Datendatei, erzeugt



zusätzlich eine .pat-Datei, öffnet das Fenster Datenblatt und sie können wie im Folgenden beschrieben fortfahren.

**Der nächste Schritt** besteht in der sukzessiven Abarbeitung der Menüpunkte im Menu **Datenanalyse**. Der Menüpunkt **Datenblatt** dient der Kontrolle und ggf. Bearbeitung der geladenen Datensätze und ist bereits aktiviert worden.

Schließen Sie das Datenblattfenster, und führen Sie als nächsten Schritt den Menüpunkt **Datenanalyse\Kennungen deaktivieren** aus (bzw. den entsprechenden Button):



Unter **Kennungen** werden Parameter verstanden, die zusätzliche Informationen über einen Datensatz bereitstellen, aber nicht in das zu erstellende NN-Modell eingehen sollen. Dazu zählen typischerweise **Versuchsnummern, Datum und Uhrzeit, etc.** Diese Parameter sollten aus Performancegründen direkt abgeschaltet werden. Dies gilt insbesondere, wenn es sich nicht um rein numerische Werte handelt (z.B. um Datum und Uhrzeit), da diese Parameter von NN-Tool als Klassifikatoren interpretiert werden (die Analyse von Klassifikatoren ist numerisch wesentlich aufwendiger als die Analyse kontinuierlicher numerischer Werte). **Hinweis: Das Abschalten von Kennungen ist nicht zwingend erforderlich. Parameter können auch später noch abgeschaltet werden. Bei großen Datensätzen und nichtnumerischen Größen ist es jedoch zweckmäßig.**

Klicken Sie das Feld Versuchsnr bzw. das dahinterstehende X an und drücken Sie die Taste **Kennung**. Damit wird der Parameter als inaktive Kennung gesetzt. Mit der Taste **Aktiv** kann diese Einstellung ggf. zurückgenommen werden (soll hier nicht durchgeführt werden).



Schließen Sie das Fenster und führen Sie nun die Datenanalyse aus (Menu **Datenanalyse**, Menüpunkt **Analyse** bzw. entsprechender Button). In diesem Schritt bestimmt NN-Tool wesentliche Kenngrößen jedes Parameters und zeigt diese im folgenden Fenster an:

Parameter	Min	Max	Mittel	Stdabw	Median	vollständig	Transform.	E/A	A/P	verfügbar
1 Versuchsnr	Kennung					500	Lin	K	P	
2 Komp1	14,7000	84,4900	47,4236	17,3077	50,6900	498	Lin	E	A	
3 Komp2	0,0000	29,9900	15,0309	10,6174	15,2800	500	Lin	E	A	
4 Komp3	1,0000e-02	30,0000	14,9764	10,6314	14,8150	500	Lin	E	A	
5 Komp4	0,0000	20,0000	9,9997	7,0585	10,0750	500	Lin	E	A	
6 Komp5	2,0000	8,0000	5,0048	2,1235	5,0100	500	Lin	E	A	
7 Komp6	0,0000	15,0000	7,5172	5,3091	7,4000	500	Lin	E	A	
8 Eig1	-872,1356	1506,8319	151,7104	626,3127	92,0545	500	Lin	E	A	
9 Eig2	-44,8463	51,6725	-0,8354	24,2734	-2,4893	499	Lin	E	A	

Buttons: Eingang, Aktiv, Ausgang, Passiv, Transformationen (Lin, Log, Sig, Auto-Eingänge, Auto-Ausgänge, Alle Lin), nach Anzahl abschalten, Klassifikator manuell, Histogramm, Datenverlauf, User-Modul definieren.

Parameter: 9 Datensätze: 500

Nach der ersten Durchführung der Analyse kann das Fenster auch über den Menüpunkt **Parameter setzen** direkt aufgerufen werden (dabei wird die Analyse selbst nicht erneut ausgeführt). Für jeden Parameter (der keine Kennung ist) werden der minimale, maximale Wert sowie Mittelwert, Standardabweichung und Median (d.h. der Wert bei dem 50% der Daten kleiner und 50% größer sind) und die Anzahl des Auftretens (Punkt "vollständig") angezeigt. Anhand der Daten des Fensters sollte zunächst überprüft werden, ob die Datensätze korrekt eingelesen wurden (Plausibilitätsprüfung). Falls Parameter als Klassifikatoren identifiziert wurden, steht im Feld Min das Schlüsselwort Class. Dieser Fall wird im nächsten Beispiel ausführlich behandelt.

Die Spalte Transformation gibt die verwendete Skalierung an und wendet sich an den erfahrenen Anwender (siehe Anhang).

Die Spalte E/A gibt an, ob ein Parameter Eingangs- oder Ausgangsgröße bzw. Kennung ist. Entsprechend legt die Spalte A/P fest, ob ein Parameter aktiv oder passiv ist. Nur aktive Parameter nehmen an der Netzmodellierung teil. Ein Datensatz muss auch nur bezüglich der aktiven Eingangsparameter vollständig sein. Nach Markieren eines oder mehrerer Parameter mit der Maus können diese Einstellungen mit den entsprechenden Tasten umgesetzt werden. In unserem Beispiel werden die beiden letzten Parameter als Ausgangsparameter festgelegt. Anschließend sollte das Fenster wie folgt aussehen:

	Parameter	Min	Max	Mittel	Stdabw	Median	vollständig	Transform.	E/A	A/P	verfügbar
1	Versuchsnr	Kennung					500	Lin	K	P	
2	Komp1	14,7000	84,4900	47,4236	17,3077	50,6900	498	Lin	E	A	
3	Komp2	0,0000	29,9900	15,0309	10,6174	15,2800	500	Lin	E	A	
4	Komp3	1,0000e-02	30,0000	14,9764	10,6314	14,8150	500	Lin	E	A	
5	Komp4	0,0000	20,0000	9,9997	7,0585	10,0750	500	Lin	E	A	
6	Komp5	2,0000	8,0000	5,0048	2,1235	5,0100	500	Lin	E	A	
7	Komp6	0,0000	15,0000	7,5172	5,3091	7,4000	500	Lin	E	A	
8	Eig1	-872,1356	1506,8319	151,7104	626,3127	92,0545	500	Lin	A	A	498
9	Eig2	-44,8463	51,6725	-0,8354	24,2734	-2,4893	499	Lin	A	A	497

Parameter: 9   Datensätze: 500   Eingänge/-Knoten: 6/6   Ausgänge/-Knoten: 2/2

In der letzten Spalte werden nun für die Ausgangsparameter die jeweils **verfügbaren Datensätze** angezeigt. Ein Datensatz ist für einen Ausgang verfügbar, wenn er in sämtlichen aktiven Eingangsparameter und im jeweiligen Ausgangsparameter vollständig ist. Er muss also insbesondere nicht in allen Ausgängen vollständig (d.h. mit einem Wert belegt) sein. Für eine gute Modellierung kann die Zahl der verfügbaren Datensätze gar nicht hoch genug sein. Insbesondere sollte die Zahl deutlich größer sein als die Zahl der Eingangsparameter.

Die Statusleiste zeigt nun zusätzlich die Zahl der aktiven Eingangs- und Ausgangsparameter bzw. die entsprechenden Knoten an. Diese Zahlen unterscheiden sich nur bei Verwendung von Klassifikatoren (siehe nächstes Beispiel). **Da für einen kontinuierlichen (numerischen) Eingangs- oder Ausgangsparameter genau ein Knoten benötigt wird, sollten bei ausschließlicher Verwendung kontinuierlicher Größen beide Zahlen übereinstimmen (Plausibilitätsprüfung).** Dies trifft im oberen Beispiel auch zu.

Schließen Sie nun dieses Fenster. Damit ist diese erste, einfache Datenanalyse abgeschlossen. Auf die Vielzahl weiterer Optionen unter Transformation oder den weiteren Tabellenblättern wird später eingegangen.

Nach der Datenanalyse besteht nun die Möglichkeit, das neuronale Netz zu erstellen bzw. zu trainieren.

## 4. Netztraining

Starten Sie nun den Lernvorgang (das „Training“) des Netzes über den Menüpunkt „**Netztraining/Lernvorgang starten**“. NN-Tool bestimmt für jeden der Ausgangsparameter (hier 2) automatisch die jeweils optimale Netzstruktur und bindet die beiden Einzelnetze anschließend zu einem Gesamtnetz zusammen.

**Datenaufteilung:** Dazu wird für jeden Ausgang zunächst die Menge der jeweils vollständigen Datensätze bestimmt und in einen Lernset und einen Testset aufgeteilt. In den von uns verwendeten Standardeinstellungen wird jeder 5. vollständige Datensatz dem Testset zugeordnet, die übrigen 80% der vollständigen Datensätze gehören dem Lernset an. **Bitte beachten Sie:** Bei unvollständigen Datensätzen sind diese Zuordnungen i.A. für die verschiedenen Ausgangsparameter unterschiedlich.

**Lernvorgang:** Anschließend beginnt für jeden Ausgangsparameter getrennt die Bestimmung der optimalen Netzstruktur. Dazu werden für jeden Ausgang einzeln eine ganze Reihe von Netzstrukturen (d.h. unterschiedlich vielen inneren Knoten) und Lernschritten („Iterationen“) durchgetestet.

Mit der **Festlegung der inneren Knoten im sogenannten Hiddenlayer** eines neuronalen Netzes wird die Modellierungsfähigkeit des Netzes im Wesentlichen bestimmt. Je mehr innere Knoten vorliegen, desto flexibler ist die Netzstruktur und desto komplexere Zusammenhänge können gelernt werden. Bei einer zu großen Zahl innerer Knoten besteht jedoch die Gefahr des „Overfittings“, d.h. das Netz lernt die Daten auswendig. Insbesondere werden dann auch die Messfehler mitgelernt. Es ist also ganz entscheidend die richtige Anzahl innerer Knoten festzulegen. Diese Zahl ist jedoch i.A. nicht von vorne herein bekannt. Die richtige Zahl innerer Knoten wird durch die Komplexität der zu lernenden Funktionen bestimmt. Sind beispielsweise sämtliche Funktionen linear in den Eingangsgrößen, würde ein innerer Knoten pro Ausgangsgröße genügen. Sind die Zusammenhänge zwar nichtlinear, aber immer noch monoton, kommt man i.a. ebenfalls mit wenigen inneren Knoten aus. Typischerweise steht eine solche Information normalerweise nicht zur Verfügung. In der Regel wird die Komplexität bei realistischen Beispielen aus der Verfahrenstechnik oder der chemischen Forschung vom Anwender überschätzt. Die auftretenden Funktionen sind häufig monoton oder zumindest monomodal (d.h. weisen höchstens ein Maximum auf).

In ähnlicher Weise wie die Zahl der inneren Knoten wirkt sich auch die **Zahl der Lernschritte („Iterationen“)** aus. Mehr Lernschritte verbessern i.A. das entstehende Modell; ab einem bestimmten Punkt, der selbst wieder von der Zahl der inneren Knoten abhängt, setzt jedoch Overfitting ein.

Das Problem besteht also darin für jeden Ausgangsparameter die optimale Kombination von inneren Knoten und Iterationen zu finden. Dazu wendet NN-Tool ein **Rasterverfahren** an. Die Netze werden für verschiedene Netzstrukturen und Lernschritte jeweils auf dem Lernset trainiert und ihre Modellgüte wird dann auf dem davon unabhängigen Testset bestimmt. Es beginnt für den ersten Ausgang mit einem Netz mit nur einem inneren Knoten und zehn Lernschritten, dann kommt die erste Bewertung dieser Struktur auf dem Testset, anschließend weitere Tests nach 20,30,40 bis zu 100 Lernschritten. Danach dieselbe Prozedur für ein Netz mit 2 inneren Knoten, dann 4 und 8 innere Knoten. Die Zwischenergebnisse werden im folgenden Fenster angezeigt:

Ausgangsknoten: 1				
Iterationen	innere Knoten	Testdatensätze	Korrelationskoeffizient	
1	4	90	99.9716	
1	4	100	99.9719	
1	8	10	99.9809	
1	8	20	99.9836	
1	8	30	99.9852	
1	8	40	99.9879	
1	8	50	99.9902	
1	8	60	99.9915	
1	8	70	99.9925	
1	8	80	99.9931	
1	8	90	99.9936	
1	8	100	99.9940	
Ausgangsknoten: 2				
Lerndatensätze		398		
Testdatensätze		99		Lernvorgang starten
Iterationen	innere Knoten	Testdatensätze	Korrelationskoeffizient	
2	1	10	86.9421	
2	1	20	86.4787	
2	1	30	86.2648	
2	1	40	86.1018	
2	1	50	85.9802	
2	1	60	85.8926	
2	1	70	85.8325	
2	1	80	85.7928	
2	1	90	85.7672	

Hier wurde gerade das Training für den ersten Ausgang (1. Spalte) beendet. Zuletzt wurden für den ersten Ausgang die Netze mit 4 und anschließend mit 8 inneren Knoten (2. Spalte) und 10 bis 100 Lernschritten („Iterationen“) getestet. Dabei ergaben sich Korrelationskoeffizienten von über 99.99 %. Der Begriff wird später noch genauer erläutert. Ein Korrelationskoeff. von 100% entspricht einem perfekten Modell ohne Fehler, bei 0% konnte gar nichts gelernt werden.

Anschließend ging es mit dem Ausgangsknoten 2 (entspricht hier dem Ausgangsparameter 2, vgl. auch folgendes Beispiel) weiter. Für diesen gibt es 497 vollständige Datensätze von denen 398 zum Lernen und 99 zum Testen verwendet werden. Der Prozess startet wieder mit einem Netz mit nur einem inneren Knoten und 10 Lernschritten etc.

Wenn der gesamte Prozess durchgelaufen ist, werden die für die jeweiligen Ausgangsknoten (entspricht hier den Ausgangsparametern) optimalen Einzelnetze zum Gesamtnetz zusammengefügt.

Anschließend führt NN-Tool eine Prognose für jeden einzelnen Datensatz des Lernsets und des Testsets aus und ermittelt dann den Korrelationskoeffizienten („**Korrelation**“) sowie den mittleren Fehlerbetrag (zunächst in % der Skalenvariation = „**Rel. Fehler in %**“, anschließend in der Dimension der Originalgröße = „**Abs. Fehler**“) zwischen den berechneten und den vorgegebenen Ausgangswerten für jeden Ausgangsparameter. Darüberhinaus werden noch die Koeffizienten der Ausgleichsgeraden durch die Punkte („**a0- und a1-Koeffizient**“) sowie die Anzahl der Datensätze des betreffenden Sets dargestellt. Die Bedeutung der beiden Koeffizienten wird weiter unten genauer erklärt. Der Korrelationskoeffizient sowie der a1-Koeffizient sind häufig geeignetere Maße für die Güte einer Prognose als der mittlere Fehler, da in diese Koeffizienten zusätzlich die Verteilung der Datensätze eingeht. Ein perfektes Modell entspricht einem Korrelationskoeffizienten von 1, einem a0-Koeffizienten von 0 sowie einem a1-Koeffizienten von 1. Die Ergebnisse werden im folgenden Fenster („Modellgüte“) dargestellt:

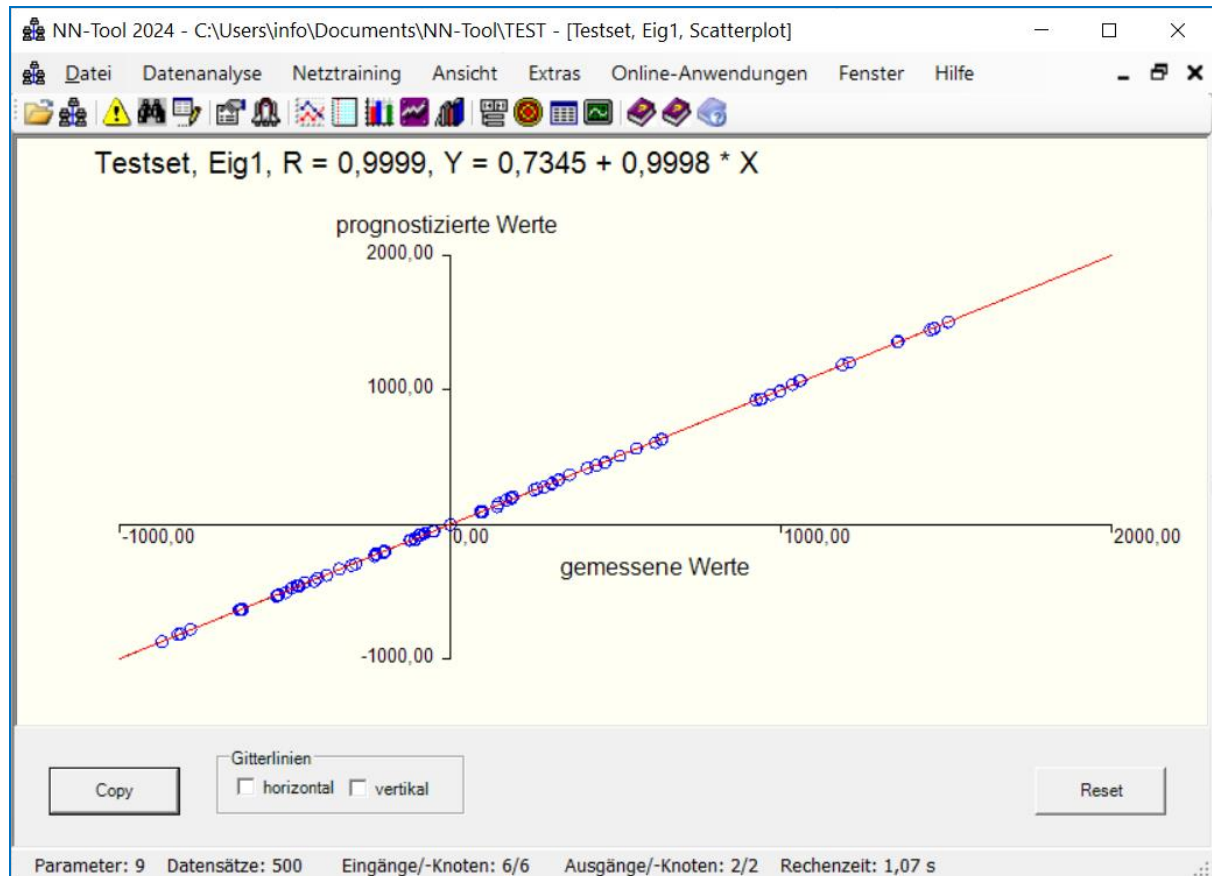
Nr.	Ausgangsparameter	Korrelation	Rel. Fehler in %	Abs. Fehler	a0-Koeffizient	a1-Koeffizient	Anzahl
1	Eig1	0,9999	0,2455	5,8409	0,7345	0,9998	99
2	Eig2	1,0000	0,1454	0,1403	0,0124	1,0001	99

Parameter: 9 Datensätze: 500 Eingänge/-Knoten: 6/6 Ausgänge/-Knoten: 2/2 Rechenzeit: 1,07 s

Die Farbumterlegung des betreffenden Parameters charakterisiert die Güte der Modellierung: „grün“ = gute Modellierung, „gelb“ = mäßige, „rot“ = schlechte Modellgüte

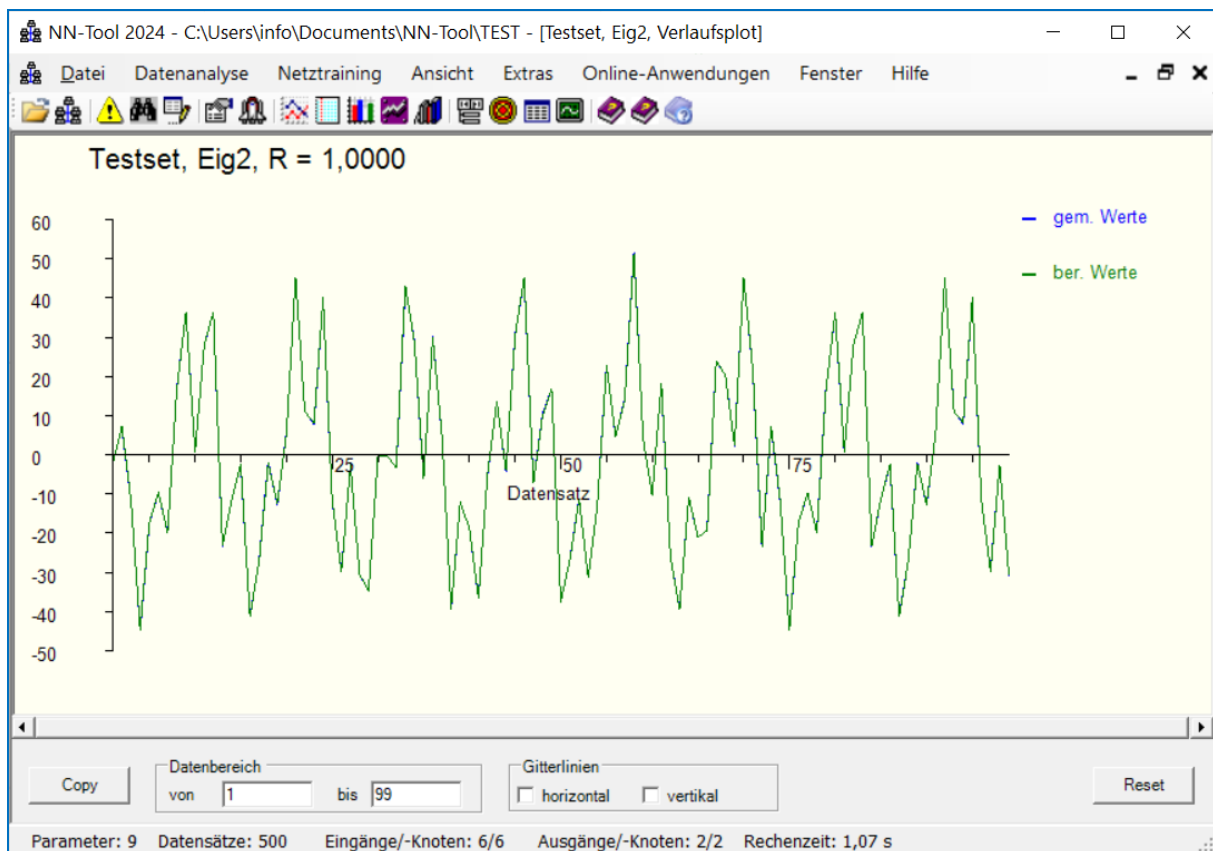
Das Fenster kann jederzeit auch manuell aufgerufen werden. Dazu wählt man im Menu **Netztraining** den Menüpunkt **Prognose Lern- und Testset** aus (oder entsprechenden Button drücken).

Durch Anklicken kann nun einer der Ausgangsparameter angewählt werden und dazu die folgende Grafik (**Scatterplot**) angezeigt werden (Scatterplot kann auch durch Doppelclick auf den Parameternamen angefordert werden):



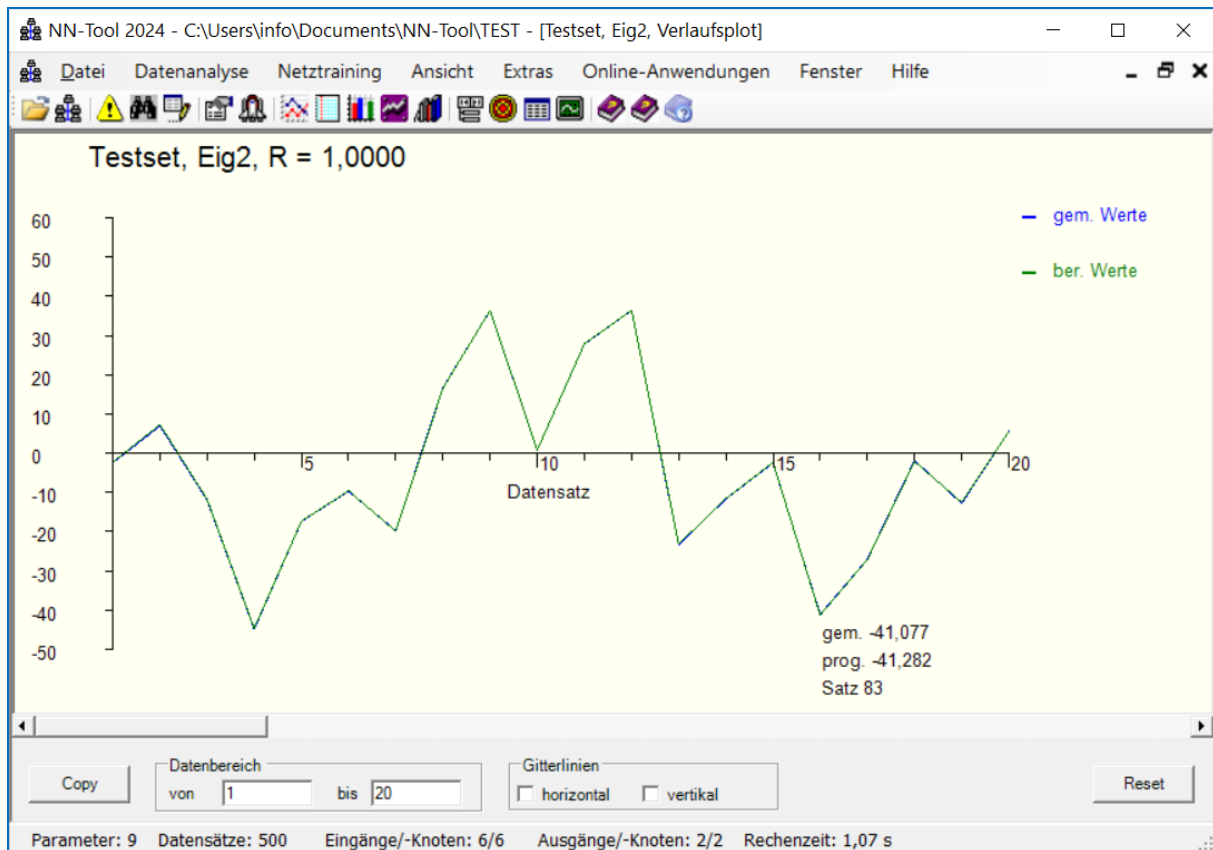
Für jeden Datensatz des Testsets ist hier der berechnete (prognostizierte) gegen den vorgegebenen (gemessenen) Wert aufgetragen. Darüber hinaus ist die "Winkelhalbierende" (blaue Linie, liegt hier unter der roten Linie) sowie die Ausgleichsgerade (rote Linie) durch die Punktwolke eingetragen. **a0-Koeffizient und a1-Koeffizient definieren die Ausgleichsgerade.** Der a1-Koeffizient muss für ein gutes Modell in der Nähe von 1 liegen. Dagegen muss der a0-Koeffizient nur klein in Bezug auf die typischen Werte eines Ausgangsparameters sein. Insofern ist hier ein Wert von 0,73 auch völlig unkritisch. Idealerweise sollten alle Punkte auf der „Winkelhalbierenden“ (d.h. auf der Geraden  $y = x$ ) liegen und die Ausgleichsgerade sollte mit der Winkelhalbierenden übereinstimmen. Auf diese Weise ermöglicht diese Darstellung eine schnelle visuelle Kontrolle der Güte des Netzes, die bei mehr als 2 Eingangsgrößen durch keine andere graphische Darstellung vergleichbar leicht möglich ist. Durch Anklicken der Taste **Copy** kann die Grafik in die Zwischenablage eingestellt werden und von da aus durch Kopieren beispielsweise in ein Word-Dokument übernommen werden.

Neben dem Scatterplot steht zur Beurteilung der Modellgenauigkeit auch noch der **Verlaufsplot** zur Verfügung (kann auch durch Drücken der rechten Maustaste aktiviert werden).



Bei dieser Darstellung werden für die einzelnen Datensätze des Testsets (bzw. Lernsets) der prognostizierte und der gemessene Wert über der Nummer des Datensatzes aufgetragen. In beiden Grafikarten können durch Klicken in die Grafik mittels der linken Maustaste zusätzliche Informationen abgerufen werden. Mit der rechten Maustaste kann in die Grafik gezoomt werden, mit der mittleren (bzw. der Taste „Reset“) zurückgesetzt werden. Im Folgenden ist nur der Bereich der ersten 20 Testdatensätze dargestellt, zusätzlich wurde ein Punkt der Grafik angeklickt (beachten Sie: der 16. Datensatz des Testsets ist der 83. Satz der Gesamtdatenmenge):





Durch Drücken der Taste **Excel** im Fenster „Modellgüte“ werden die gemessenen und berechneten Werte der Ausgangsparameter in eine Excel-Tabelle übertragen. Dadurch lassen sich beliebig formatierte Darstellungen anfertigen.

Entsprechende Informationen lassen sich auch für den **Lernset** abrufen (Fenster Modellgüte, Blatt Korrelationen Lernset). Das Blatt **Netzstruktur** zeigt die jeweils ausgewählte optimale Netzstruktur an. Mittels des Buttons „**Details**“ kann der gesamte Lernverlauf angezeigt werden.

Ist noch keine hinreichende Modellgüte erzielt worden, kann der Lernvorgang mittels einer Vielzahl von Parametereinstellungen optimiert werden. Darauf wird aber erst im nächsten Beispiel eingegangen. **Man beachte jedoch, dass der prinzipiell erreichbare Korrelationskoeffizient durch die Messfehler in den Lern- und Testdaten begrenzt** ist, da ja bei Fehler- oder Korrelationskoeffizientenberechnungen stets mit den vorhandenen Daten verglichen wird.

Im vorliegenden Beispiel liegt ein praktisch perfektes Modell vor. Dies liegt natürlich an den spezifischen Beispieldaten und trifft keineswegs immer zu (vgl. nächstes Beispiel).

Falls ein hinreichend genaues Modell erreicht worden ist, kann dieses Modell nun zu Analysen des Einflusses der Eingangs- auf die Ausgangsgrößen herangezogen werden. Führen Sie den Menüpunkt **Einflussanalyse** im Menu **Netztraining** aus (oder drücken Sie den entsprechenden Button). Das Programm bestimmt nun den mittleren Einfluß und den mittleren absoluten Einfluss einer Eingangsgröße  $x_j$  auf eine Ausgangsgröße  $y_i$ , in dem an

allen vorliegenden Datenpunkten die entsprechenden partiellen Ableitungen des skalierten NN-Modells (bzw. die Beträge davon) berechnet und dann gemittelt werden:

$$E_{i,j,\text{mittel}} = \frac{1}{p} \sum_{i=1}^p \frac{\partial y_i}{\partial x_j} \quad , \quad E_{i,j,\text{mittelabs}} = \frac{1}{p} \sum_{i=1}^p \left| \frac{\partial y_i}{\partial x_j} \right|$$

Der mittlere Einfluß ist immer kleiner gleich dem mittleren absoluten Einfluß. Bei monotonen Abhängigkeiten gilt:

$$|E_{i,j,\text{mittel}}| = E_{i,j,\text{mittelabs}}$$

Ist dagegen der Betrag des mittleren Einflusses echt kleiner, dann wirkt sich die Größe  $x_j$  mal positiv und mal negativ auf die Größe  $y_j$  aus (abhängig von den Werten der anderen Größen).

Eingangsparameter	Mittl. Einfluss	Mittl. abs. Einfluss	Min.	Max.
Komp2	0,5414	0,5414	0,0000	29,9900
Komp3	-0,4229	0,4229	1,0000e-02	30,0000
Komp1	0,3354	0,3384	14,7000	84,4900
Komp6	-0,0257	0,0606	0,0000	15,0000
Komp4	-0,0184	0,0868	0,0000	20,0000
Komp5	-9,0841e-03	0,0259	2,0000	8,0000

Ausgangsparameter  
Eig1  
Eig2

Grafik Excel

Sortierung  
☒ mittl. Einflüsse 
 ☐ mittl. abs. Einflüsse

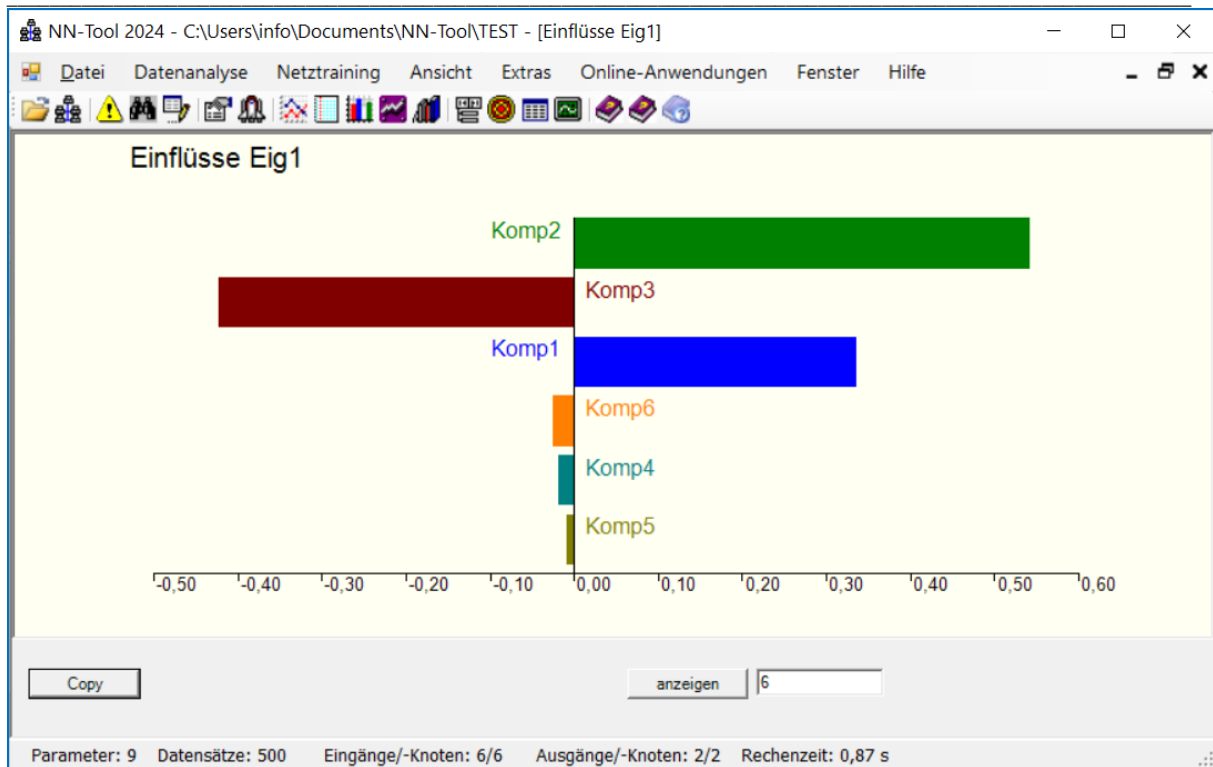
Setpoint  
☒ Alle 
 ☐ Satz 
 1

Parameter: 9 Datensätze: 500 Eingänge/-Knoten: 6/6 Ausgänge/-Knoten: 2/2 Rechenzeit: 0,87 s

Für den unten angewählten Ausgangsparameter werden in der oberen Tabelle für sämtliche Eingänge die mittleren und die mittleren absoluten Einflüsse aufgelistet. Die Angaben min und max beziehen sich auf den Variationsbereich der Parameter und stellen lediglich zusätzliche Informationen bereit. Dem Beispiel ist zu entnehmen, daß sich die beiden ersten gelisteten Parameter „Komp2“ und „Komp3“ monoton auf die Ausgangsgröße „Eig1“ auswirken, die anderen Parameter jedoch nicht. Die beiden wichtigsten Einflüsse sind „Komp2“ und „Komp3“. Eine Erhöhung von „Komp2“ wirkt sich positiv, eine Erhöhung von „Komp3“ negativ auf die Ausgangsgröße „Eig1“ aus. Durch Drücken der Taste **Excel** lassen sich die Werte in eine Exceltabelle übertragen.

Mittels der Taste **Grafik** (oder **Doppelklick auf den gewünschten Ausgangsparameter**) kann eine entsprechende grafische Darstellung angefordert werden.



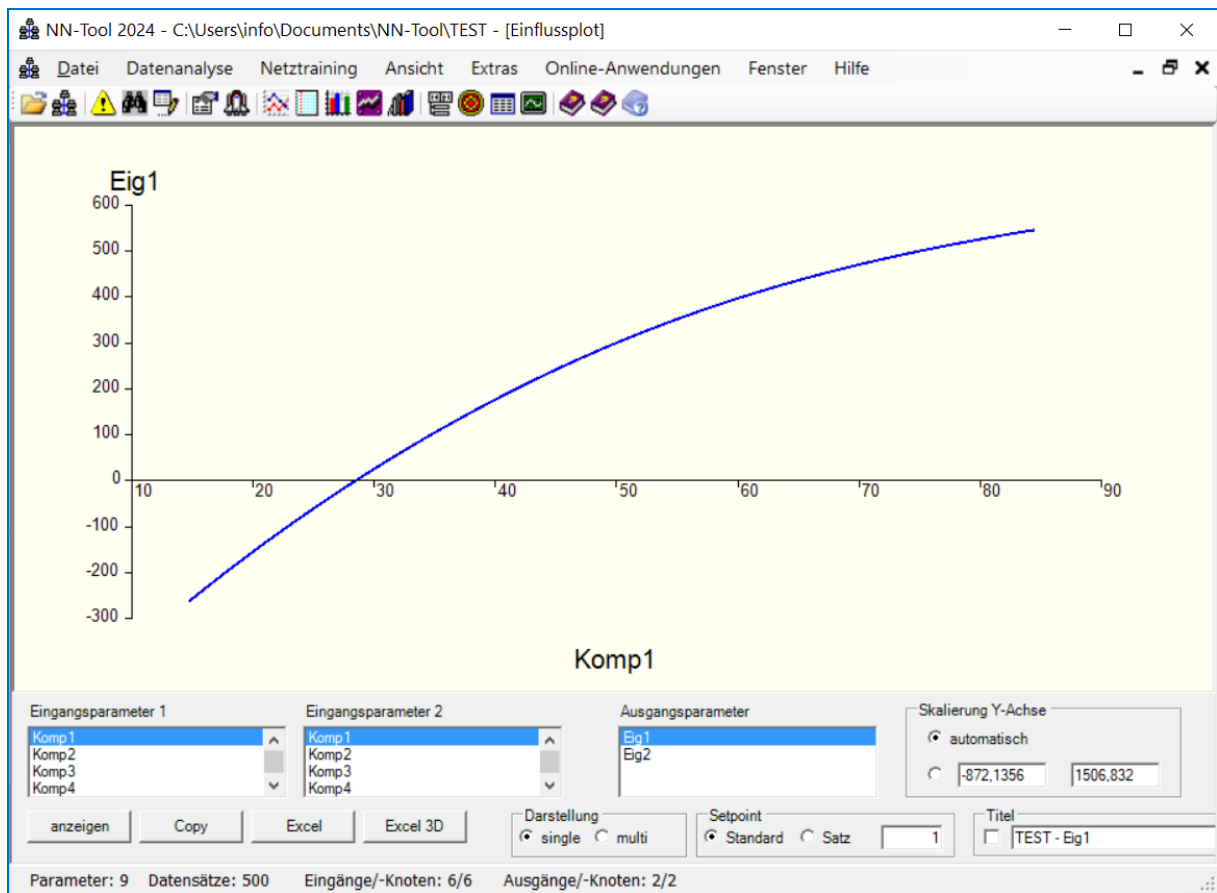


Unter Verwendung der Option „Setpoint“ werden die Einflüsse der Eingänge auf die Ausgänge für einen fest definierten Datensatz (den Setpoint) berechnet, d.h. es wird nicht über alle Datensätze gemittelt. Dazu ist ein Datensatz auszuwählen und die Taste „Neu berechnen“ zu drücken. Mittels der Option „Alle“ und der Taste „Neu berechnen“ wird wieder über alle Datensätze gemittelt.

### Wichtiger Hinweis / unbedingt ausprobieren:

An dieser Stelle sollten Sie von der Option **„Automatische Dokumentation“** Gebrauch machen. Dieses Feature im Hauptmenu **„Netztraining“** ermöglicht es, vollautomatisch sämtliche für ein Netz relevanten Informationen zu erzeugen und in einer **Excel-Mappe mit einer Vielzahl von Tabellenblättern** abzuspeichern. Dieses schließt Grafiken wie Scatter- und Verlaufsplot mit ein.

Mit Hilfe der **Option Einflußplot** kann nun der Einfluß der einzelnen Eingangsgrößen auf die Ausgangsgrößen graphisch dargestellt werden:

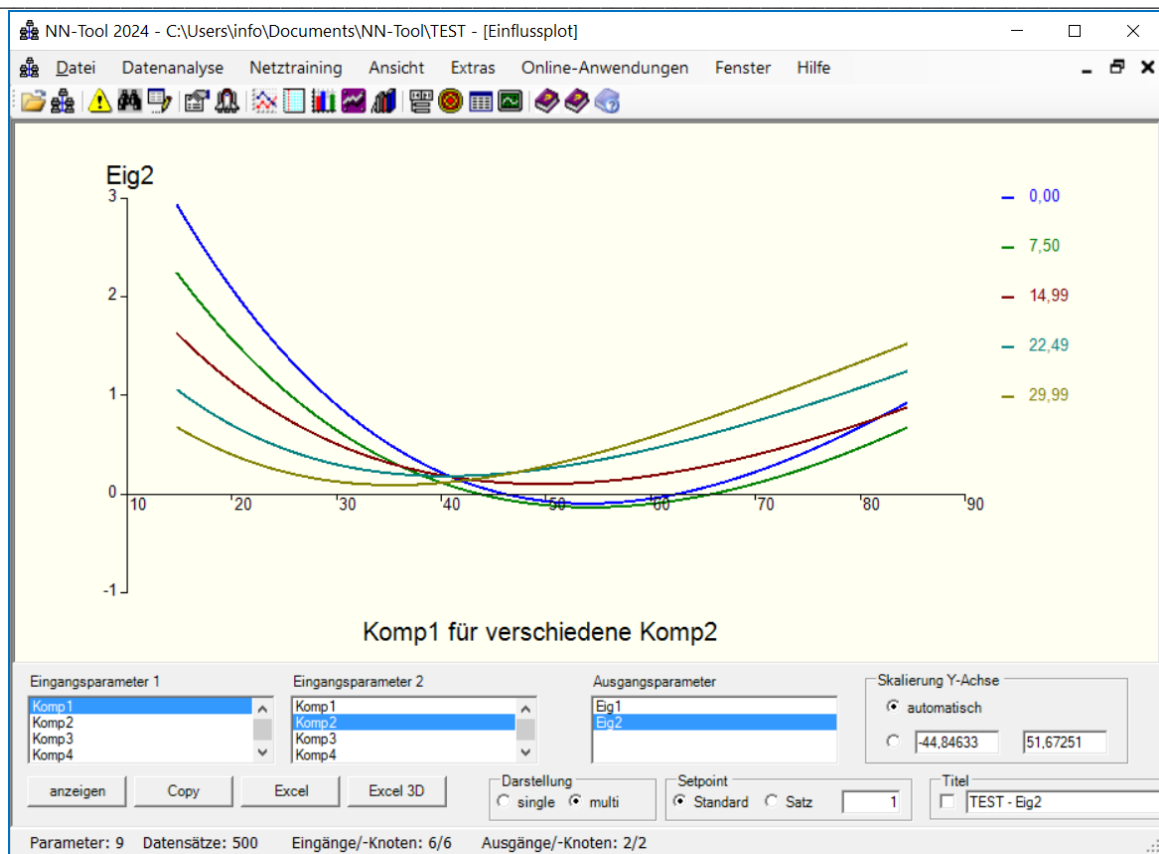


Dabei wird ein Eingang über seinen Skalenbereich variiert. Alle anderen Eingänge werden dabei auf einem Setpoint (einstellbar über die Option Setpoint) festgehalten. Standardmässig wird der Setpoint über die Mittelwerte der Eingangsgrößen festgelegt. Die Y-Achse wird zunächst auf maximale Auflösung skaliert. Sie kann aber durch Vorgabe entsprechender Werte frei skaliert werden. Über das Kästchen bei Titel kann noch ein Text hinzugefügt werden.

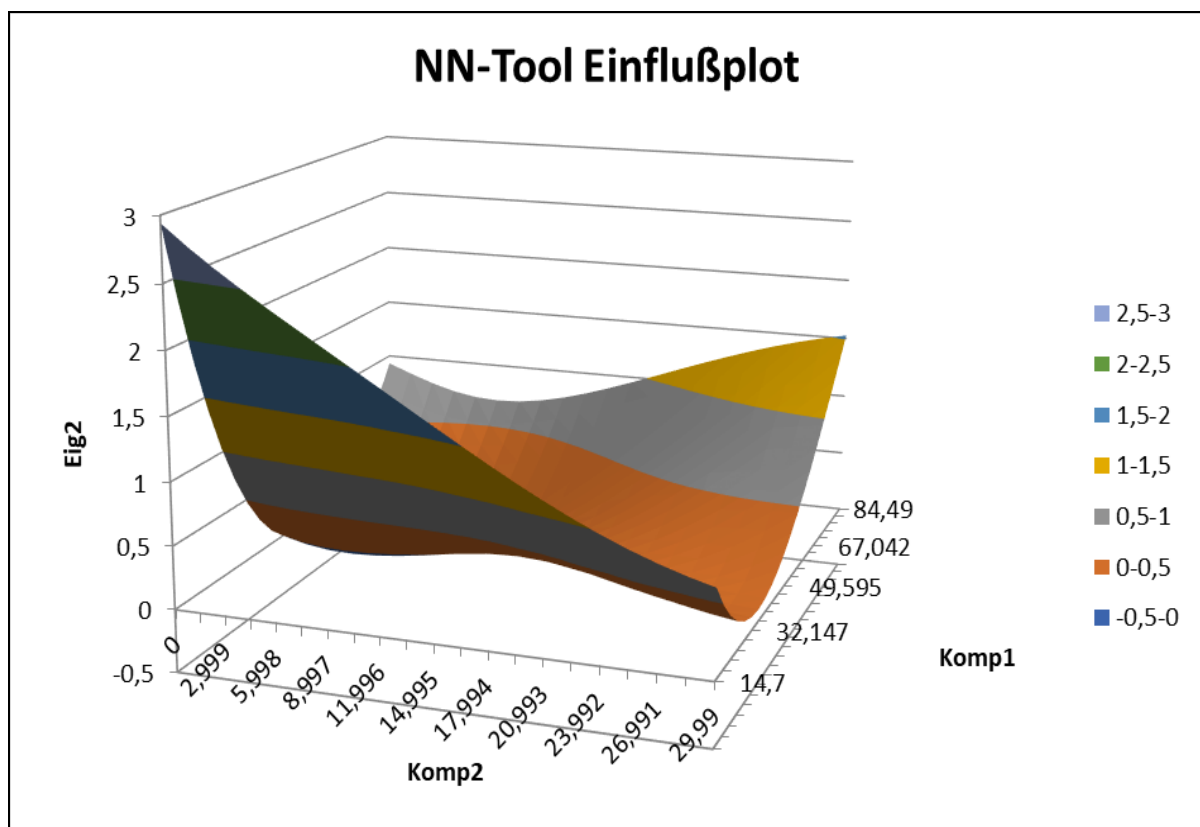
Wie bereits aus der vorhergehenden Analyse bekannt, wirkt sich „Komp3“ bei dieser Einstellung des Setpoints negativ auf „Eig1“ aus.

**Anmerkung:** Dass der Einfluß von „Komp3“ auf „Eig1“ monoton ist, ließe sich an dieser Grafik allein nicht feststellen. Es könnte ja sein, daß bei anderen als den eingestellten Werten der übrigen Einflußgrößen der Einfluß von „Komp3“ auf „Eig1“ positiv wäre.

Bei Verwendung der Option multi oder der Auswahl eines zweiten Eingangsparameters kann der Einfluß zweier Eingangsgrößen simultan dargestellt werden:



Mit „Excel“ können die Diagrammdaten nach Excel übertragen werden. Mit "Excel 3D" kann eine 3D-Darstellung unter Excel angefordert werden.



Bei den dargestellten Grafiken wurden zunächst alle Eingangsgrößen auf ihre jeweiligen Mittelwerte gesetzt. Sollen dafür andere Werte verwendet werden, ist dies durch Auswahl eines Datensatzes in der Option Setpoint möglich.

**Netz Neu laden:** Beenden Sie nun NN-Tool. Das erstellte Netz kann nach Neustart jederzeit wieder geladen werden („**Datei \ Netz laden**“) oder durch Doppelklick auf die entsprechende „**dsc**“-**Datei** (Description-Datei).

## 5. Klassifikatoren und Lernparameter

Dieser Abschnitt behandelt weitere wesentliche Eigenschaften von NN-Tool. Zur Erläuterung wird ein anderes Beispielproblem mit dem Anwendungsnamen **class6** verwendet. Starten Sie NN-Tool neu, erzeugen Sie aus der Exceldatei class6.xls die entsprechende Datendatei class6.pat. Öffnen und analysieren Sie die Datendatei. Kennungen treten nicht auf. Die Parameterbeschreibung sieht wie folgt aus:

Parameter	Min	Max	Mittel	Stdabw	Median	vollständig	Transform.	E/A	A/P	verfügbar
1 a1	19,5000	73,6300	49,8416	11,0664	48,1250	98	Lin	E	A	
2 a2	Class	3	gelb	78		97	Lin	E	A	
3 a3	5,1200	53,0700	25,0459	11,2188	24,5000	100	Lin	E	A	
4 e1	2,0000	96,0000	57,3667	28,8068	63,0000	90	Lin	E	A	
5 e2	1,0000	99,0000	52,0816	28,7527	56,0000	98	Lin	E	A	
6 e3	0,0000	99,0000	51,0600	28,9080	51,5000	100	Lin	E	A	
7 e4	1,0000	99,0000	47,4200	28,6071	43,5000	100	Lin	E	A	
8 e5	1,0000	97,0000	47,3300	30,2425	46,5000	100	Lin	E	A	
9 e6	0,0000	99,0000	50,9300	28,6276	54,5000	100	Lin	E	A	
10 e7	0,0000	98,0000	48,7000	30,3650	46,5000	100	Lin	E	A	
11 e8	Class	4	suess	69		96	Lin	E	A	
12 e9	1,0000	99,0000	47,1500	30,5477	45,0000	100	Lin	E	A	
13 Parameter_13			0,0000	0,0000	0,0000	0	Lin	E	P	
14 Parameter_14			0,0000	0,0000	0,0000	0	Lin	E	P	
15 a4	68,5100	97,4100	86,3276	6,5903	86,8900	100	Lin	E	A	
16 Parameter_16			0,0000	0,0000	0,0000	0	Lin	E	P	
17 a5	-20,0000	118,0000	56,3000	41,4535	58,5000	100	Lin	E	A	

Parameter: 17 Datensätze: 100

Dieses Beispiel enthält **zwei Klassifikatoren** (farblich gekennzeichnet):

1. a2 mit den Klassen (Instanzen): „gelb“, „rot“ und „blau“
2. e8 mit den Klassen „suess“, „sauer“, „salzig“, „bitter“

NN-Tool erkennt einen Klassifikatoren daran, dass in der entsprechenden Spalte der Daten nicht nur rein numerische Werte auftreten. Andererseits wird jeder Parameter, der nichtnumerische Elemente (z.B. Blanks) enthält, als Klassifikator behandelt.

Bei den Klassifikatoren werden in den Spalten Min, Max, Mittel, Stdabw die folgenden Werte angezeigt:

Spalte Min: Schlüsselwort **Class**.

Spalte Max: Anzahl der verschiedenen Klassen (Instanzen) des Klassifikators.

Spalte Mittel: Häufigste Klasse des Klassifikators.

Spalte Stdabw: Anzahl des Auftretens der häufigsten Klasse.

Weiterhin enthält das Beispiel einige völlig leere Spalten namens Parameter\_13, Parameter\_14, Parameter\_16. Diese Namen wurden von NN-Tool vergeben. Parameter, die nicht wenigstens zwei verschiedene Werte annehmen, werden von NN-Tool automatisch deaktiviert und können auch manuell nicht wieder aktiviert werden.

In diesem Beispiel sind die Parameter a1-a3 und a4 und a5 die Ausgangsgrößen. Nach Umschalten ergibt sich:

The screenshot shows the NN-Tool 2024 interface. The main window displays a table of parameters and their properties. Below the table are several control buttons for configuring the neural network.

Parameter	Min	Max	Mittel	Stdabw	Median	vollständig	Transform.	E/A	A/P	verfügbar
1 a1	19,5000	73,6300	49,8416	11,0664	48,1250	98	Lin	A	A	83
2 a2	Class	3	gelb	78		97	Lin	A	A	82
3 a3	5,1200	53,0700	25,0459	11,2188	24,5000	100	Lin	A	A	85
4 e1	2,0000	96,0000	57,3667	28,8068	63,0000	90	Lin	E	A	
5 e2	1,0000	99,0000	52,0816	28,7527	56,0000	98	Lin	E	A	
6 e3	0,0000	99,0000	51,0600	28,9080	51,5000	100	Lin	E	A	
7 e4	1,0000	99,0000	47,4200	28,6071	43,5000	100	Lin	E	A	
8 e5	1,0000	97,0000	47,3300	30,2425	46,5000	100	Lin	E	A	
9 e6	0,0000	99,0000	50,9300	28,6276	54,5000	100	Lin	E	A	
10 e7	0,0000	98,0000	48,7000	30,3650	46,5000	100	Lin	E	A	
11 e8	Class	4	suess	69		96	Lin	E	A	
12 e9	1,0000	99,0000	47,1500	30,5477	45,0000	100	Lin	E	A	
13 Parameter_13			0,0000	0,0000	0,0000	0	Lin	E	P	
14 Parameter_14			0,0000	0,0000	0,0000	0	Lin	E	P	
15 a4	68,5100	97,4100	86,3276	6,5903	86,8900	100	Lin	A	A	85
16 Parameter_16			0,0000	0,0000	0,0000	0	Lin	A	P	
17 a5	-20,0000	118,0000	56,3000	41,4535	58,5000	100	Lin	A	A	85

Buttons below the table:

- Eingang, Aktiv, Ausgang, Passiv
- Transformationen: Lin, Log, Sig, Auto-Eingänge, Auto-Ausgänge, Alle Lin
- nach Anzahl abschalten, Klassifikator manuell, Histogramm, Datenverlauf
- User-Modul definieren

Parameter: 17 Datensätze: 100 Eingänge/-Knoten: 9/12 Ausgänge/-Knoten: 5/7

Es bleiben 9 aktive Eingangsparameter, denen 12 Eingangsknoten zugeordnet sind, da der Klassifikator „e8“ für seine 4 Klassen zur Kodierung 4 Eingangsknoten benötigt. Entsprechend ergeben sich für die 5 aktiven Ausgangsgrößen 7 Ausgangsknoten.

Obwohl die Zusatzfunktionen für die Lernparameter auch für diese Anwendung nicht unbedingt benötigt werden, werden sie hier kurz vorgestellt:

**Transformationen/Histogramm:** diese Tasten ermöglichen bei nicht gleichmäßig verteilten Parametern eine entsprechende Transformation, um zu besser verteilten Parametern zu kommen (vgl. Anhang: Nichtlineare Skalierungstransformationen). Diese Transformationen können mit den Tasten **Lin**, **Log** und **Sig** manuell für jeden Parameter einzeln gesetzt werden. Dazu kann mittels der Taste **Histogramm** (oder Doppelclick auf einen Parameter) eine grafische Darstellung der Verteilung ausgewählter Parameter angezeigt werden. In dieser Darstellung läßt sich auch die Wirkung der Transformationen studieren (Ausprobieren!). Mittels der Tasten Auto-Eingänge, Auto-Ausgänge wird NN-Tool veranlasst, selbstständig die beste Skalierungstransformation zu wählen (getrennt für Ein- und Ausgänge). Mittels der Taste „Alle Lin“ wird auf die lineare Transformation zurückgesetzt (Standardfall).

**nach Anzahl abschalten:** Bei großen Datensätzen mit sehr vielen Parametern können mit dieser Funktion auf einen Tastendruck sämtliche Parameter auf passiv gesetzt werden, die nicht hinreichend oft gemessen worden sind.

**Klassifikator manuell:** Wie bereits erwähnt, erkennt NN-Tool einen Klassifikator am Auftreten nichtnumerischer Parameter. Nun kann es jedoch vorkommen, dass die Instanzen

eines Klassifikators selbst numerische Werte sind, z.B. Klassifikator Katalysatortyp mit den Werten 1,2,3,4 . Ein solcher Klassifikator muß manuell umgeschaltet werden. Dazu wird der entsprechende Parameter markiert und dann der Menüpunkt ausgeführt. Dies kann durch erneutes Ausführen der Analyse wieder rückgängig gemacht werden.

**Hinweis:** Falls beabsichtigt ist, das neuronale Netz als Excel-Add-In zu nutzen, sollte diese Option nicht genutzt werden! Im Gegensatz zu NN-Tool unterscheidet Excel nicht zwischen einer Zahl und einem Text, der aussieht wie eine Zahl. In solchen Fällen sollten stattdessen die Parameterwerte selbst zuvor durch Anhängen oder Voranstellen eines Buchstabens zu Klassen gemacht werden (vgl. Excelfunktion „verketten“).

**Datenverlauf:** Durch Auswählen eines Parameters und Drücken der Taste **Datenverlauf** (oder rechter Mausklick) wird eine grafische Darstellung des Verlaufs des ausgewählten Parameters angezeigt (Ausprobieren!)

### Registerblatt Lernparameter:

In Beispiel1 (Anwendung Test) wurden für jeden Ausgangsparameter Netze mit 1,2,4,8 inneren Knoten und 10, 20, 30 ... 100 Lernschritten durchgetestet.

Dieses Blatt ermöglicht eine genauere Einstellung des Lernvorgangs:

NN-Tool 2024 - C:\Users\info\Documents\NN-Tool\CLASS6 - [Datenanalyse]

Filei Datenanalyse Netztraining Ansicht Extras Online-Anwendungen Fenster Hilfe

Datenanalyse Lernparameter Input-Output-Zuordnung Lernset- / Testseteinteilung

Innere Knoten pro Ausgangsparameter  
1 min 8 max

Lernschritte  
10 min 100 max 10 Schrittweite

5 Anteil Testsätze / n-fach Crossvalidation

Aufteilung der Datensätze

- ☒ jeder n-te Satz im Testset
- ☐ zusammenhängender Testset
- ☐ spezifische Einteilung
- ☐ Crossvalidation n-fach zyklisch
- ☐ Crossvalidation n-fach blockweise
- ☐ n-fach Crossvalidation zufällig (kein Testset) - %-Anteil 20
- ☐ Crossvalidation Leave-One-Out

Specials

0 Fehlerbewertung bei Auswahl in %

Gewichtsinitialisierung  
☒ Standard (=0) ☐ Variante

Multithreading  
☐ ja ☒ nein

Parameter: 17 Datensätze: 100 Eingänge/-Knoten: 9/12 Ausgänge/-Knoten: 5/7

Der automatische Lernalgorithmus von NN-Tool erstellt pro Ausgangsknoten, d.h. für jede zu lernende Funktion automatisch ein optimales Netz. Die erstellten Netze werden zuletzt zu

einem Gesamtnetz zusammengefügt. Dies führt i.a. zu einer besseren **Prognosegüte** der Netze als wenn alle Ausgangsgrößen gemeinsam optimiert werden. Dabei wird auch für jeden Ausgangsparameter einzeln die Zahl der vollständigen Datensätze ermittelt. Dies bedeutet, dass ein Datensatz, der in einer oder mehreren Ausgangsgrößen unvollständig ist, immer noch zum Netztraining für alle anderen Ausgangsgrößen herangezogen wird (auf die Vollständigkeit der Eingangsgrößen kann jedoch wegen der Eindeutigkeit des Problems nicht verzichtet werden). Durch diese zusätzlichen Methoden wird sichergestellt, dass das Maximum an Informationen aus den Datensätzen herausgeholt wird. Der Anwender gibt dabei nur einen **Rahmen für die Optimierung** vor. Innerhalb dieses Rahmens wird das jeweils beste Netz pro Ausgangsknoten ermittelt. Die Netzgüte wird dabei durch den Fehler der Prognosen auf dem Testset ermittelt. Mit der Registerkarte Lernparameter geben Sie den oben erwähnten Rahmen vor. Im Einzelnen bedeuten die Felder:

- **Innere Knoten pro Ausgangsparameter:** Diese Werte legen die Minimal- und Maximalgröße der Teilnetze fest, z.B. 1 und 32. In diesem Fall testet das System die folgenden Netzgrößen für jeden Ausgangsparameter (bzw. Knoten bei Klassifikatoren): 1,2,4,8,16,32, d.h. die Netzgröße wird jeweils verdoppelt. (Eine Änderung von 10 auf 11 innere Knoten bringt keine spürbare Veränderung der Lernfähigkeit; aus diesem Grund wird jeweils verdoppelt). Das Gesamtnetz kann also in diesem Fall maximal aus  $7 \cdot 32$  inneren Knoten bestehen (für die 7 Ausgangsknoten je maximal 32 innere Knoten). In der Regel ergeben sich jedoch deutlich kleinere Gesamtnetze.
- **Lernschritte:** Hier wird die untere und die obere Grenze der Zahl der Lernschritte festgelegt sowie die Zahl der Lernschritte, nach denen jeweils der Testset geprüft wird. Die Einstellung 10,100,10 bedeutet: Das System macht zunächst 10 Lernschritte (Minimalzahl), anschließend wird die Netzgüte getestet, dann werden 10 weitere Lernschritte (Schrittweite) gemacht und wieder getestet, etc. bis die Maximalzahl der Schritte durchgeführt wurde. Dies wird für jeden Ausgangsknoten und jede Einstellung der inneren Knoten wiederholt.
- **Anteil Testsätze / n-fach Crossvalidation:** gibt den Anteil an, der von den vollständigen Sätzen eines Ausgangsparameters zum Testen (d.h. nicht zum Lernen) verwendet werden soll. Eine 5 bedeutet, daß jeder 5. vollständige Satz zum Testen verwendet wird. **Beachten Sie:** dies kann für jeden Ausgangsparameter eine andere Zuordnung sein, wenn die Ausgangsgrößen unvollständig sind. Falls das Training im Crossvalidationmodus durchgeführt wird (s.u.), gibt der Wert die Zahl der verschiedenen Testsets an.
- **Aufteilung der Datensätze:** Bei manchen Anwendungen hängt die Güte des resultierenden Netzmodells empfindlich davon ab, welche Datensätze zum Lernen (Lernset) und welche zum Testen verwendet werden. Dies ist insbesondere der Fall, wenn nur wenige Datensätze zur Verfügung stehen oder die Datensätze untereinander hochkorreliert sind (z.B. bei Zeitreihenanalysen). Aus diesem Grund stehen die folgenden Optionen für die Aufteilung zur Verfügung:
  - **jeder n-te Satz im Testset / zusammenhängender Testset:** gibt an, wie die Datensätze des Testsets aus den für einen Ausgangsknoten zur Verfügung stehenden Sätzen auszuwählen sind. Für Versuchsreihen ist i.a. die Option: „jeder n-te Satz“ sinnvoller, für Zeitreihen ist wegen der typischerweise hohen Korrelation benachbarter Datensätze die Option „zusammenhängender Testset“ vorzuziehen.



NN-Tool wählt die entsprechende Option selbsttätig, d.h. auch hier ist normalerweise keine Änderung notwendig.

- **spezifische Einteilung:** Ermöglicht die detaillierte Zuordnung jedes Datensatzes zum Lern- oder Testset über die Registerkarte Lernset-/Testseteinteilung. Dabei kann den Datensätzen auch zusätzlich eine Gewichtung mitgegeben werden. Dabei gilt: Gewicht 1 entspricht Standard, 0 = Satz wird nicht berücksichtigt, 2,3,4,... Satz geht entsprechend oft in die Lern- bzw. Testmenge ein.
- **Crossvalidation:** Ermöglicht es mit mehrfachen, dynamisch erzeugten Testsets pro Ausgang zu arbeiten. Diese Option verringert die Abhängigkeit der ausgewählten Netzstruktur von einem speziellen Testset. Falls beispielsweise im Feld n-fach Crossvalidation der Wert 5 eingetragen ist, werden für jeden Ausgangsparameter 5 Aufteilungen von Lern- und Testset vorgenommen. Für jede dieser Aufteilungen wird das gesamte Testprogramm bzgl. der inneren Knoten und der Lernschritte automatisch durchgeprüft. Dann wird jeder Ausgangsknoten auf allen Datensätzen mit der optimalen Netzstruktur fertigtrainiert. Zuletzt werden die einzelnen Teilnetze wieder zum Gesamtnetz integriert. **Hinweis:** Eine unter dem Punkt spezifische Einteilung vorgenommene Datengewichtung wird auch bei Crossvalidation berücksichtigt. Für die dynamische Aufteilung der Sets stehen selbst wieder **vier verschiedene Strategien** zur Verfügung:
  - **Crossvalidation n-fach zyklisch:** Bei dieser Aufteilung und  $n=5$  besteht der erste dynamische Testset aus dem 1,6,11,... vollständigen Datensatz, der zweite Testset aus dem 2,7,12,... Datensatz etc. Diese Aufteilung ist i.a. **besonders geeignet für unkorrelierte Versuchsreihen**.
  - **Crossvalidation n-fach blockweise:** Bei dieser Aufteilung und  $n=5$  besteht der erste dynamische Testset aus dem ersten Fünftel der vollständigen Datensätze, der zweite Testset aus dem zweiten Fünftel etc. Diese Aufteilung ist i.a. besonders für **zeitabhängige Daten** („Zeitreihen“, siehe entsprechendes Beispiel) **optimal** geeignet. Falls unklar ist ob benachbarte Datensätze korreliert sind, wird diese Einstellung empfohlen, d.h. **mit blockweise ist man im Zweifel auf der richtigen Seite**.
  - **Crossvalidation n-fach zufällig:** Bei dieser Aufteilung und  $n=5$  besteht jeder der fünf dynamischen Testsets aus Datensätzen, die mit der unter %-Anteil angegebenen Wahrscheinlichkeit dem Testset zugeordnet wurden. In diesem Modus kann keine endgültige Bewertung der Testsetgüte durchgeführt werden (nur Lernset).
  - **Crossvalidation Leave-One-Out:** Im Extremfall "Leave-One-Out" werden genau so viele Testsets gebildet und zur Netzbestimmung herangezogen, wie es vollständige Datensätze zum betreffenden Parameter gibt. Jeder Testset besteht dann nur aus einem Datensatz (dabei ist während des Trainingslaufs keine Korrelationsangabe möglich!). Dieser Betriebsmodus ist der **bei Weitem rechenintensivste**, liefert aber – zumindest bei unkorrelierten Daten – die **höchste Modellgenauigkeit**. Besonders für kleine Datensätze von (hoffentlich) unkorrelierten Versuchsreihen geeignet. Bei Zeitreihen ist diese Option wegen der im Allgemeinen hohen Korrelation der Datensätze untereinander nicht empfehlenswert.

- **Fehlerbewertung bei Auswahl in %:** Wie bereits erwähnt, testet NN-Tool im Automatikmode für jeden Ausgangsparameter einzeln die optimale Netzkonfiguration bei verschiedener Anzahl von Lernschritten und inneren Knoten. Zuletzt wird das Netz herangezogen, das die beste Vorhersagegenauigkeit auf dem Testset erzielt. Dazu werden normalerweise die mittleren relativen Fehler der einzelnen Netze verglichen. Mit dieser Option können nun zusätzliche Strafterme auf Netze mit vielen inneren Knoten gelegt werden. Ein Wert von  $x > 0$  bei dieser Option bedeutet, daß der mittlere rel. Fehler eines Teilnetzes mit  $n$  Knoten noch mit einem Faktor  $(1 + nx/100)$  multipliziert wird. Dies liefert dann den bewerteten Fehler. Dieser wird dann zur Netzauswahl herangezogen. Diese Option ist von Bedeutung, wenn von einer hohen Korrelation der Datensätze der Testmenge mit entsprechenden Datensätzen der Lernmenge ausgegangen werden muß. In diesem Fall stellt die Testmenge keine allein ausreichende Bewertungsgrundlage zur Verfügung.
- **Gewichtsinitialisierung:** Der Lernalgorithmus von NN-Tool ist wie alle ähnlichen Methoden ein iteratives Verfahren, bei dem ausgehend von Anfangswerten für die Gewichte des Netzes sukzessiv immer bessere Lösungen gefunden werden. Ein solches Verfahren benötigt eine Initialisierung der Netzgewichte. Bei NN-Tool sind diese anfänglich alle auf Null gesetzt. **Hinweis:** in der Literatur findet man oft die Aussage, dass diese Setzung immer einem lokalen Minimum des Lernverfahrens entsprechen würde, und dass das Lernverfahren damit nicht konvergieren würde. Dies trifft für den NN-Tool Lernalgorithmus im Allgemeinen nicht zu. Dem Verfasser ist nur ein einziges, hochsymmetrisches Beispiel bekannt (nämlich XOR), für das diese Standardinitialisierung tatsächlich ein lokales Minimum darstellt und NN-Tool dafür nicht konvergiert. In solchen Fällen kann als Gewichtsinitialisierung „Variante“ gewählt und damit dann auch Konvergenz erreicht werden.
- **Multithreading:** Mit dieser Option wird der Lernvorgang von NN-Tool - falls möglich - auf mehrere parallel ausführbare Teilprozesse (sogenannte Threads) aufgeteilt. Dieses Feature erlaubt es die einzelnen Cores von PCs mit mehreren Cores gleichmäßig auszulasten um eine höhere Rechengeschwindigkeit zu erreichen. Eine Nutzung dieser Option ist **nur unter den folgenden Randbedingungen sinnvoll:**
  1. Rechner besitzt mindestens 2, besser 4 Cores.
  2. Es sind mehrere Ausgänge zu trainieren und/oder die Option Crossvalidation wird genutzt.
  3. Der Lernvorgang ist wegen vieler Datensätze und/oder Parameter rechenaufwendig.

### Registerblatt Input-Output-Zuordnung:

Ermöglicht die Festlegung von Eingangs- zu Ausgangsparametern. Standardmäßig wird jeder aktive Eingangsparameter zur Modellierung jedes Ausgangsparameters herangezogen. Mittels dieser Funktion kann NN-Tool angewiesen werden, bestimmte Eingänge bei bestimmten Ausgängen unberücksichtigt zu lassen oder nur einen rein monotonen Zusammenhang zuzulassen. Falls ein Eingang unberücksichtigt wird, wirkt sich dies entsprechend auf die Anzahl der vollständigen Datensätze eines Ausgangsparameters aus. D.h. ein selten

gemessener Eingangsparameter, der für einen bestimmten Ausgang deaktiviert wird, reduziert nicht die Zahl der für diesen Ausgang vollständigen Datensätze.

### Registerblatt Lernset- / Testseteinteilung:

Ermöglicht die individuelle Zuordnung jedes einzelnen Satzes zum Lern- oder Testset. Diese Zuordnung wirkt sich natürlich nur dann aus, wenn bei „Aufteilung der Datensätze“ auf der Registerkarte Lernparameter die Option „spezifische Einteilung“ ausgewählt wurde. Darüber hinaus kann hier eine Gewichtung der Datensätze vorgenommen werden.

Diese Funktionen werden hier nicht benötigt, aber vielleicht experimentieren Sie damit mal. Durch erneutes Ausführen der Analyse lassen sich die Veränderungen rückgängig machen. Wir übernehmen die **Defaulteinstellung** und starten den Lernvorgang:

Ausgangsknoten	Lernknoten	Lernschritt	Korrelation (%)
3	8	30	63.1474
3	8	40	64.4086
3	8	50	64.1306
3	8	60	64.0472
3	8	70	62.9208
3	8	80	62.5038
3	8	90	62.2727
3	8	100	61.9039
Ausgangsknoten: 4			
Lerndatensätze		66	
Testdatensätze		16	
4	1	10	87.7268
4	1	20	88.6167
4	1	30	90.0035
4	1	40	90.8076
4	1	50	91.3114
4	1	60	92.6893
4	1	70	90.3785
4	1	80	90.2920
4	1	90	90.2095
4	1	100	90.1268
4	2	10	93.3681
4	2	20	93.3820
4	2	30	92.8434

Es öffnet sich ein Fenster wie oben dargestellt (Falls es zu schnell geht: diese Informationen können anschließend auch noch abgerufen werden, s.u.). In der Darstellung ist gerade die Optimierung des 3. Ausgangsknoten abgeschlossen worden und die Optimierung des 4. hat begonnen (1. Spalte). Bezüglich des 4. Ausgangsknoten sind 82 Datensätze vollständig, davon werden 66 zum Trainieren und 16 zum Testen des Teilnetzes für den 4. Knoten verwendet. Zunächst wird das Netz mit einem inneren Knoten (2. Spalte) nach 10, 20, 30... Lernschritten (3. Spalte) getestet. Die entsprechenden Korrelationen (in %) für die Prognosen auf dem Testset sind in der 4. Spalte aufgelistet. Nach dem Durchlauf werden die jeweils besten Netze zum Gesamtnetz vereinigt. Die ermittelte Netzstruktur kann im folgenden Fenster angesehen werden (Hinweis: Standardmäßig ist die Registerkarte „Korrelationen Testset“ aktiviert):

NN-Tool 2024 - C:\Users\info\Documents\NN-Tool\CLASS6 - [Modellgüte]

Menü: Datei, Datenanalyse, Netztraining, Ansicht, Extras, Online-Anwendungen, Fenster, Hilfe

Taben: Netzstruktur, Korrelationen Lemset, Korrelationen Testset

Nr.	Ausgangsparameter	Lemdatensätze	Testdatensätze	Innere Knoten	Iterationen	Korrelationen	Rel. Fehler in %
1	a1	67	16	2	100	0,9958	1,2331
2	Class a2 rot	66	16	1	100	0,4772	12,1952
3	Class a2 blau	66	16	2	20	0,9519	12,3568
4	Class a2 gelb	66	16	8	100	0,9506	3,5065
5	a3	68	17	2	20	0,8914	9,5389
6	a4	68	17	2	20	0,9013	9,1655
7	a5	68	17	4	100	0,9992	0,8480

Details

Parameter: 17   Datensätze: 100   Eingänge/-Knoten: 9/12   Ausgänge/-Knoten: 5/7   Rechenzeit: 0,71 s

**Darüberhinaus werden die gesamten Informationen in eine Datei namens class6.pro geschrieben. Die Datei kann mit dem Button „Details“ direkt angezeigt werden.** Im Fenster (bzw. der Datei) ist für jeden Ausgangsknoten die Beschreibung für das jeweils beste Einzelnetz angegeben. Z.B. ergab sich für den dritten Ausgangsknoten (entspricht der Klasse „blau“ des Klassifikators „a2“) ein optimales Netz mit 2 inneren Knoten bei 20 Lernschritten. Dieses Netz weist eine Korrelation von 95.2 % und einen rel. Fehler von 12.36% auf dem Testset auf. Zur Erstellung des entsprechenden Teilnetzes wurden 66 Datensätze herangezogen. Die Fehlerberechnungen beruhen auf den 16 Sätzen des Testsets. Die 7 Einzelnetze wurden dann zu einem Gesamtnetz mit 21 inneren Knoten zusammengebunden.

**Eine komplette Dokumentation Ihrer Modellierung erhalten Sie im Hauptmenu „Netztraining“, Menüpunkt „Automatische Dokumentation“. Unbedingt ausprobieren!!**

Anschließend kann das Netz nun zur Erstellung von Einflußplots oder zu Optimierungen mittels der integrierten Anwendungsmodule (siehe Kapitel „Anwendungsmodul Mischpult“ und Kapitel „Komplexe Optimierungsprobleme“) eingesetzt werden.

Im nächsten Kapitel wollen wir uns nun **Optionen zur Verbesserung der Modellgenauigkeit** ansehen.

## 6. Optimierung der Modellgüte – Crossvalidation und Input-Optimierungslauf

Im vorangegangenen Beispiel „class6“ hatten wir die ungünstige Situation, dass verhältnismäßig vielen Inputs (Eingangsparametern) nur relativ wenige Datensätze gegenüberstanden.

**Grundsätzlich sollte für eine gute Modellierung das Verhältnis der Zahl der Datensätze zur Anzahl der Inputs so hoch wie möglich sein. Man kann eigentlich nie zu viele Datensätze haben.**

Eine geringe Zahl von Datensätzen wirkt sich darüber hinaus auch noch dahingehend aus, dass die ermittelten **Fehler und Korrelationskoeffizienten statistisch unsicherer** sind. Den letzten Punkt gehen wir nun als erstes an, in dem wir zu **Crossvalidation** übergehen.

Laden Sie zunächst das zuvor (Kapitel 5) erstellte Netz „class6“ und wechseln Sie auf „Netztraining \ Lernparameter setzen“. Wählen Sie die Option „**Crossvalidation n-fach blockweise**“ (wir sind uns nicht sicher ob benachbarte Datensätze korreliert sind) sowie die Option „**Multithreading**“:

Parameter: 17   Datensätze: 100   Eingänge/-Knoten: 9/12   Ausgänge/-Knoten: 5/7

Starten Sie nun den Lernvorgang:

Nr.	Ausgangsparameter	Korrelation	Rel. Fehler in %	Abs. Fehler	a0-Koeffizient	a1-Koeffizient	Anzahl
1	a1	0,9965	1,1534	0,6244	-0,4142	1,0074	83
2	Class a2	0,9024					82
3	a3	0,8955	8,3115	3,9854	1,9826	0,9220	85
4	a4	0,9126	7,4291	2,1470	3,4655	0,9568	85
5	a5	0,9995	0,6953	0,9595	-0,2435	1,0016	85

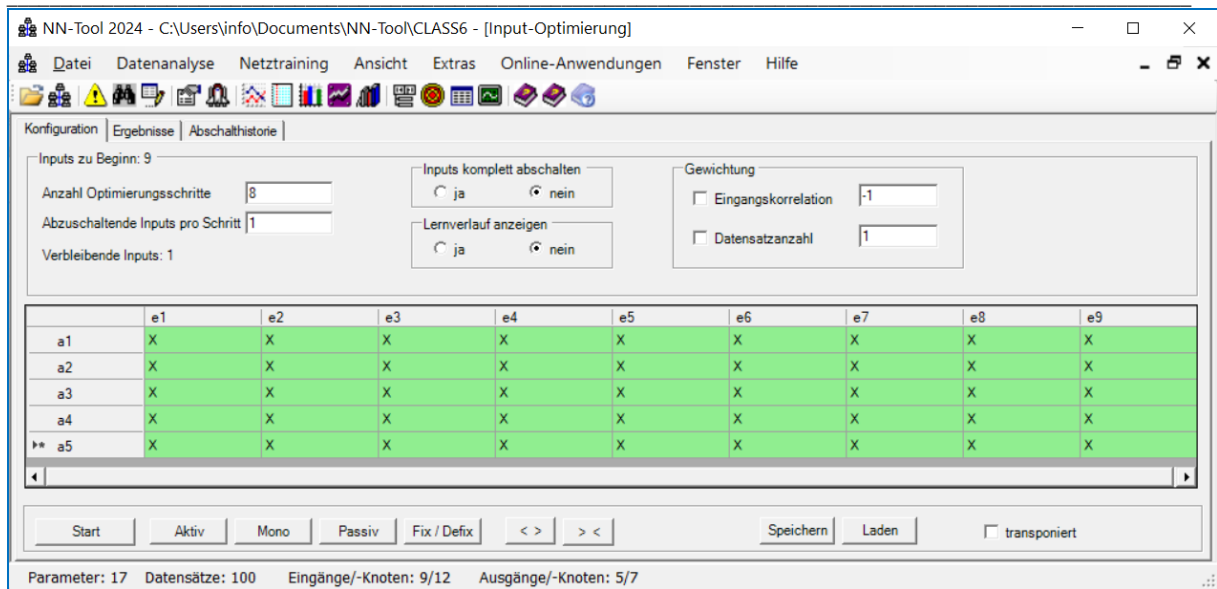
Parameter: 17   Datensätze: 100   Eingänge/-Knoten: 9/12   Ausgänge/-Knoten: 5/7   Rechenzeit: 2,06 s

Wir haben dabei zunächst eine (leichte) Verbesserung der Modellgüte erreicht. Vor allem aber sind die ermittelten Fehler und Korrelationen auf dem Testset nun besser statistisch abgesichert (Feld „Anzahl“).

Im nächsten Schritt wollen wir nun das **Verhältnis der Zahl der Datensätze zu den Inputs verbessern**. Da wir uns keine Datensätze schnitzen können, können wir stattdessen versuchen die Zahl der Inputs zu reduzieren. An dieser Stelle kommt die Option „**Input-Optimierungslauf**“ ins Spiel.

Diese Option im Hauptmenu Netztraining ermöglicht es, sukzessive (sequenziell) die Anzahl der Eingangsparameter für die verschiedenen Ausgänge zu reduzieren, um damit eine **optimale Eingangsparameterkonfiguration zu finden**. Ziel ist es, für jeden Ausgang nur die jeweils notwendigen Eingänge zu behalten und die übrigen passiv zu schalten (vgl. Input-Output-Zuordnung). Im Gegensatz zur eher üblichen Methodik „**Optimale Eingänge**“ (Anhang 4), die auf Vorab-Korrelationsanalysen beruht, werden hier die optimalen Inputs auf der Basis bereits erstellter Netze bestimmt. Dabei werden im Gegensatz zur Methodik „Optimale Eingänge“ auch **nichtlineare Einflüsse der Inputparameter** berücksichtigt. Die Methode ist also wesentlich allgemeiner (und i.A. auch genauer) als die übliche Vorgehensweise der Parameterauswahl auf der Basis von Korrelationskoeffizienten (siehe „Optimale Eingänge“). Nichts ist umsonst. Sie ist auch wesentlich rechenaufwendiger. Das Handling ist jedoch außerordentlich komfortabel.

Laden Sie zunächst das zuvor mittels Crossvalidation und Multithreading erstellte Netz „class6“ und wählen Sie „**Netztraining \ Input-Optimierungslauf**“:



### Vorgehensweise:

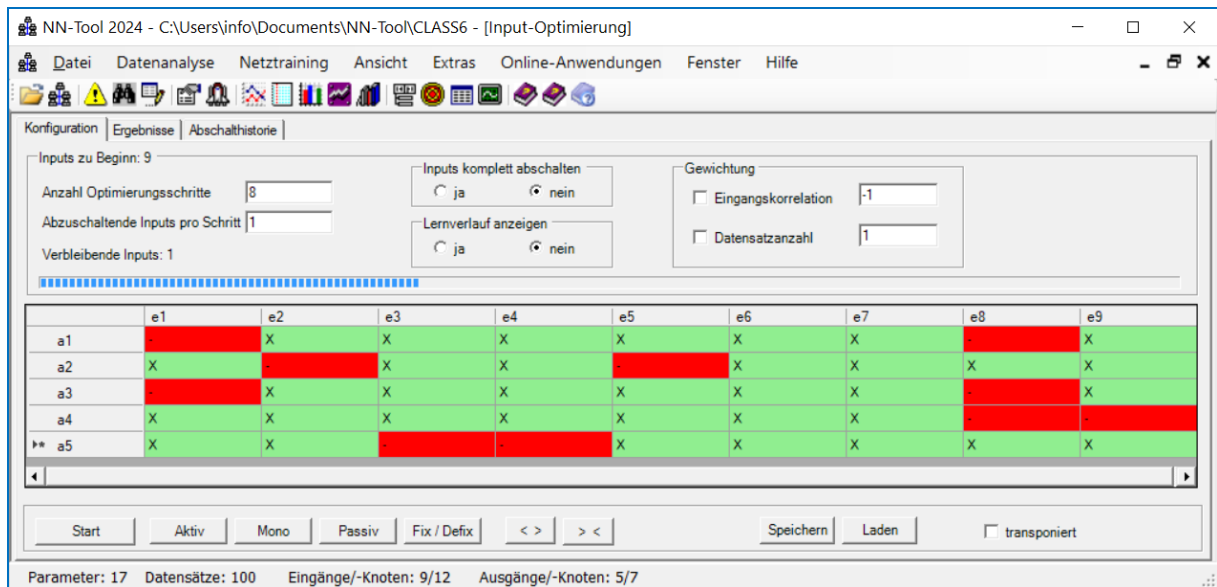
Der Nutzer legt zunächst eine Anzahl von Optimierungsschritten (z.B. 8) sowie die pro Schritt abzuschaltenden Inputs (z.B. 1) fest. Die hier gezeigten Werte sind die **Defaults**, nämlich bei  $n$  Inputs ( $n-1$ ) Schritte, bei denen jeweils nur ein Input abgeschaltet wird. Andere Einstellungen machen vor allem Sinn, falls sehr viele potentielle Inputs bei großen Datensätzen vorhanden sind (Rechenzeit). Bei hundert Inputs könnte man beispielsweise 33 Durchläufe mit je 3 abzuschaltenden Inputs ausführen. Wichtig ist es, dass zum Abschluss noch mindestens ein Input übrigbleibt. **Es wird empfohlen, den Lernvorgang mittels „Crossvalidation“ und „Multithreading“ auszuführen (Lernparameter vorab entsprechend setzen).**

Nach Drücken der **Start-Taste** erfolgt zunächst ein 0-ter Trainingslauf, bei dem sämtliche Inputs aktiv auf sämtliche Outputs wirken (es sei denn, die Parameter sind komplett passiv geschaltet). Anschließend erfolgen die Optimierungsläufe (im Beispiel 8). Dabei wird für jeden Optimierungslauf das folgende Programm automatisch abgearbeitet:

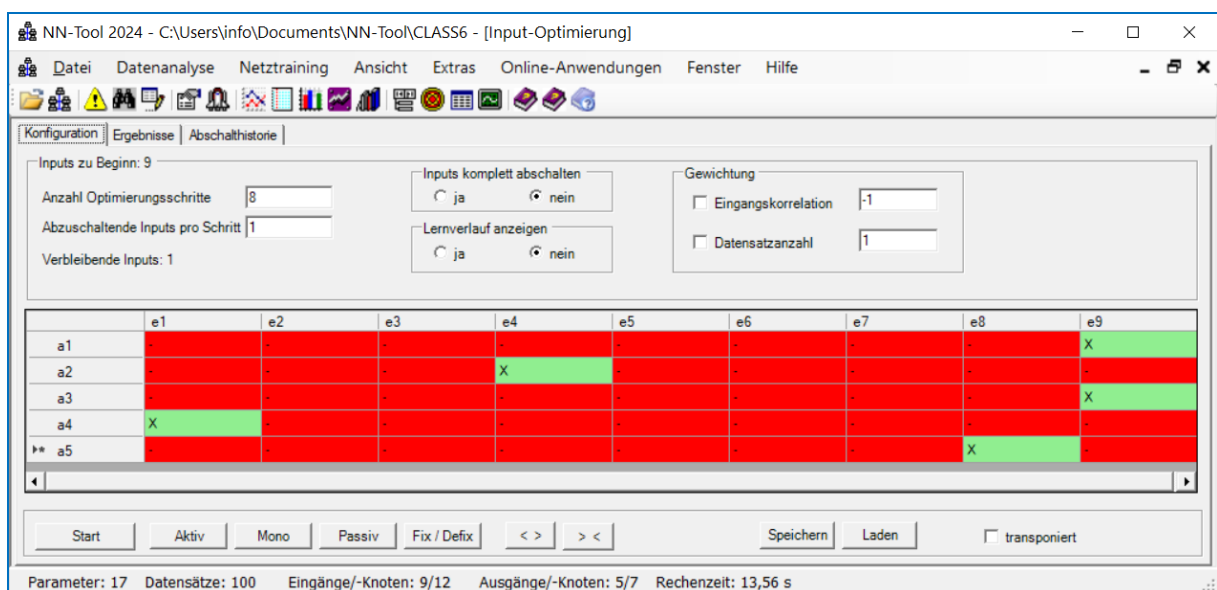
1. Auf der Basis des vorhergehenden Durchlaufs werden die Einflüsse der Inputs auf die Outputs berechnet.
2. Die Inputs mit den pro Output geringsten **absoluten** Einflüssen des vorhergehenden Laufs werden passiv gesetzt (hier jeweils 1).
3. Mit der neuen Input-Output-Konfiguration wird ein neues Netz erstellt. Dabei gilt die Einstellung unter „Lernparameter setzen“.
4. Die Netzgüte der Konfiguration wird ermittelt und abgespeichert.



## Zwischendurch:



Nach letztem Durchlauf ergibt sich das Netz mit den wenigsten Inputs (hier jeweils nur ein Input pro Output):



Bei jedem Durchlauf verringert sich die Zahl der jeweils aktiven Inputs. Nach sämtlichen Durchläufen (Achtung: wegen des 0.-ten Laufs ist es einer mehr als ausgewählt) werden die Ergebnisse für alle Outputs und alle Konfigurationen angezeigt:



NN-Tool 2024 - C:\Users\info\Documents\NN-Tool\CLASSE - [Input-Optimierung]

Datei Datenanalyse Netztraining Ansicht Extras Online-Anwendungen Fenster Hilfe

Konfiguration Ergebnisse Abschaltgeschichte

	abgeschaltet	verblieben	Korr. Lemset	rel. Fehler in %	Lernumfang	Korr. Testset	rel. Fehler in %	Testumfang	Konfiguration
a1	0	9	1.0000	0.0846	83	0.9965	1.1534	83	
a1	1	8	0.9999	0.2075	87	0.9989	0.5685	87	
a1	2	7	0.9522	5.0360	96	0.9307	6.0422	96	
a1	3	6	0.9174	6.6239	96	0.8754	8.0867	96	
a1	4	5	0.8662	8.1206	96	0.8201	9.3182	96	
a1	5	4	0.7770	10.6470	98	0.7403	11.4735	98	
a1	6	3	0.7289	10.9869	98	0.6938	11.6087	98	
a1	7	2	0.6148	12.7676	98	0.5652	13.6104	98	
a1	8	1	0.5188	13.6461	98	0.4877	13.9365	98	X
a2	0	9	1.0000		82	0.9024		82	
a2	1	8	1.0000		83	0.9277		83	
a2	2	7	1.0000		83	0.9398		83	
a2	3	6	1.0000		83	0.9277		83	
a2	4	5	1.0000		83	0.9518		83	
a2	5	4	1.0000		87	0.9770		87	
a2	6	3	1.0000		87	1.0000		87	
a2	7	2	1.0000		87	0.9885		87	
a2	8	1	0.8041		97	0.8144		97	X
a3	0	9	0.9959	1.5694	85	0.8955	8.3115	85	
a3	1	8	0.9967	1.4704	89	0.9352	6.5252	89	
a3	2	7	0.9246	7.0423	98	0.8390	9.3952	98	
a3	3	6	0.8863	8.4625	100	0.8306	10.3868	100	

Konfiguration

Inaktive Inputs entfernen ☒ ja ☐ nein

Auswahl nach

Parameter: 17 Datensätze: 100 Eingänge/-Knoten: 9/12 Ausgänge/-Knoten: 5/7 Rechenzeit: 13,56 s

**Achtung:** Das erstellte Netz ist nun dasjenige mit den wenigsten Inputs. Die entsprechende Konfiguration ist durch X in der Spalte „Konfiguration“ markiert. Im Allgemeinen ist dies jedoch nicht die optimale Konfiguration (vor allem nicht für jeden Output).

Mit der Maus (durch Klicken in die Spalte Konfiguration) kann nun für jeden Ausgang eine gewünschte Konfiguration ausgewählt werden. Alternativ kann durch Drücken der Tasten **Korrelation** bzw. **Fehler** die jeweils optimale Konfiguration für alle Outputs ausgewählt werden. **Durch Drücken der Taste Erstellen wird diese Konfiguration dann erzeugt.** Mit der Taste **Dokumentation** kann der gesamte Vorgang dokumentiert werden.

### Weiteres Vorgehen:

a) Drücken Sie zunächst die Taste „Fehler“:

NN-Tool 2024 - C:\Users\info\Documents\NN-Tool\CLASS6 - [Input-Optimierung]

Datei Datenanalyse Netztraining Ansicht Extras Online-Anwendungen Fenster Hilfe

Konfiguration Ergebnisse Abschaltgeschichte

	abgeschaltet	verblieben	Korr. Lemset	rel. Fehler in %	Lernumfang	Korr. Testset	rel. Fehler in %	Testumfang	Konfiguration
a1	0	9	1,0000	0,0846	83	0,9965	1,1534	83	
a1	1	8	0,9999	0,2075	87	0,9989	0,5685	87	X
a1	2	7	0,9522	5,0360	96	0,9307	6,0422	96	
a1	3	6	0,9174	6,6239	96	0,8754	8,0867	96	
a1	4	5	0,8662	8,1206	96	0,8201	9,3182	96	
a1	5	4	0,7770	10,6470	98	0,7403	11,4735	98	
a1	6	3	0,7289	10,9869	98	0,6938	11,6087	98	
a1	7	2	0,6148	12,7676	98	0,5652	13,6104	98	
a1	8	1	0,5188	13,6461	98	0,4877	13,9365	98	
a2	0	9	1,0000		82	0,9024		82	
a2	1	8	1,0000		83	0,9277		83	
a2	2	7	1,0000		83	0,9398		83	
a2	3	6	1,0000		83	0,9277		83	
a2	4	5	1,0000		83	0,9518		83	
a2	5	4	1,0000		87	0,9770		87	
a2	6	3	1,0000		87	1,0000		87	X
a2	7	2	1,0000		87	0,9885		87	
a2	8	1	0,8041		97	0,8144		97	
a3	0	9	0,9959	1,5694	85	0,8955	8,3115	85	
a3	1	8	0,9967	1,4704	89	0,9352	6,5252	89	X
a3	2	7	0,9246	7,0423	98	0,8390	9,3952	98	
a3	3	6	0,8863	8,4625	100	0,8306	10,3868	100	

Konfiguration

Inaktive Inputs entfernen ☒ ja ☐ nein

Auswahl nach

Erstellen

Parameter: 17 Datensätze: 100 Eingänge/-Knoten: 9/12 Ausgänge/-Knoten: 5/7 Rechenzeit: 14,89 s

Damit wird für jeden Output die bezüglich des Fehlers optimale Konfiguration ausgewählt, zu erkennen an den X-Positionen.

- Drücken Sie dann die Taste „**Dokumentation**“. Es wird eine Excelmappe erzeugt, die den gesamten Ablauf protokolliert.
- Drücken Sie **zuletzt** die Taste „**Erstellen**“. Erst dadurch wird die ausgewählte Konfiguration erzeugt.

NN-Tool 2024 - C:\Users\info\Documents\NN-Tool\CLASS6 - [Modellgüte]

Datei Datenanalyse Netztraining Ansicht Extras Online-Anwendungen Fenster Hilfe

Netzstruktur Korrelationen Lernset Korrelationen Testset

Nr.	Ausgangsparameter	Korrelation	Rel. Fehler in %	Abs. Fehler	a0-Koeffizient	a1-Koeffizient	Anzahl
1	a1	0,9989	0,5685	0,3077	-0,3332	1,0057	87
2	Class a2	1,0000					87
3	a3	0,9352	6,5252	3,1288	0,4996	0,9693	89
4	a4	0,9246	6,5049	1,8799	3,4863	0,9601	89
5	a5	1,0000	0,1747	0,2411	0,1000	0,9996	96

Scatterplot Verlaufplot Histogramm Excel

Parameter: 17 Datensätze: 100 Eingänge/-Knoten: 9/12 Ausgänge/-Knoten: 5/7 Rechenzeit: 1,96 s

**Es ergibt sich ein bezüglich der Modellgenauigkeit deutlich verbessertes Netz.** Die Input-Output-Zuordnung zeigt die erstellte Konfiguration:

	e1	e2	e3	e4	e5	e6	e7	e8	e9
a1	X	X	X	X	X	X	X	-	X
a2	X	-	-	X	-	X	-	-	-
a3	X	X	X	X	X	X	X	-	X
a4	X	X	X	X	X	X	X	-	X
a5	-	-	-	-	-	-	X	X	-

Parameter: 17   Datensätze: 100   Eingänge/-Knoten: 9/12   Ausgänge/-Knoten: 5/7   Rechenzeit: 1,96 s

Abschließend wird dann wieder die Option „**Automatische Dokumentation**“ empfohlen.

### Hinweise:

- 1) **Häufiger Fehler:** Es kann leicht passieren, dass die Taste „**Erstellen**“ gedrückt wird, ohne dass zuvor die optimale Netzkonfiguration ausgewählt wurde. In dem Fall wird das Netz mit der minimalen Anzahl von Inputs generiert. Dieses zeigt i.A. eine sehr schlechte Modellgenauigkeit. Dann muss die Input-Output-Zuordnung zurückgesetzt werden und der gesamte Vorgang wiederholt werden.
- 2) Bei **unvollständigen Eingangsparametern** kann sich die Anzahl der pro Ausgang verfügbaren Datensätze durch das sequenzielle Abschalten von Inputs während des Ablaufs ändern. Damit ändern sich dann ggf. auch die Lern- und Testset- Einteilungen. Um in diesen Fällen statistisch valide Ergebnisse zu bekommen, wird die **Verwendung von Crossvalidation empfohlen.**

### Zusätzliche Optionen:

- **Registerblatt Abschalthistorie:** Die Tabelle gibt an, in welchem Schritt des Verfahrens der jeweilige Inputparameter abgeschaltet wurde.
- **Inputs komplett abschalten:** Bei dieser Option werden diejenigen Inputs bestimmt, deren maximaler Einfluss auf sämtliche Ausgänge am geringsten ist und dann komplett für alle Ausgänge abgeschaltet.
- **Lernverlauf anzeigen:** Bei Wahl der Option „Lernverlauf anzeigen nein“ läuft der Lernalgorithmus von NN-Tool unsichtbar im Hintergrund ab. Der Laufbalken wird (aus technischen Gründen) immer nur nach einem kompletten Lernvorgang aktualisiert. Es kann also ein wenig dauern.
- **Gewichtung:** Durch Setzen der beiden möglichen Optionen „Eingangskorrelation“ und „Datensatzanzahl“ wird ein gewichtetes Kriterium zur Bestimmung der abzuschaltenden Inputs angewendet. Für den i-ten Inputparameter gilt die Bewertungsformel:

$$P(i) = 10 \cdot AI(i) + \alpha \cdot MC(i) + \beta \cdot RDC(i)$$

Die Performance P des i-ten Inputs ergibt sich aus dem **10fach gewichteten** absoluten Einfluss „AI“ des Inputs auf den gerade betrachteten Output (bzw. auf alle Outputs bei der Option „Inputs komplett abschalten“) und den mit den Parametern  $\alpha$  und  $\beta$

gewichteten Straftermen „**MC**“ (= maximale Korrelation des i-ten Inputs zu den übrigen aktiven Inputs) und „**RDC**“ (= relative Datensatzanzahl = Anzahl des Auftretens des i-ten Inputs / Gesamtzahl der Datensätze). Da eine möglichst geringe Korrelation der Inputs untereinander gewünscht ist, ist der Parameter  $\alpha$  natürlich negativ zu wählen. Diese beiden Werte können in der Benutzeroberfläche vorgegeben werden. Ist der entsprechende Haken nicht gesetzt, wird der zugehörige Parameter auf Null gesetzt. Dann werden die Inputs nur anhand ihrer Einflüsse abgeschaltet.

Im kommenden Kapitel werden wir uns eine spezielle Modellstruktur ansehen, die insbesondere für das Thema „**Predictive Maintenance**“ von Interesse ist.

## 7. Messdatenüberwachung – Data Reconciliation und Predictive Maintenance

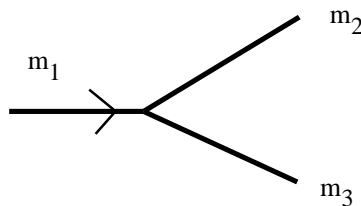
Diese NN-Tool Komponente ermöglicht es eine Vielzahl von Messstellen eines Prozesses bzw. einer großtechnischen Anlage automatisch zu überwachen. Dabei sollen die folgenden **Ziele** erreicht werden:

- **Überwachung sämtlicher relevanter Messstellen von Prozessen / Anlagen / Teilanlagen.**
- **Data Reconciliation:** Korrektur von Messfehlern.
- **Berechnung fehlender Messgrößen.**
- **Predictive Maintenance:** frühzeitige Detektion unzulässiger Betriebszustände.
- **Möglichst geringer Engineeringaufwand.**

**Methode:** Kombination von neuronalen Netzen und Hauptkomponentenanalyse (nichtlineare PCA).

Der wesentliche Ansatz beruht in der üblicherweise starken Redundanz der Messwerte.

**Simplex Beispiel:** An einer Verzweigung der folgenden Form



gilt für die Massenströme  $m_1 = m_2 + m_3$ . Die physikalischen Größen  $m_1$ ,  $m_2$ ,  $m_3$  liegen also in der durch  $m_3 = m_1 - m_2$  definierten 2-dimensionalen Fläche. Wir haben also 3 Messwerte bei nur 2 Freiheitsgraden. Diese überzähligen (redundanten) Messungen können zur Messdatenüberwachung genutzt werden.

Bei einer **großtechnischen Anlage** hat man in der Regel die folgende Situation:

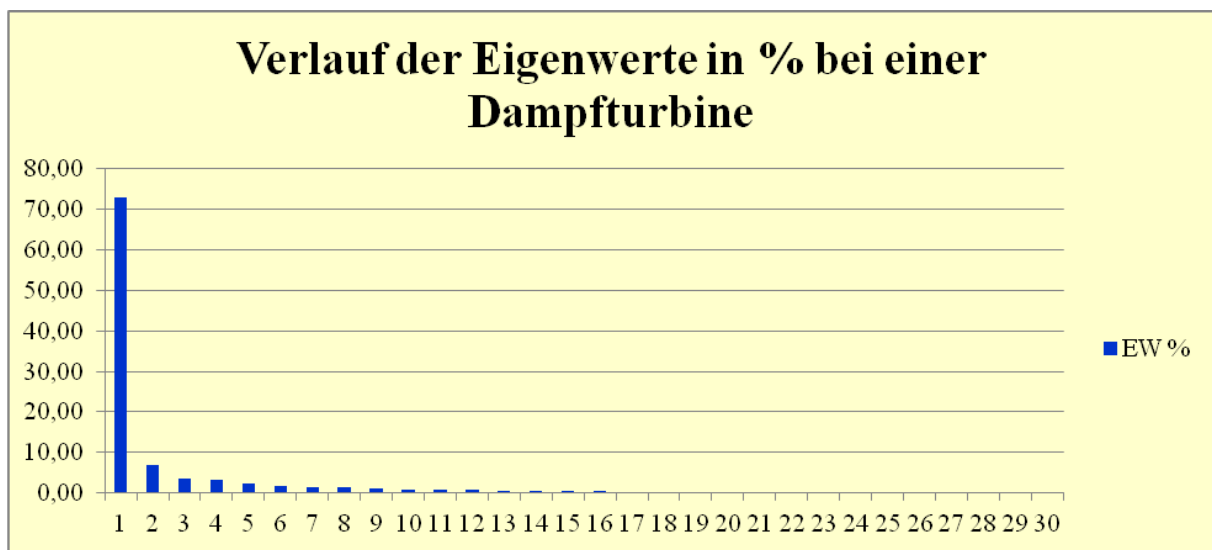
- Vielzahl redundanter Messgrößen.
- Bestehende physikalische Beziehungen reduzieren die Freiheitsgrade.
- Zahl der Freiheitsgrade  $f$  wesentlich kleiner als Anzahl der Messgrößen.
- Prozesszustand bereits durch  $f$  Messwerte eindeutig bestimmt.
- Nutzung weiterer Messwerte zur Messdatenüberwachung.
- Aber Modell erforderlich:
  - Physikalisches Modell
  - Statistisches Modell aus Messdaten

Wir wollen nun mit geringstmöglichen Engineeringaufwand eine Messdatenüberwachungsanwendung („Data Reconciliation Anwendung“) automatisch auf der

Basis von Messdaten erzeugen. Der Ansatz besteht in einer Kombination der Hauptkomponentenanalyse (PCA = Principal Component Analysis) mit neuronalen Netzen. Ausgangspunkt ist die sogenannte **Korrelationsmatrix** (siehe auch Anhang: Spezielle Funktionen zur Datenanalyse und Vorverarbeitung). Dafür gilt:

- **Korrelationsmatrix** besteht aus den Korrelationskoeffizienten jeder Messgröße mit jeder anderen.
- **Hauptkomponentenanalyse (PCA)**: Bestimmung der Eigenwerte und Eigenvektoren der Korrelationsmatrix (lineare Algebra).
- **Eigenwerte** der Korrelationsmatrix sind der Schlüssel zur Bestimmung der Anzahl der **Freiheitsgrade** des Prozesses.
- **Eigenvektoren** zu den größten Eigenwerten bilden (linear) **optimale Zustandsgrößen** des Prozesses.
- **Messdatenüberwachung** ergibt sich durch **Rückrechnung** aller Messgrößen aus den Eigenvektoren.
- Rückrechnung kann linear (klassische PCA) oder nichtlinear durch neuronale Netze erfolgen.

Über die PCA ermitteln wir die Eigenwerte der Korrelationsmatrix. Die Eigenwerte enthalten die wesentliche Information über die Freiheitsgrade des Systems, d.h. wie viele Variablen sind wirklich unabhängig vorgebar. Das folgende Diagramm zeigt die Situation bei einer Dampfturbine:



- Mehr als 60 Messgrößen. Es gibt also auch entsprechend viele Eigenwerte. Im Diagramm ist der Verlauf für die 30 größten Eigenwerte dargestellt.
- Aber nur ca. ein Dutzend relevante Eigenwerte / Freiheitsgrade. Die ersten 11 Eigenwerte enthalten bereits mehr als 95% der Gesamteigenwertstärke.
- Übrige Eigenwerte bilden im Wesentlichen nur das Messrauschen ab. Die entsprechenden Komponenten können vernachlässigt werden.
- Die Messgrößen lassen sich dann aus den Eigenvektoren zu den relevanten Eigenwerten rekonstruieren. D.h. das System ist durch Vorgabe von 11 Werten bereits bestimmt.

## Vorgehensweise:

1. Lade und analysiere Datei mit Prozessdaten.
2. Berechne Korrelationsmatrix.
3. Führe PCA durch.
4. Bestimme Anzahl relevanter Eigenwerte (Freiheitsgrade).
5. Berechne Transformationen auf die Eigenvektoren (und zurück).
6. Erweitere Prozessdaten um Koordinaten in den Eigenvektoren.
7. Erzeuge neuronales Netzmodell von den Koordinaten zu den Messdaten.
8. Verknüpfe Transformationen und Netzmodell zur fertigen Messdatenüberwachungsanwendung.
9. Kopple Anwendung mit Prozessleitsystem.

Die Punkte 1. bis 8. lassen sich nun automatisch mittels der NN-Tool Komponente Data Reconciliation durchführen. Für die Ankoppelung (Punkt 9) steht eine entsprechende Library zur Verfügung. Die Vorgehensweise wird im Folgenden an der Beispieldatei „**DataRec\_Demo.xls**“ im NN-Tool Verzeichnis erläutert. Die Excel-Mappe enthält die Tabellenblätter „**DataRec\_Demo**“ und „**DataRec\_Error\_2**“.

- Erzeugen Sie zunächst die entsprechenden .pat-Files.
- Öffnen Sie nun **DataRec\_Demo.pat**, setzen Sie den **Parameter „Data Time“ als Kennung (wichtig!!)**, und führen Sie die Analyse durch:

Parameter: 20    Datensätze: 200

Parameter	Min	Max	Mittel	Stdabw	Median	vollständig	Transform.	E/A	A/P	verfügbar
1 Date Time	Kennung					200	Lin	K	P	
2 Parameter_1	-1,3157	0,6836	-0,2650	0,4304	-0,2793	200	Lin	E	A	
3 Parameter_2	-1,1589	0,2967	-0,3448	0,2943	-0,3491	200	Lin	E	A	
4 Parameter_3	-1,5678e-03	1,6317	0,7002	0,3180	0,6797	200	Lin	E	A	
5 Parameter_4	0,2572	1,8329	0,9501	0,3436	0,9469	200	Lin	E	A	
6 Parameter_5	-1,7866	-0,5923	-1,1543	0,2606	-1,1381	200	Lin	E	A	
7 Parameter_6	-1,5843	0,8323	-0,2615	0,4627	-0,2597	200	Lin	E	A	
8 Parameter_7	-0,2091	2,8237	1,0884	0,5458	1,0753	200	Lin	E	A	
9 Parameter_8	-1,5959	0,1381	-0,6433	0,3503	-0,6471	200	Lin	E	A	
10 Parameter_9	-0,1786	1,4888	0,5972	0,3696	0,5782	200	Lin	E	A	
11 Parameter_10	-0,9098	0,1516	-0,3279	0,2254	-0,3106	200	Lin	E	A	
12 Parameter_11	-1,0008	2,0391	0,8397	0,5395	0,9360	200	Lin	E	A	
13 Parameter_12	-2,8213	1,5599	-0,4747	0,7975	-0,4254	200	Lin	E	A	
14 Parameter_13	-2,3921	2,0867	-0,1637	0,8944	-0,1905	200	Lin	E	A	
15 Parameter_14	-2,4890	4,2384	0,6536	0,9444	0,6388	200	Lin	E	A	
16 Parameter_15	-3,5764	1,8320	-0,6284	0,7496	-0,6160	200	Lin	E	A	
17 Parameter_16	-1,7485	2,2105	0,8483	0,6682	0,8916	200	Lin	E	A	
18 Parameter_17	-3,9572	3,0860	-1,3486	1,1372	-1,4838	200	Lin	E	A	
19 Parameter_18	-2,4516	3,4683	1,2891	0,9552	1,3976	200	Lin	E	A	
20 Parameter_19	-4,0735	2,5880	0,5087	1,0209	0,5461	200	Lin	E	A	

Buttons: Eingang, Aktiv, Ausgang, Passiv, Transformationen (Lin, Log, Sig, Auto-Eingänge, Auto-Ausgänge, Alle Lin), nach Anzahl abschalten, Klassifikator manuell, Histogramm, Datenverlauf, User-Modul definieren

- Die Anwendung enthält 19 numerische Parameter zwischen denen mathematische Beziehungen bestehen. Welche? Wie viele? Zunächst unklar.

- Starten Sie nun im Hauptmenu „**Netztraining**“ den Punkt „**Data Reconciliation**“. Dabei werden alle aktiven numerischen Parameter einbezogen (hier 19). Klassifikatoren können z.Z. nicht berücksichtigt werden. Es erscheint das folgende Fenster:

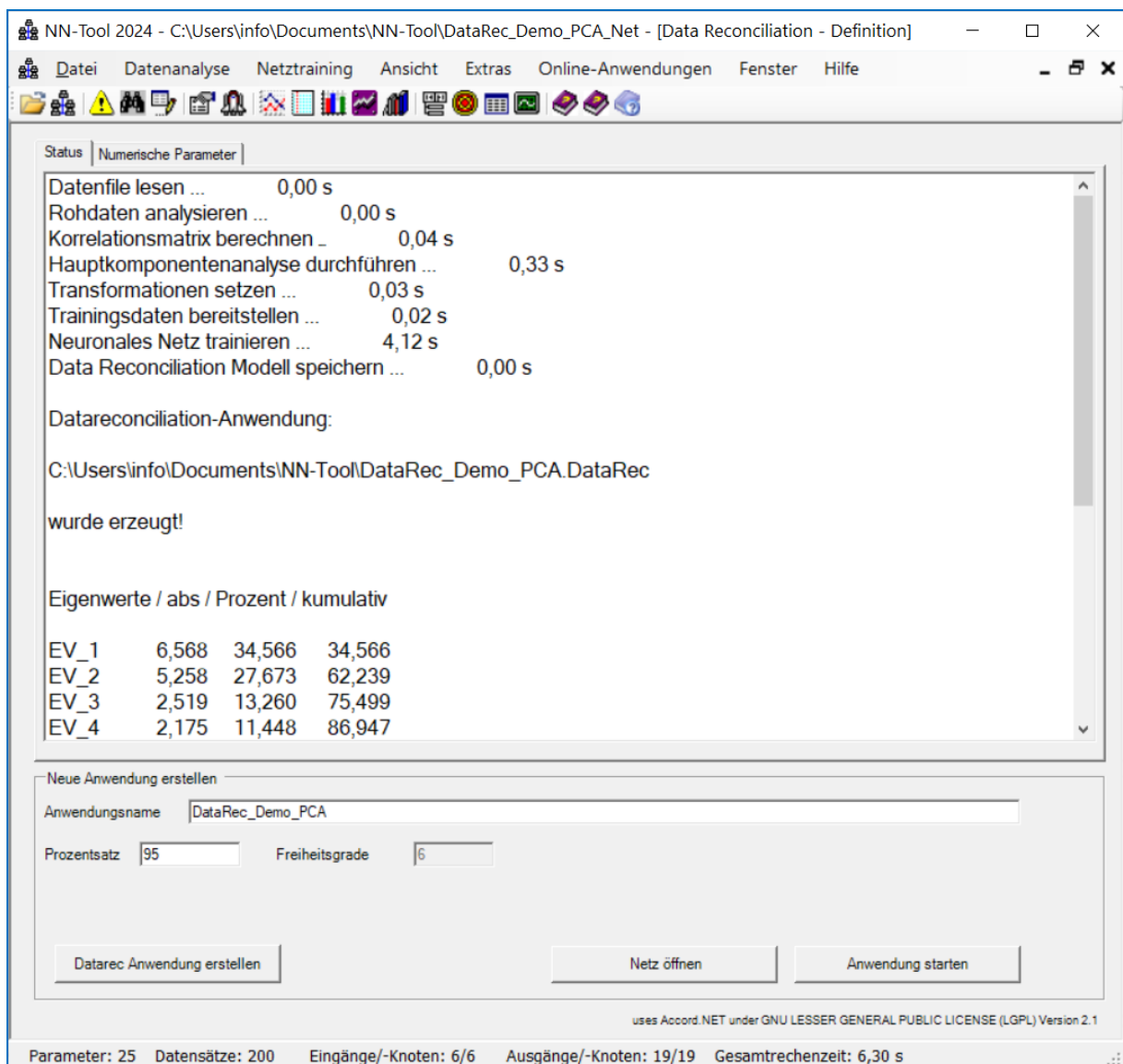
Hier können Sie zunächst einige numerische Parameter der Methode einstellen:

- Zunächst müssen Sie sich entscheiden ob Sie die Zahl der Freiheitsgrade direkt festsetzen wollen (Option „**PCA – Freiheitsgrade setzen**“) oder ob die Zahl der Freiheitsgrade auf der Basis des **vorgegebenen Prozentsatzes** automatisch ermittelt werden soll. Ein **Prozentsatz von 95%** (wie im Beispiel) bedeutet: „Berücksichtige so viele Eigenwerte, dass die aufsummierten ersten f vielen Eigenwerte 95% der Gesamtsumme der Eigenwerte überschreiten“. Diese Option wird empfohlen.
- Die beiden Optionen „**PCA – Maximalzahl Iterationsschritte**“ und „**PCA – Rekonstruktionsgenauigkeit**“ regeln das Verfahren zur Berechnung der Hauptkomponenten, falls ein Datensatz unvollständig ist. In diesem Fall kann die Berechnung dennoch durchgeführt werden, so lange mehr Messwerte in einem Datensatz vorhanden sind als die Zahl der Freiheitsgrade f beträgt. Beispiel: 19



Messstellen bei 6 Freiheitsgraden bedeutet, dass ein einzelner Datensatz dann noch überwacht werden kann, wenn er noch mindestens 7 Messgrößen aufweist, d.h. der Satz darf maximal 12 fehlende Messwerte haben. Die Berechnung erfordert im Falle von fehlenden Messgrößen jedoch ein iteratives numerisches Verfahren, welches durch die genannten Parameter gesteuert wird.

- Die weiteren Parameter steuern dann den **Lernalgorithmus** des neuronalen Netzes in der üblichen Weise. Da die Daten bei typischen Anwendungen in der Regel zeitlich aufeinander folgende Messungen sind, werden die Optionen „**zusammenhängender Testset**“ bzw. bei nicht zu großen Datensätzen vor allem die Option „**Crossvalidation n-fach blockweise**“ empfohlen.
- Wählen Sie die Option „**Crossvalidation n-fach blockweise**“ und drücken Sie die Taste „**Datarec Anwendung erstellen**“. Das System führt dann die notwendigen Verfahrensschritte durch



und zeigt dann die ermittelte Anzahl von Freiheitsgraden (hier 6) sowie die erzielte Modellgenauigkeit an:

NN-Tool 2024 - C:\Users\info\Documents\NN-Tool\DataRec\_Demo\_PCA\_Net - [Modellgüte]

Datei Datenanalyse Netztraining Ansicht Extras Online-Anwendungen Fenster Hilfe

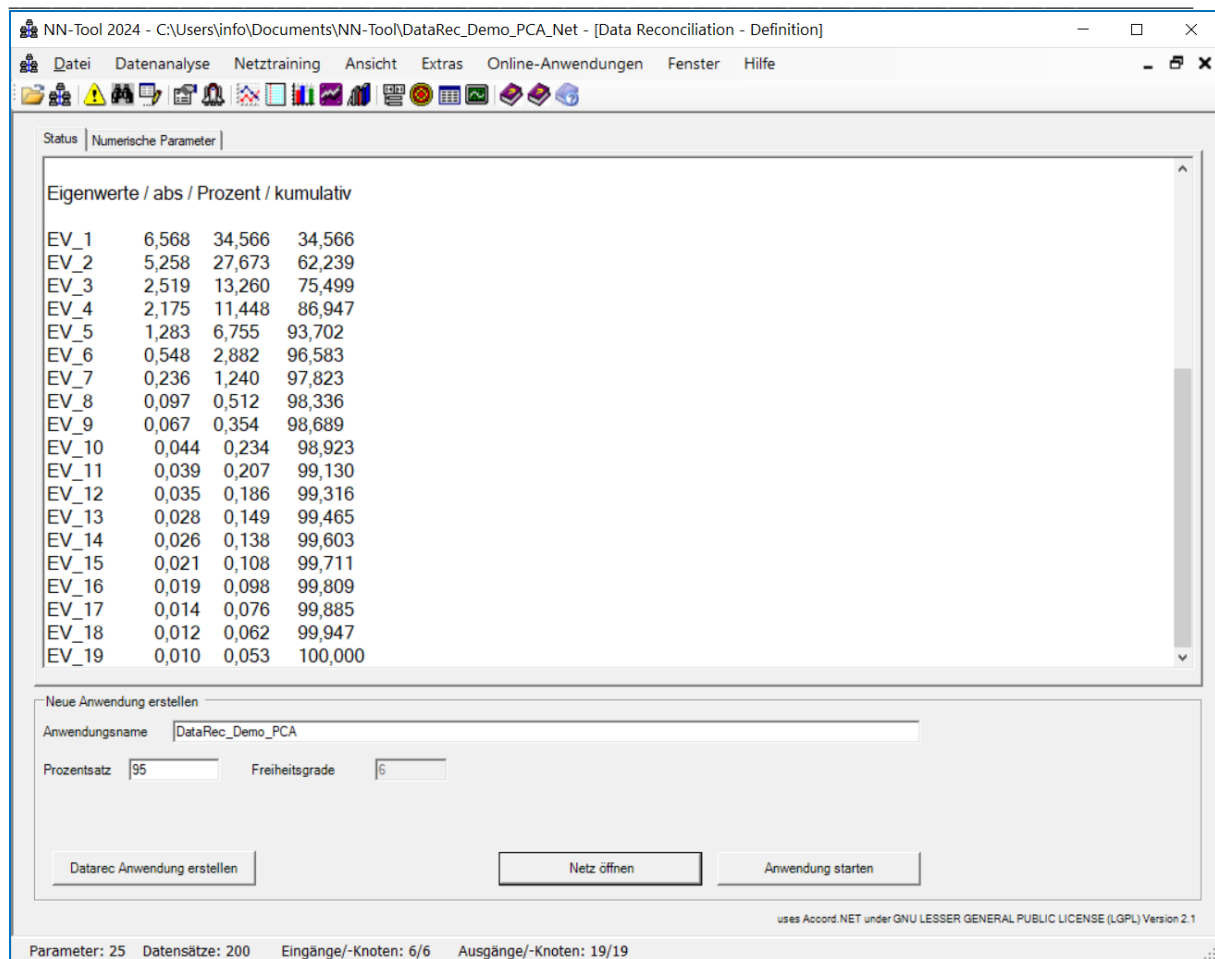
Netzstruktur Korrelationen Lernset **Korrelationen Testset**

Nr.	Ausgangsparameter	Korrelation	Rel. Fehler in %	Abs. Fehler	a0-Koeffizient	a1-Koeffizient	Anzahl
1	Parameter_1	0.9885	2.4952	0.0499	-3.2569e-03	0.9832	200
2	Parameter_2	0.9825	3.0596	0.0445	-9.0488e-03	0.9764	200
3	Parameter_3	0.9767	3.4727	0.0567	0.0314	0.9539	200
4	Parameter_4	0.9701	4.2756	0.0674	0.0449	0.9541	200
5	Parameter_5	0.8916	8.1350	0.0972	-0.2046	0.8248	200
6	Parameter_6	0.9865	2.4051	0.0581	-7.8374e-03	0.9699	200
7	Parameter_7	0.9743	3.1408	0.0953	0.0330	0.9697	200
8	Parameter_8	0.9633	4.0863	0.0709	-0.0438	0.9329	200
9	Parameter_9	0.9841	3.1105	0.0519	0.0158	0.9754	200
10	Parameter_10	0.9744	3.4746	0.0369	-0.0154	0.9499	200
11	Parameter_11	0.9905	1.9192	0.0583	0.0156	0.9812	200
12	Parameter_12	0.9954	1.2971	0.0568	-4.7352e-03	0.9909	200
13	Parameter_13	0.9947	1.4816	0.0664	-1.9154e-03	0.9915	200
14	Parameter_14	0.9889	1.3068	0.0879	0.0149	0.9701	200
15	Parameter_15	0.9932	1.1960	0.0647	-9.0346e-03	0.9826	200
16	Parameter_16	0.9852	2.1334	0.0845	0.0313	0.9681	200
17	Parameter_17	0.9884	1.8231	0.1284	-0.0308	0.9800	200
18	Parameter_18	0.9884	1.8547	0.1098	0.0273	0.9820	200
19	Parameter_19	0.9926	1.1901	0.0793	0.0283	0.9639	200

Scatterplot Verlaufsplot Histogramm Excel

Parameter: 25 Datensätze: 200 Eingänge/-Knoten: 6/6 Ausgänge/-Knoten: 19/19 Gesamtrechnzeit: 6,30 s

Schließen Sie das Fenster für die Modellgüte (mit „Netz öffnen“ können Sie es jederzeit wieder hervorholen):



Wie Sie an der Darstellung sehen können, reichen hier die ersten 6 Eigenwerte aus um die Marke von 95% zu überschreiten.

Drücken Sie nun die Taste „**Anwendung starten**“ und drücken Sie im neuen Fenster auf „**Start**“:

NN-Tool 2024 - C:\Users\info\Documents\NN-Tool\DataRec\_Demo\_PCA\_Net - [Data Reconciliation - Anwendung]

Datei Datenanalyse Netztraining Ansicht Extras Online-Anwendungen Fenster Hilfe

Nr.	Input	gemessen	korrigiert	Delta	Mitt. Abs. Fehler	Delta Rel. %	Mitt. Rel. Fehler	Ratio
1	Parameter_1	0.6014	0.5935	0.0079	0.0499	0.3935	2.4953	0.1577
2	Parameter_2	-0.6143	-0.6412	0.0270	0.0445	1.8515	3.0594	0.6052
3	Parameter_3	0.8538	0.8713	0.0175	0.0567	1.0696	3.4727	0.3080
4	Parameter_4	0.5562	0.6130	0.0569	0.0674	3.6086	4.2756	0.8440
5	Parameter_5	-1.3891	-1.3035	0.0856	0.0972	7.1679	8.1349	0.8811
6	Parameter_6	0.0929	0.0417	0.0512	0.0581	2.1178	2.4053	0.8805
7	Parameter_7	1.1816	1.0573	0.1244	0.0953	4.1002	3.1408	1.3055
8	Parameter_8	-0.6043	-0.7280	0.1238	0.0709	7.1368	4.0863	1.7465
9	Parameter_9	1.0929	1.2100	0.1171	0.0519	7.0214	3.1106	2.2573
10	Parameter_10	-0.1949	-0.2096	0.0147	0.0369	1.3875	3.4747	0.3993
11	Parameter_11	0.0257	0.0813	0.0556	0.0583	1.8299	1.9192	0.9534
12	Parameter_12	-1.6751	-1.7504	0.0753	0.0568	1.7183	1.2971	1.3247
13	Parameter_13	-0.2746	-0.2812	0.0066	0.0664	0.1472	1.4816	0.0994
14	Parameter_14	1.9816	2.1812	0.1995	0.0879	2.9662	1.3068	2.2699
15	Parameter_15	-1.9642	-1.7174	0.2468	0.0647	4.5633	1.1960	3.8156
16	Parameter_16	0.4045	0.3922	0.0123	0.0845	0.3103	2.1334	0.1455

Start [25] Satznr. Farbfaktoren [2.5] Gelb [Offnen] Direkt [ ] Threads [7] Seconds [ ]

Stop Fortsetzen [600] Zeitintervall (ms) [4.0] Rot [Speichern] [Excellexport] [ ]

Data Reconciliation - Anwendung

... [C:\Users\info\Documents\NN-Tool\DataRec\_Demo\_PCA\DataRec

Eingabedatei ... [C:\Users\info\Documents\NN-Tool\DataRec\_Demo\_PCA.pat

Ausgabedatei ... [C:\Users\info\Documents\NN-Tool\DataRec\_Demo\_PCA\_Results.pat

Parameter: 25 Datensätze: 200 Eingänge/-Knoten: 6/6 Ausgänge/-Knoten: 19/19

Es werden nun sukzessive die Datensätze aus der „Eingabedatei“ eingelesen und mittels der gerade erstellten **Data Reconciliation Anwendung** „C:\Users\baer\Documents\NN-Tool\DataRec\_Demo\_PCA.DataRec“ korrigiert (Spalten „gemessen“ und „korrigiert“). Die Spalte „Delta“ misst die Differenz. Falls die Differenz im Verhältnis zum Modellfehler zu groß ausfällt, erfolgt eine Farbmarkierung (hier gelb). Da hier zunächst nur die Trainingsdaten durch das System durchgeschoben werden, treten hier kaum Abweichungen auf. Stoppen Sie den Vorgang und wählen Sie nun als Eingabedatei die am Anfang erzeugte Datei „DataRec\_Error\_2.pat“ und starten Sie dann erneut:

Nr.	Input	gemessen	korrigiert	Delta	Mittl. Abs. Fehler	Delta Rel. %	Mittl. Rel. Fehler	Ratio
1	Parameter_1	-0.1098	-0.1101	0.0003	0.0499	0.0128	2.4953	0.0051
2	Parameter_2	-0.2520	-0.2502	0.0018	0.0445	0.1236	3.0594	0.0404
3	Parameter_3	0.7940	0.7256	0.0684	0.0567	4.1853	3.4727	1.2052
4	Parameter_4		0.7933	0.0175	0.0674	1.1094	4.2756	0.2596
5	Parameter_5		-0.9088	0.0226	0.0972	1.8898	8.1349	0.2323
6	Parameter_6		0.5349	0.0072	0.0581	0.3000	2.4053	0.1247
7	Parameter_7	2.1168	1.9473	0.1694	0.0953	5.5863	3.1408	1.7786
8	Parameter_8	-0.3343	-0.2332	0.1011	0.0709	5.8314	4.0863	1.4270
9	Parameter_9	1.0186	1.0697	0.0511	0.0519	3.0618	3.1106	0.9843
10	Parameter_10	-0.4556	-0.5193	0.0637	0.0369	6.0034	3.4747	1.7277
11	Parameter_11	0.0379	0.0590	0.0211	0.0583	0.6934	1.9192	0.3613
12	Parameter_12	-0.3825	-0.3883	0.0057	0.0568	0.1307	1.2971	0.1008
13	Parameter_13	0.8307	0.8385	0.0078	0.0664	0.1737	1.4816	0.1172
14	Parameter_14	-0.2014	-0.1933	0.0080	0.0879	0.1193	1.3068	0.0913
15	Parameter_15	0.0564	0.0526	0.0038	0.0647	0.0707	1.1960	0.0591
16	Parameter_16	-0.1770	-0.1347	0.0423	0.0845	1.0686	2.1334	0.5009

Start [5] Satznr. Farbfaktoren [2.5] Gelb [Offnen] Direkt [ ] Threads [7] Seconds [ ]  
 Stop Fortsetzen [600] Zeitintervall (ms) [4.0] Rot [Speichern] [Excellexport] [ ]  
 Data Reconciliation - Anwendung  
 ... [C:\Users\info\Documents\NN-Tool\DataRec\_Demo\_PCA\DataRec]  
 Eingabedatei ... [C:\Users\info\Documents\NN-Tool\DataRec\_Error\_2.pat]  
 Ausgabedatei ... [C:\Users\info\Documents\NN-Tool\DataRec\_Demo\_PCA\_Results.pat]  
 Parameter: 25 Datensätze: 200 Eingänge/-Knoten: 6/6 Ausgänge/-Knoten: 19/19

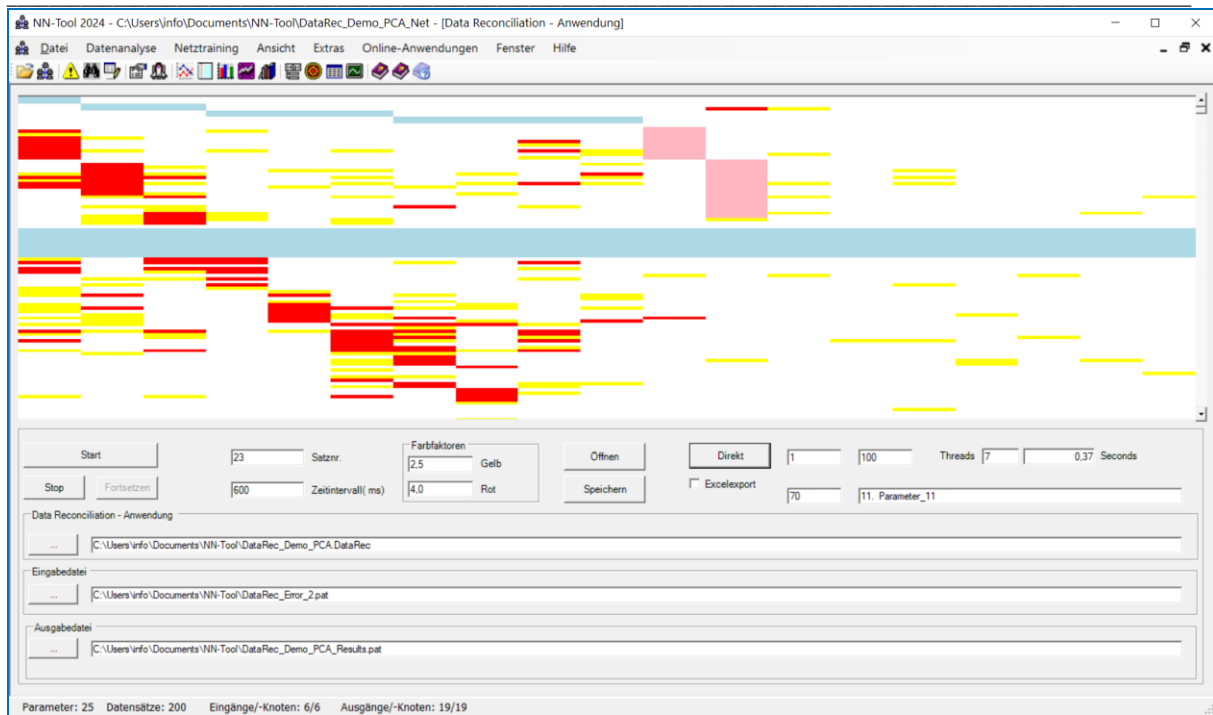
Die blau unterlegten Zellen in der Spalte gemessen weisen darauf hin, dass die entsprechenden Messwerte nicht vorhanden sind. Ab Datensatz 11 treten in dieser Eingabedatei dann auch massive Messfehler auf:

Nr.	Input	gemessen	korrigiert	Delta	Mittl. Abs. Fehler	Delta Rel. %	Mittl. Rel. Fehler	Ratio
1	Parameter_1	-0.5927	-0.5889	0.0038	0.0499	0.1914	2.4953	0.0767
2	Parameter_2	0.2000	-0.0748	0.2748	0.0445	18.8771	3.0594	8.1702
3	Parameter_3	0.9040	0.9131	0.0092	0.0567	0.5614	3.4727	0.1617
4	Parameter_4	0.8892	0.8382	0.0510	0.0674	3.2371	4.2756	0.7571
5	Parameter_5	-1.2005	-0.9330	0.2675	0.0972	22.3979	8.1349	2.7533
6	Parameter_6	-0.2960	-0.4468	0.1508	0.0581	6.2406	2.4053	2.5945
7	Parameter_7	0.9881	1.0958	0.1077	0.0953	3.5502	3.1408	1.1304
8	Parameter_8	-0.4264	-0.4674	0.0410	0.0709	2.3667	4.0863	0.5792
9	Parameter_9	0.4756	0.4209	0.0547	0.0519	3.2777	3.1106	1.0537
10	Parameter_10	-0.4365	-0.4547	0.0183	0.0369	1.7202	3.4747	0.4951
11	Parameter_11	0.9909	0.9951	0.0043	0.0583	0.1400	1.9192	0.0730
12	Parameter_12	-0.6594	-0.6182	0.0411	0.0568	0.9388	1.2971	0.7237
13	Parameter_13	-0.6133	-0.5295	0.0838	0.0664	1.8700	1.4816	1.2621
14	Parameter_14	0.9402	1.0869	0.1467	0.0879	2.1803	1.3068	1.6685
15	Parameter_15	-1.1334	-0.9703	0.1631	0.0647	3.0151	1.1960	2.5210
16	Parameter_16	1.0905	1.1616	0.0711	0.0845	1.7963	2.1334	0.8420

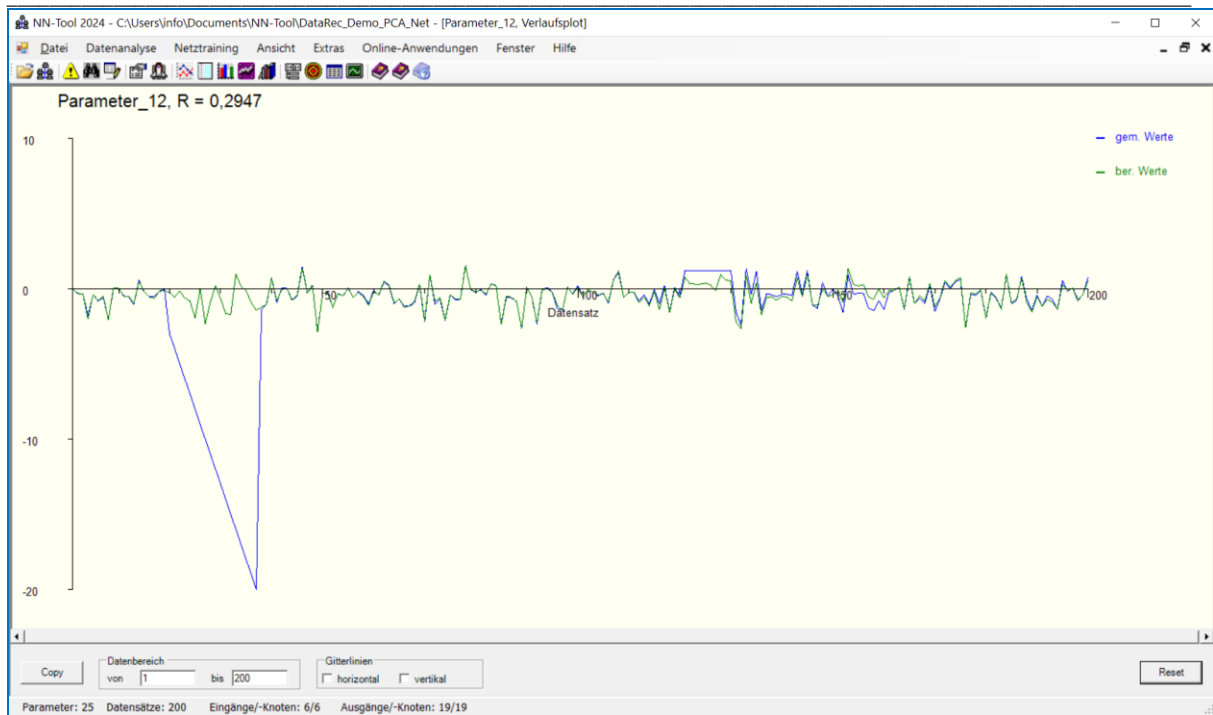
Start [23] Satznr. Farbfaktoren [2.5] Gelb [Offnen] Direkt [ ] Threads [7] Seconds [ ]  
 Stop Fortsetzen [600] Zeitintervall (ms) [4.0] Rot [Speichern] [Excellexport] [ ]  
 Data Reconciliation - Anwendung  
 ... [C:\Users\info\Documents\NN-Tool\DataRec\_Demo\_PCA\DataRec]  
 Eingabedatei ... [C:\Users\info\Documents\NN-Tool\DataRec\_Error\_2.pat]  
 Ausgabedatei ... [C:\Users\info\Documents\NN-Tool\DataRec\_Demo\_PCA\_Results.pat]  
 Parameter: 25 Datensätze: 200 Eingänge/-Knoten: 6/6 Ausgänge/-Knoten: 19/19

Die Farbmakierung in der Spalte „Ratio“ markiert denjenigen Wert, der die größte Abweichung zeigt (fehlende / rekonstruierte Messwerte ausgenommen).

Mittels der Taste „Direkt“ lassen sich alle Datensätze auf einmal prognostizieren. Gleichzeitig werden die Ratiowerte in dem folgenden Diagramm dargestellt:



Jede Zeile der Darstellung entspricht einem Datensatz, hier im Range von 1 bis 100. Weitere Sätze werden mittels des Scrollrades angezeigt. Fährt man mit der Maus in das Diagramm werden die Nummer des entsprechenden Datensatzes (hier Nummer 70) sowie die Parameterbezeichnung angezeigt (hier 11. Parameter namens Parameter\_11). Die Farben blau, gelb und rot haben die bereits bekannten Bedeutungen, die Farbe weiß entspricht einem Messwert, der nicht zu stark von dem berechneten Wert abweicht (Ratio < Gelbwert, d.h. nach bisheriger Bezeichnung grün). Darüber hinaus treten auch noch rosafarbene Markierungen auf. Diese weisen darauf hin, dass der entsprechende Messwert seinen Trainingsrange verlassen hat. Ein solcher Messwert kann ggf. die gesamte Datenvalidierung stark verzerren. Aus diesem Grund wird er für die Datenvalidierung nicht berücksichtigt, sondern rekonstruiert. Durch Doppel- oder Rechtsklick im Diagramm lassen sich Verlaufsplots der entsprechenden Spalte aufrufen:



Hier sieht man sehr gut den Effekt des Verlassens des Trainingsranges um Datensatz 25 herum.

Setzen Sie das Zeitintervall auf „1“ und speichern Sie die gesamte Konfiguration unter einem Dateinamen vom Typ „**DataRecApp**“ ab.

Nach einem Neustart von NN-Tool können Sie die Anwendung im Hauptmenu „**Online-Anwendungen**“, Menüpunkt „**Data Reconciliation – Anwendung**“ wieder aufrufen.

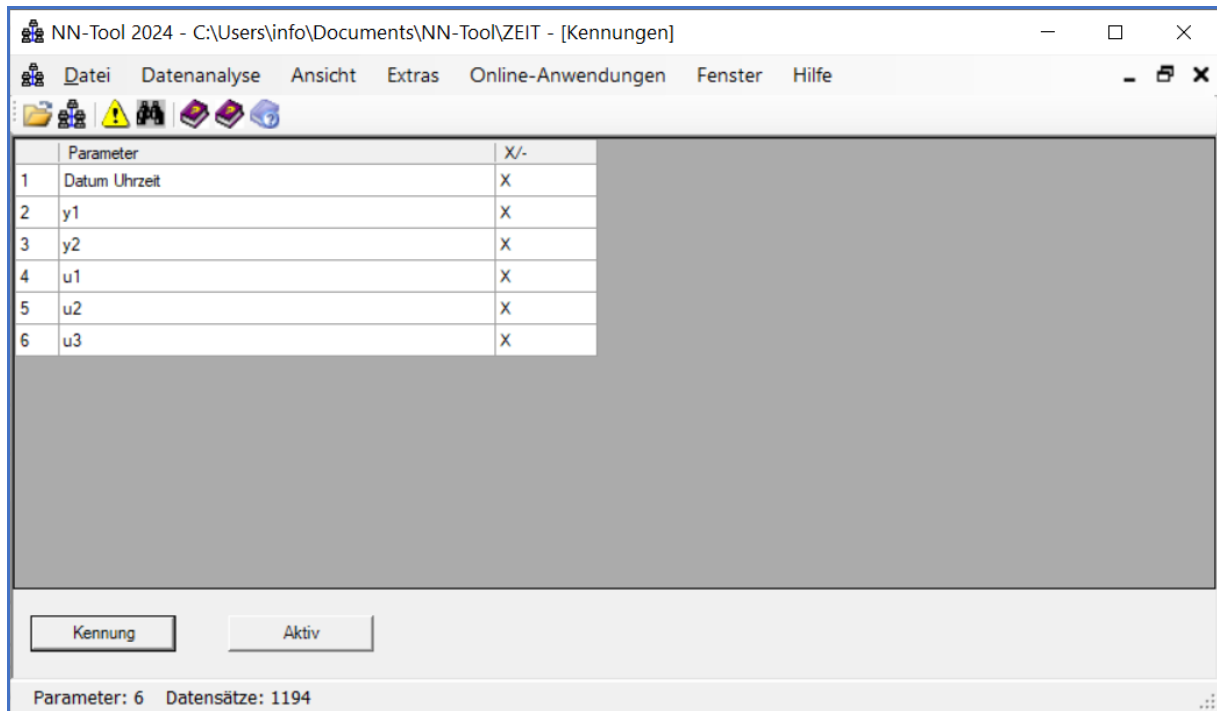
Für eine **Einbindung in Online-Systeme** steht eine entsprechende .NET-Library zur Verfügung (erhältlich auf Anfrage).

## Hinweise:

- Z.Zt. kann die Methodik nur für numerische Parameter genutzt werden.
- Für eine ggf. notwendige Vorabfilterung der Rohdaten steht die Funktionalität „**Datenfile filtern**“ im Hauptmenu **Datei** zur Verfügung (siehe **Anhang 13**).
- Beim Auftreten von Kennungen sollten diese unbedingt entsprechend markiert werden, da es sonst beim Anwendungsmodul zu Inkonsistenzen mit alternativen Testdatensätzen kommen kann.
- Zur Durchführung der Hauptkomponentenanalyse wird die Accord.Net-Library eingesetzt. Die Verwendung erfolgt unter den Bedingungen der „GNU LESSER GENERAL PUBLIC LICENSE“. Die Bedingungen können im Anhang eingesehen werden.

## 8. Zeitreihen

Charakteristisch für die Modellierung von Zeitreihen ist es, dass die Modelle Parameter zu unterschiedlichen Zeiten enthalten. Typischerweise liegen die Meßdaten jedoch in einer Form vor, dass zu bestimmten Zeittakten die Meßwerte genommen werden. Die zwischen den Parametern auftretenden Totzeiten müssen also mitverarbeitet werden. Die Vorgehensweise wird am Beispiel zeit.xls erläutert. Starten Sie NN-Tool, erzeugen und öffnen Sie die Datendatei. Es folgt ein Hinweis, dass (aus Performancegründen zunächst) nur die ersten 1000 Datensätze im Datenblatt angezeigt werden. Mittels der entsprechenden Taste lassen sich alle anzeigen. Wählen Sie nun den Menüpunkt **Kennungen deaktivieren**:



Der erste Parameter ist vom Typ Datum/Uhrzeit, würde also von NN-Tool als Klassifikator mit über 1000 Klassen behandelt, was nicht in unserem Sinne ist. Deaktivieren Sie daher diesen Parameter und führen Sie anschließend die Analyse aus:



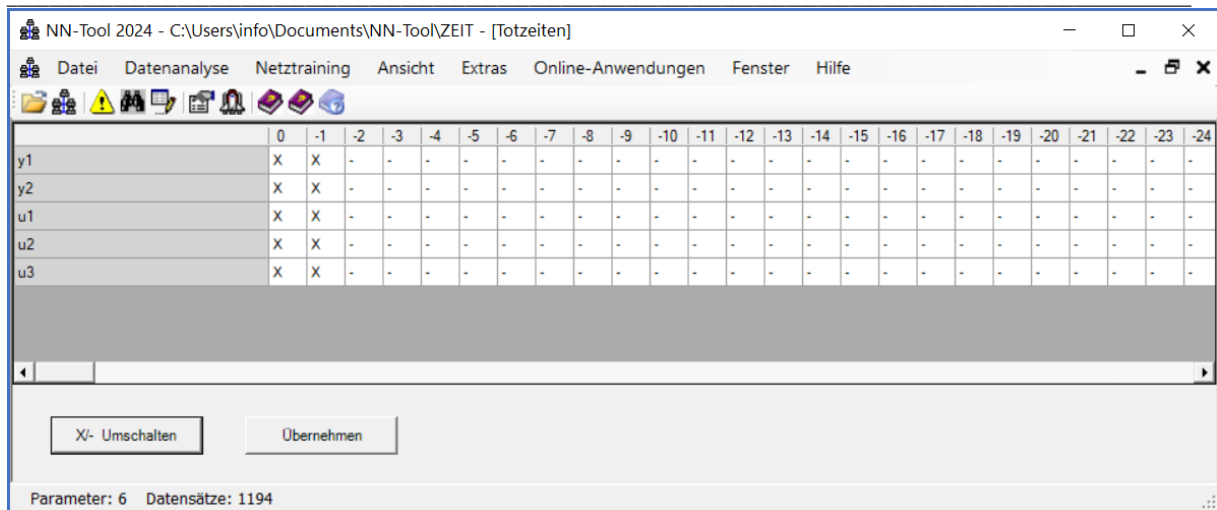
Parameter	Min	Max	Mittel	Stdabw	Median	vollständig	Transform.	E/A	A/P	verfügbar
1 Datum Uhrzeit	Kennung					1194	Lin	K	P	
2 y1	102,2447	908,9150	825,5398	85,1929	848,7028	1194	Lin	E	A	
3 y2	301,0150	544,4697	431,2293	30,1647	430,9498	1194	Lin	E	A	
4 u1	380,0169	500,4706	430,3818	28,9906	430,2661	1194	Lin	E	A	
5 u2	0,3924	1009,9077	595,2908	173,3952	544,0288	1194	Lin	E	A	
6 u3	50,0917	359,6131	227,7550	85,4305	237,8524	1194	Lin	E	A	

Bei diesem Problem besteht ein Zusammenhang zwischen den Größen y1 und y2 zum jetzigen Zeitpunkt mit den Größen y1, y2, u1, u2, u3 einen Zeitschritt in der Vergangenheit. Die Größen wurden alle 10 Minuten gemessen, die Messzeitpunkte stehen in der Spalte **Datum Uhrzeit**. Es müssen also nun die richtigen Totzeiten ins Spiel gebracht werden.

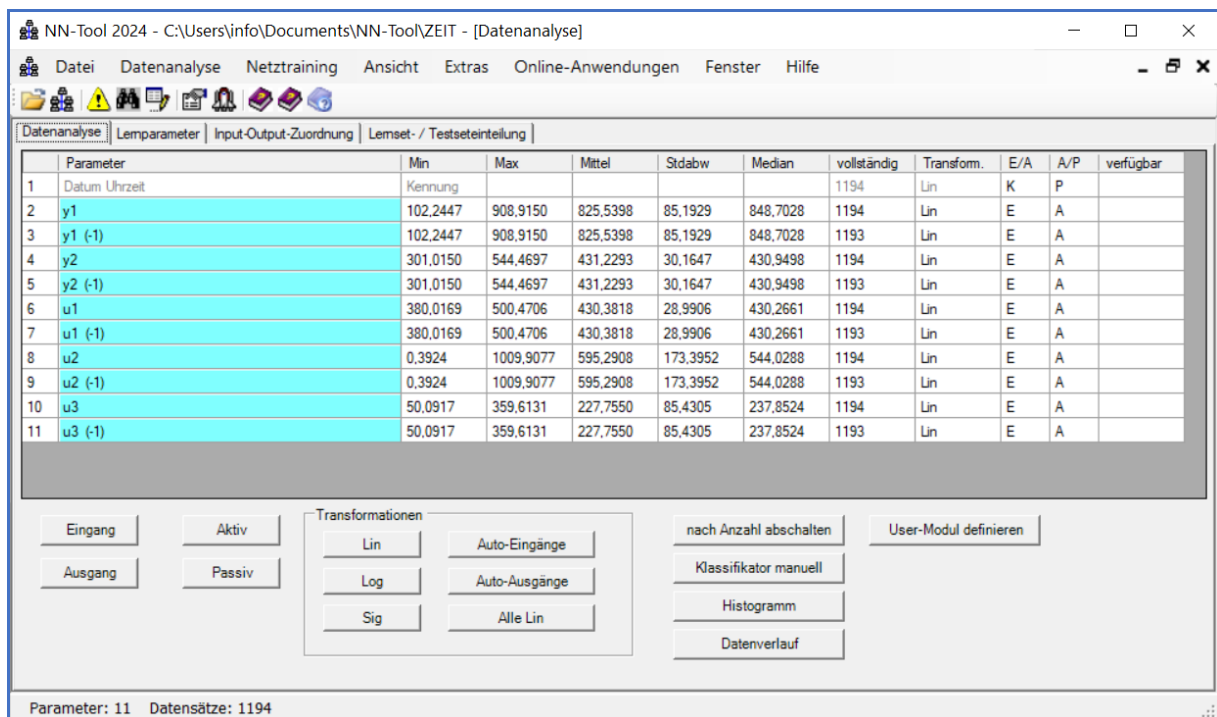
Führen Sie nun im Hauptmenu **Datenanalyse** den Menüpunkt **Totzeiten** aus:

	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14	-15	-16	-17	-18	-19	-20	-21	-22	-23	-24
y1	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
y2	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
u1	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
u2	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
u3	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Legen Sie durch X/- Umschalten die Zeittakte fest, an denen Sie die Größen einsetzen wollen. Im Beispiel werden die Größen y1 und y2 sowohl bei 0 als auch bei -1 benötigt, die Größen u1, u2, u3 nur bei -1. Aus bestimmten Gründen (um jederzeit die zeitversetzten Werte rekonstruieren zu können) benötigt NN-Tool jedoch sämtliche Größen **immer** auch beim Zeittakt 0. Wir wählen also zunächst alle Größen bei den Zeittakten 0 und -1 und setzen die nicht benötigten Werte später passiv. Legen Sie die Größen entsprechend fest. Anschließend sollte das Fenster wie folgt aussehen:



Mit **Übernehmen** werden die Einstellungen gültig und Sie kehren in das Fenster Datenanalyse zurück.



Alle Parameter treten nun doppelt auf, nämlich zum aktuellen Zeitwert sowie ein Zeittakt in der Vergangenheit (-1). Beachten Sie, daß die Größen zum Zeittakt (-1) seltener vorkommen, da sie am Anfangsdatensatz nicht bestimmt werden können. Setzen Sie die nicht benötigten Parameter u1, u2 und u3 zum Zeittakt 0 passiv. Setzen Sie die Größen y1 und y2 als Ausgangsgrößen. Sie sollten nun 5 Eingangsgrößen, nämlich sämtliche Parameter zum Zeittakt -1 sowie zwei Ausgangsgrößen haben.

Parameter	Min	Max	Mittel	Stdabw	Median	vollständig	Transform.	E/A	A/P	verfügbar
1 Datum Uhrzeit						1194	Lin	K	P	
2 y1	102,2447	908,9150	825,5398	85,1929	848,7028	1194	Lin	A	A	1193
3 y1 (-1)	102,2447	908,9150	825,5398	85,1929	848,7028	1193	Lin	E	A	
4 y2	301,0150	544,4697	431,2293	30,1647	430,9498	1194	Lin	A	A	1193
5 y2 (-1)	301,0150	544,4697	431,2293	30,1647	430,9498	1193	Lin	E	A	
6 u1	380,0169	500,4706	430,3818	28,9906	430,2661	1194	Lin	E	P	
7 u1 (-1)	380,0169	500,4706	430,3818	28,9906	430,2661	1193	Lin	E	A	
8 u2	0,3924	1009,9077	595,2908	173,3952	544,0288	1194	Lin	E	P	
9 u2 (-1)	0,3924	1009,9077	595,2908	173,3952	544,0288	1193	Lin	E	A	
10 u3	50,0917	359,6131	227,7550	85,4305	237,8524	1194	Lin	E	P	
11 u3 (-1)	50,0917	359,6131	227,7550	85,4305	237,8524	1193	Lin	E	A	

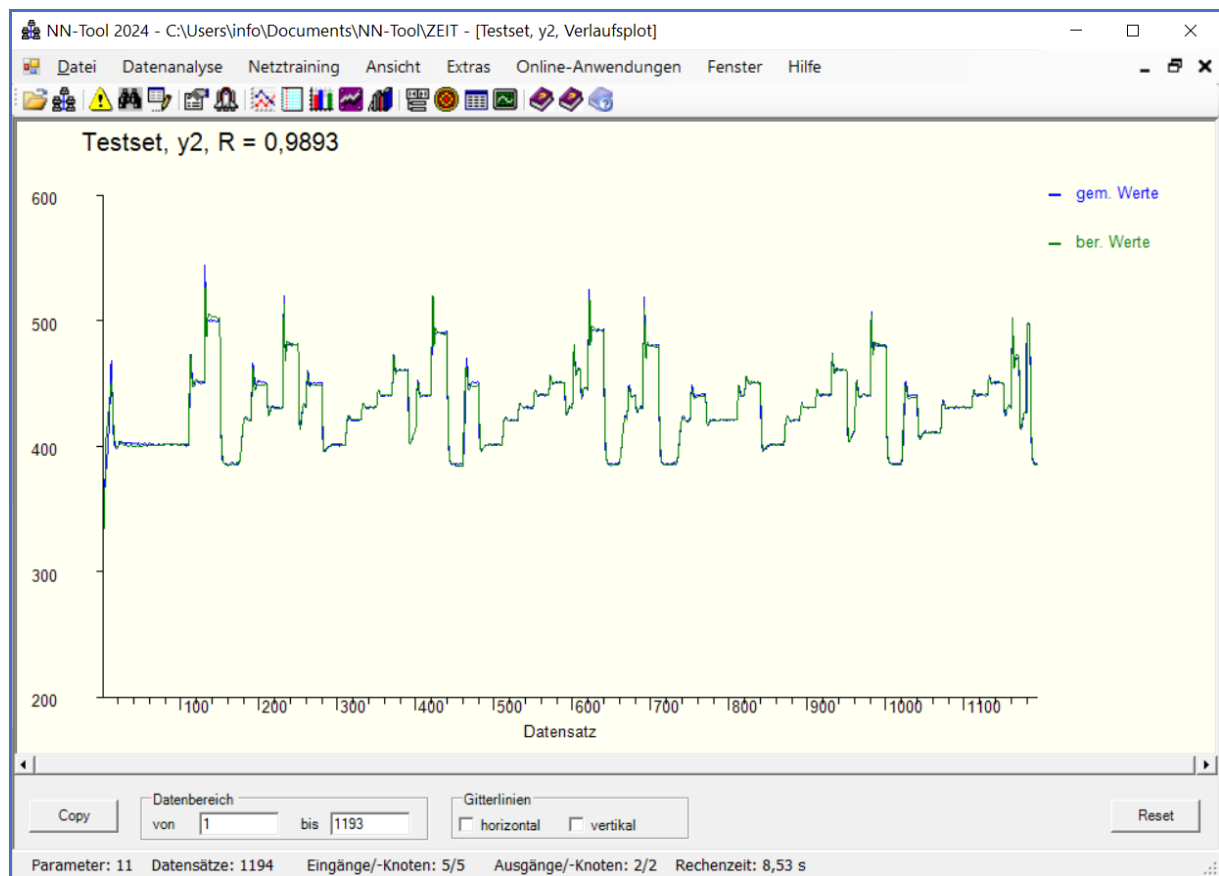
Parameter: 11 Datensätze: 1194 Eingänge/-Knoten: 5/5 Ausgänge/-Knoten: 2/2

Wechseln Sie nun auf die Registerkarte „Lernparameter“. Beachten Sie dabei, dass für Zeitreihen der Testset als zusammenhängend gewählt werden sollte. Dies hat NN-Tool hier auch als Default gesetzt. Alternativ kommt hier auch die noch stärkere, aber rechenintensivere Option „**Crossvalidation n-fach blockweise**“ in Betracht. Wählen Sie diese Option und starten Sie den Lernvorgang. Beachten Sie, dass jetzt jeder Ausgangsknoten 5 mal (für die 5 dynamisch erzeugten Testsets) durchlaufen wird:

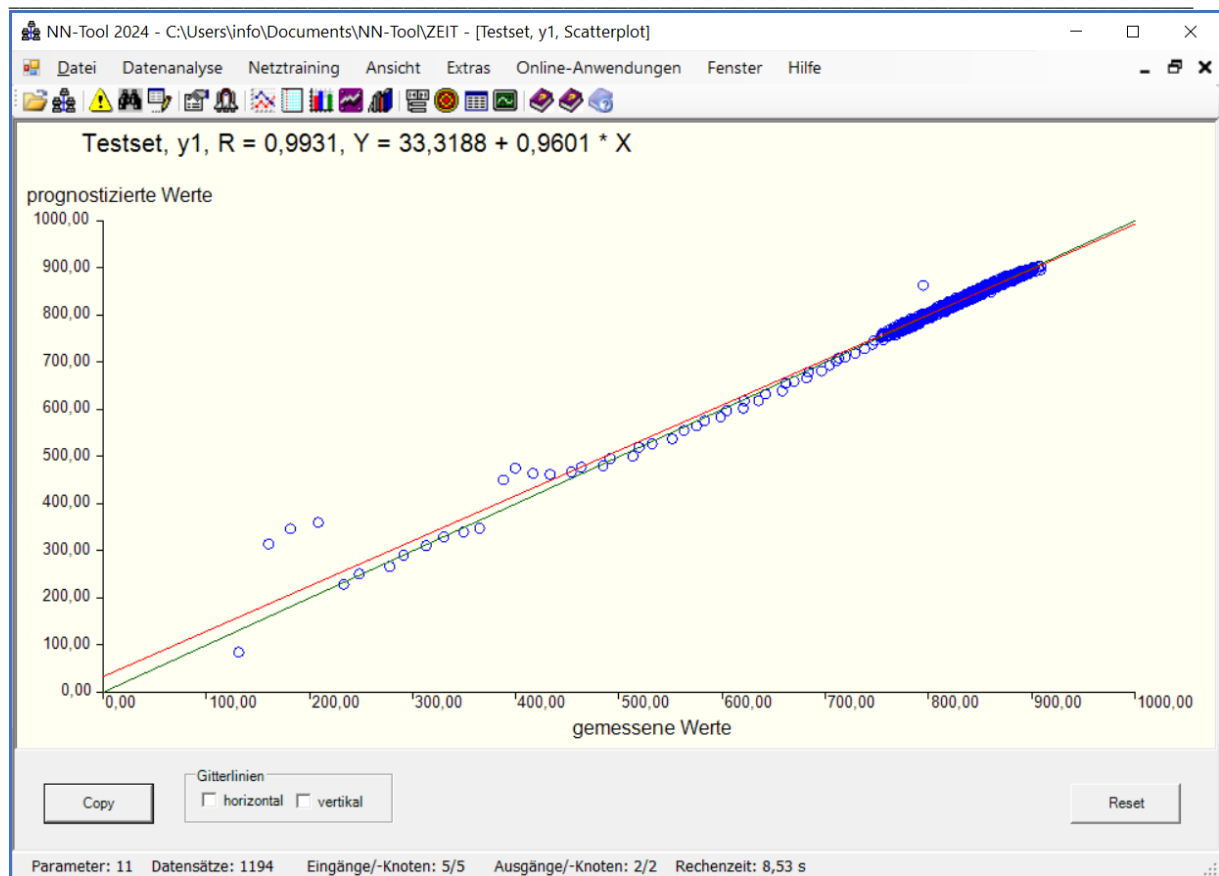
Testset	Eingänge	Ausgänge	Korrelation
2/3	1	30	92.3949
2/3	1	40	92.3607
2/3	1	50	92.3404
2/3	1	60	92.3241
2/3	1	70	92.3113
2/3	1	80	92.3015
2/3	1	90	92.2935
2/3	1	100	92.2869
2/3	2	10	98.8712
2/3	2	20	98.9540
2/3	2	30	99.0219
2/3	2	40	99.2565
2/3	2	50	99.3360
2/3	2	60	99.3853
2/3	2	70	99.4849
2/3	2	80	99.5428
2/3	2	90	99.6481
2/3	2	100	99.7209
2/3	4	10	99.7359
2/3	4	20	99.7538
2/3	4	30	99.7703
2/3	4	40	99.7799
2/3	4	50	99.7761
2/3	4	60	99.7767

Hier wird gerade der 2. Ausgang mit dem dritten Testset auf einem Netz mit vier inneren Knoten nach 60 Lernschritten getestet. Die Korrelation auf dem 3. Testset beträgt 99,7767%.

Bestimmen Sie zuletzt den Verlaufsplot auf dem Testset. Der Plot sollte wie folgt aussehen:



Interessant ist auch der folgende Scatterplot:



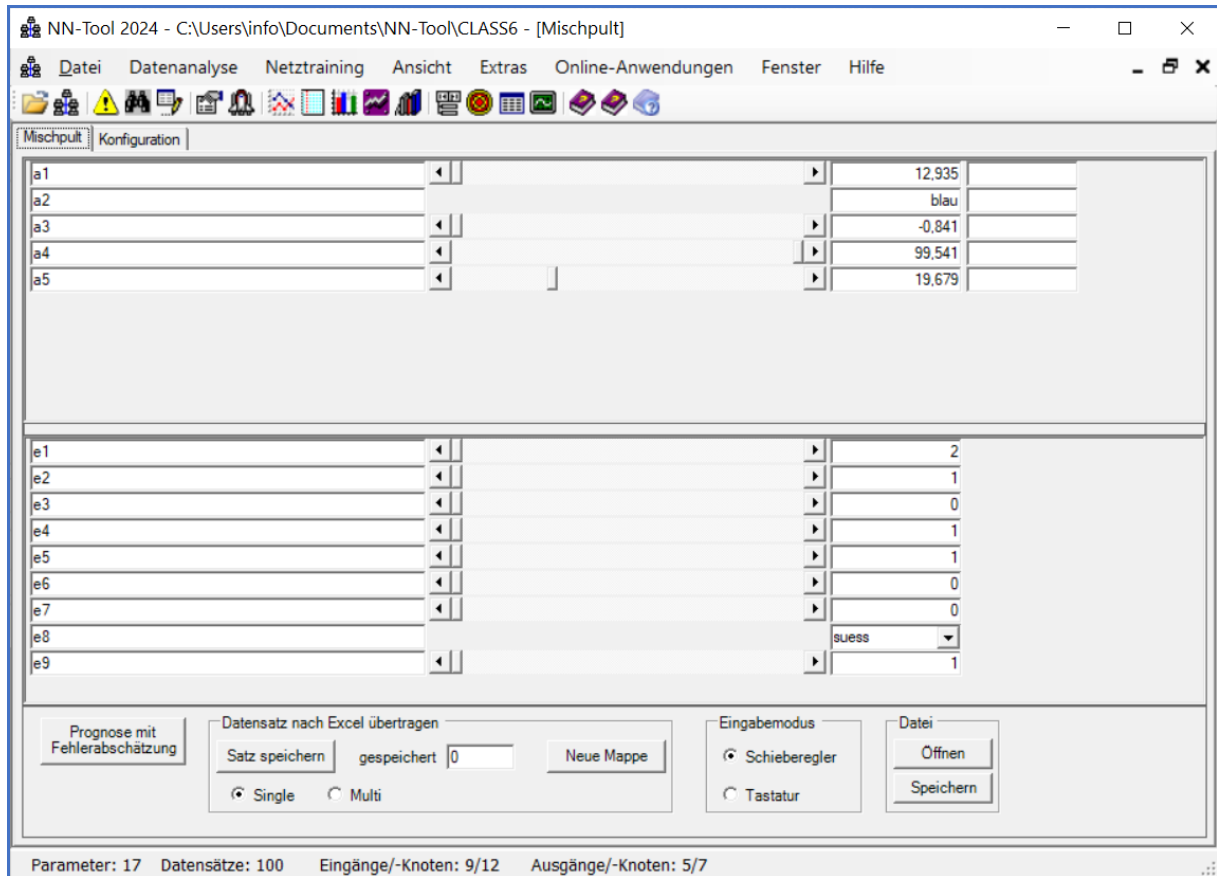
**Frage:** Wie kommt die Schwingung bei den niedrigen Werten zustande, die im entsprechenden Scatterplot auf dem Lernset nicht zu sehen ist? Antwort in Anhang „Dynamische Simulation“.

**Hinweis:** Bei größeren Datensätzen empfiehlt sich das Netztraining mit der Option „Crossvalidation n-fach blockweise“ im **Multithreading-Mode**.

Nach dieser Einführung in die Modellerstellung mit NN-Tool, werden im Folgenden die beiden **Anwendungsmodule Mischpult und Optimierer** erläutert. Weitere Anwendungsmodule finden sich dann im Anhang.

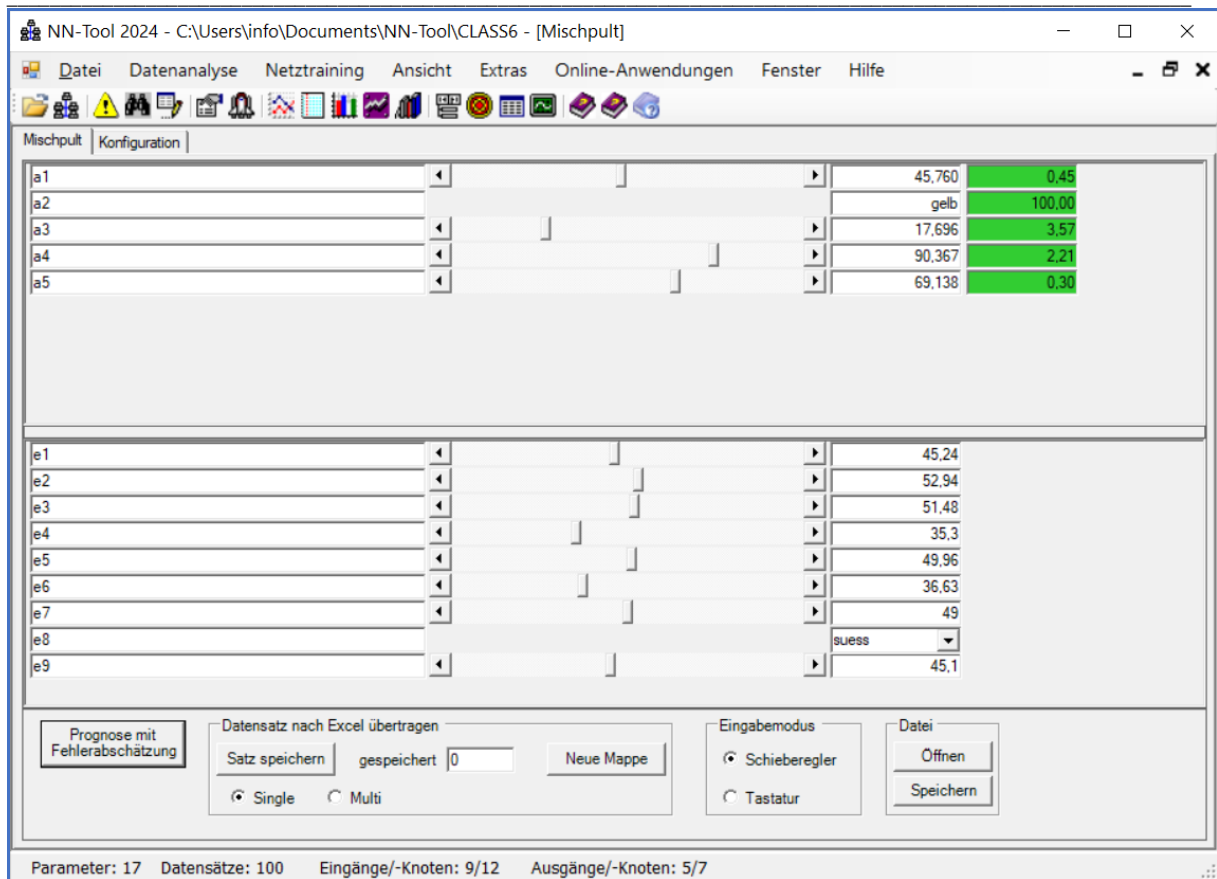
## 9. Anwendungsmodul Mischpult

Die Bedienung der NN-Tool Anwendungskomponente Mischpult wird am Beispiel der Anwendung **class6** durchgeführt (vgl. Kapitel 5, Kapitel 6). Starten Sie NN-Tool und laden Sie das erstellte Netz **class6**. Wählen Sie im Menu **Extras** den Menüpunkt **Mischpult**.



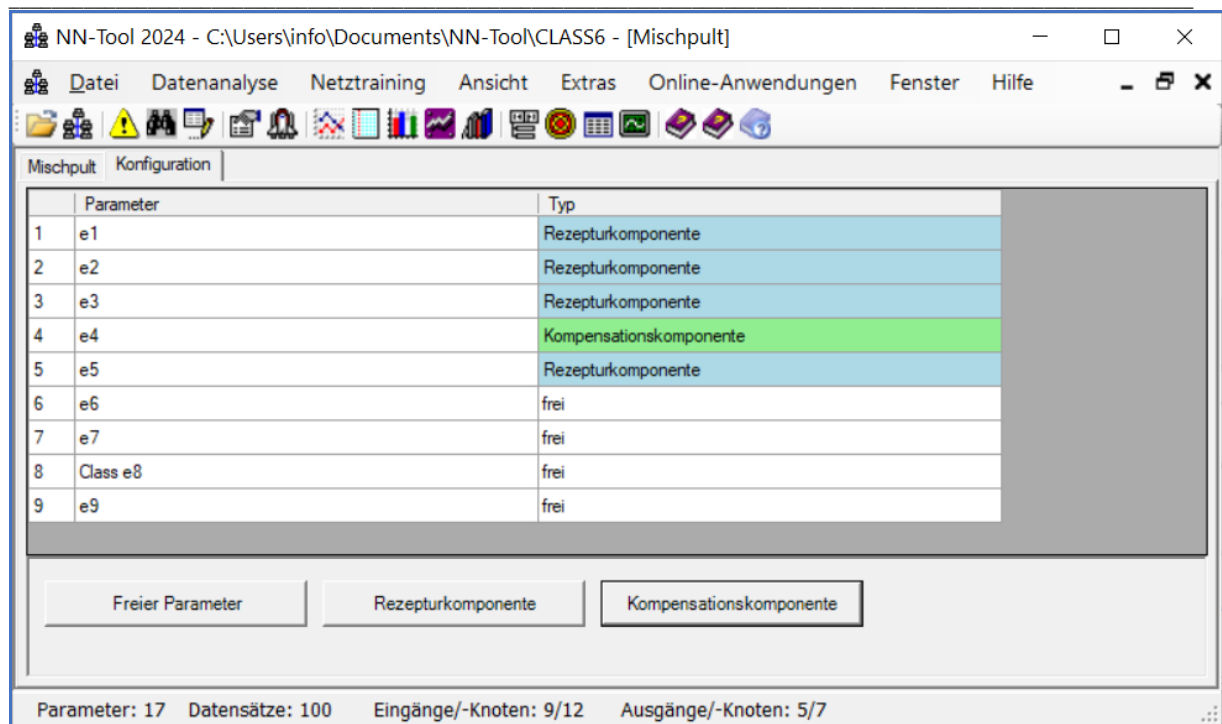
**Hinweis:** Die dargestellten Werte ergeben sich nach Training wie in Kapitel 6 beschrieben. Bei anderen Trainingsstrategien ergeben sich im Allgemeinen Abweichungen.

In dieser Anwendung können Prognosen neuer Datensätze einfach durch die Einstellungen der Schieberegler bzw. bei Klassifikatoren über die Auswahlboxen vorgenommen werden. Alternativ kann auf Tastatureingabe umgeschaltet werden. Die Einstellungen dienen als Vorgabe von Eingangsgrößen, die Berechnung der zugehörigen Ausgangsgrößen erfolgt automatisch (auch scrollend). Dabei kann zusätzlich eine Fehlerabschätzung angefordert werden. Drücken Sie die Taste **Prognose mit Fehlerabschätzung**. Es ergibt sich die folgende Darstellung:



Es werden Schätzwerte für die Abweichung eines Ausgangsparameters nach oben oder unten angezeigt. **Bei Klassifikatoren wird die Wahrscheinlichkeit für korrekte Klassifikation angezeigt (hier 100,00 %).** Zur Abschätzung des Fehlers werden bekannte Datensätze der Lernmenge herangezogen, die in der „Nähe“ des zu untersuchenden Datensatzes liegen. Die Farben zeigen die Verlässlichkeit der Fehlerabschätzung basierend auf der Anzahl der Datensätze in der Umgebung des betrachteten Datensatzes an (dunkel rot – schlecht,..., dunkel grün – gut).

Falls unter den Eingangsgrößen Rezepturkomponenten sind, müssen diese bekanntlich der **Rezepturnebenbedingung: „Summe der Konzentrationen der Komponenten = 100 %“** genügen. Statt die Werte immer aufzusummieren, können über das Registerblatt „Konfiguration“ die entsprechenden Komponenten definiert werden:



Im Beispiel wurden die Eingangsgrößen „e1“ bis „e5“ als Rezepturkomponenten festgelegt. Die Größen „e6“ bis „e9“ sind weiterhin frei veränderlich (z.B. Temperaturen und Drücke). Darüber hinaus wurde die Komponente „e4“ als „Kompensationskomponente“ festgelegt. Dies bedeutet, daß der Schieberegler für „e4“ nicht mehr von Hand bedient wird, sondern von NN-Tool immer so nachgestellt wird, dass die Rezepturnebenbedingung erfüllt ist (falls möglich!). Nach Wechsel auf das Registerblatt „Mischpult“ und Verwendung der Schieberegler ergibt sich das folgende Bild:



NN-Tool 2024 - C:\Users\info\Documents\NN-Tool\CLASS6 - [Mischpult]

Datei Datenanalyse Netztraining Ansicht Extras Online-Anwendungen Fenster Hilfe

Mischpult | Konfiguration

a1			23,561	
a2			rot	
a3			4,252	
a4			97,367	
a5			59,219	

e1			8,58	
e2			19,62	
e3			17,82	
e4			29,94	
e5			24,04	
e6			15,84	
e7			39,2	
e8			suess	
e9			35,3	

Prognose mit Fehlerabschätzung

Prüfsumme 100,00

Datensatz nach Excel übertragen

Satz speichern gespeichert 0 Neue Mappe

Single Multi

Eingabemodus

☒ Schieberegler ☐ Tastatur

Datei

Öffnen Speichern

Parameter: 17 Datensätze: 100 Eingänge/-Knoten: 9/12 Ausgänge/-Knoten: 5/7

Die farbig markierten Rezepturkomponenten erfüllen jetzt die Rezepturnebenbedingung, was auch durch die Prüfsumme bestätigt wird. Mit Hilfe der Tasten „Speichern“ und „Öffnen“ können die Konfigurationen in .mix-Dateien abgespeichert bzw. geladen werden. Mittels der Taste „Satz speichern“ können die Ergebnisse nach Excel übertragen werden.

## 10. Komplexe Optimierungsprobleme / Rezepturoptimierungen

### 10a) Anwendungsbereich des Optimierers

NN-Tool stellt unter dem Menu Extras eine Umgebung zur Definition und Lösung komplexer Optimierungsprobleme zur Verfügung. Der Optimierer ermöglicht die Berechnung **optimaler Rezepturen** (Mischungen) und **Betriebspunkte** auf der Basis eines mit NN-Tool erstellten neuronalen Netzes.<sup>1</sup> Bei der Optimierung kann eine Vielzahl von Nebenbedingungen berücksichtigt werden. Für die Eingangsgrößen gilt im Einzelnen:

- Für jede Eingangsgröße kann festgelegt werden, ob sie mitoptimiert oder auf ihrem Startwert festgehalten wird. Bei Optimierung können obere und untere Grenzen festgelegt werden.
- Die Eingangsgrößen können beliebigen Komponentengruppen (z.B. Gruppe der Katalysatoren, der Stabilisatoren, Flammhemmer etc.) zugeordnet werden. Für jede derartige Gruppe können obere und untere Grenzen sowie die Maximalzahl der eingesetzten Komponenten vorgegeben werden. Beispiel: von den 20 verfügbaren Katalysatoren sollen höchstens 3 mit einer Gesamtkonzentration zwischen 2 und 5 Prozent eingesetzt werden. Gleichzeitig soll genau ein Flammenschutzmittel mit einer Konzentration von 5 % eingesetzt werden.
- Neben den Rezepturkomponenten können weitere kontinuierliche Größen z.B. Druck, Temperatur, Verweilzeit mitberücksichtigt werden, d.h. diese Größen können entweder mitoptimiert oder auf Vorgabewerten gehalten werden.
- Als weitere Eingangsgrößen können zusätzliche Klassifikatoren mitoptimiert werden. Es kann zum Beispiel die optimale Rezeptur zusammen mit dem geeignetsten Rührertyp ermittelt werden.

Bei den Ausgangsgrößen bestehen die Optionen:

- Es können genaue Zielwerte angestrebt, oder nur das Über- bzw. Unterschreiten bestimmter Qualitätsmerkmale verhindert werden. Darüber hinaus können zulässige Bandbreiten vorgegeben werden.
- Klassifikatoren können mitoptimiert werden: „berechne farbneutrale Mischung mit den folgenden weiteren Eigenschaften...“
- Rezepturkosten können mitoptimiert werden.

Es stehen drei Lösungsstrategien zur Verfügung, die dem System unterschiedliche „Kreativität“ ermöglichen:

- Das System erstellt die zu berechnende Rezeptur von Grund auf (maximale Kreativität).
- Das System geht von bestehenden Rezepturen aus, kann aber eingesetzte Komponenten gegen andere austauschen.

---

<sup>1</sup> Der eigentliche Optimierer ist selbständig lauffähig und kann somit auch außerhalb von NN-Tool eingesetzt werden (nach entsprechender Lizenzierung). Er ist in erster Linie zur Einbindung in Endanwenderapplikationen (insbesondere Datenbanken) konzipiert und verfügt über keine nennenswerte Benutzeroberfläche. Da das Programm über Eingabedateien gesteuert wird und die Ergebnisse in eine Ausgabedatei schreibt, sollte eine Integration in beliebige Umgebungen (z.B. Oracle Datenbanken) problemlos möglich sein.

- Das System bestimmt die optimale bestehende Rezeptur und variiert nur die Konzentrationen, d.h. es tauscht insbesondere keine Komponenten aus (minimale Kreativität).

**Auf die Berücksichtigung von Rezepturkomponenten kann vollständig verzichtet werden. In diesem Fall liefert das System eine Berechnung optimaler Betriebspunkte.**

### 10b) Definition von Optimierungsproblemen

Die Vorgehensweise zur Definition und Lösung von Optimierungsproblemen wird am Beispiel einer Rezepturoptimierung mit der Anwendung **class6** durchgeführt (vgl. Kapitel 5, Kapitel 6). Die Vorgehensweise bei einer Verfahrens- oder Betriebspunktoptimierung verläuft völlig analog mit der Vereinfachung, dass auf die Berücksichtigung von Rezepturgruppen verzichtet werden kann. Starten Sie NN-Tool und laden Sie das erstellte Netz **class6**. Wählen Sie im Menu **Extras** den Menüpunkt **Optimierung**.

NN-Tool 2024 - C:\Users\info\Documents\NN-Tool\CLASS6 - [Optimierer]

Datei Datenanalyse Netztraining Ansicht Extras Online-Anwendungen Fenster Hilfe

Übersicht Rezepturgruppen Freie Parameter Zielwerte Kosten Ergebnisse Optionen

**Rezepturgruppen**

Anzahl  zugeordnete Komponenten  Gesamtkonzentration

**sonstige Eingangsparameter**

Anzahl  davon Klassifikatoren

**Zielwerte**

Lfd. Nr.	Parameter	Wert	Gewichtung

**Kosten mitoptimieren**

☐ ja ☒ nein

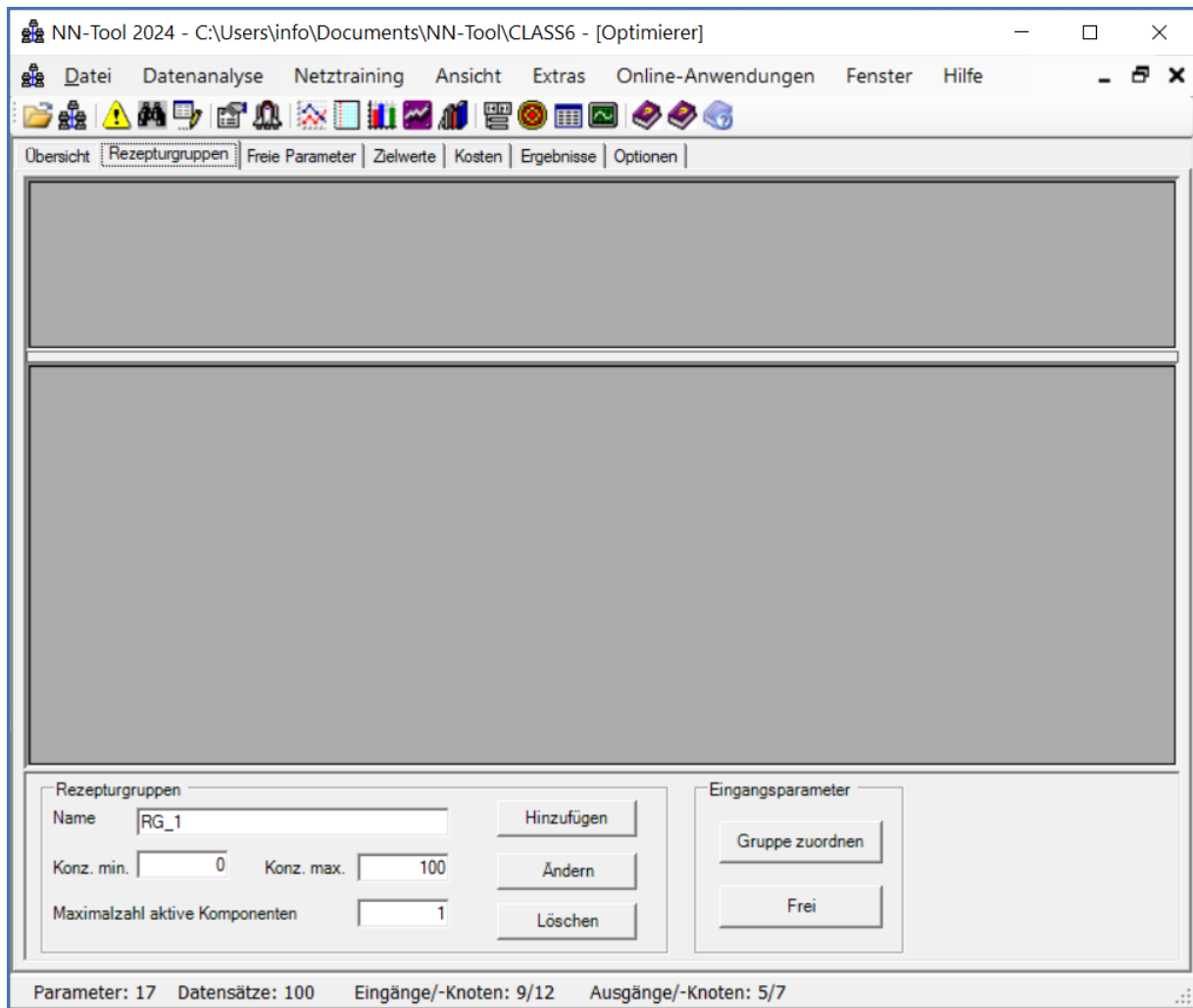
Kostendatei

Gewichtung

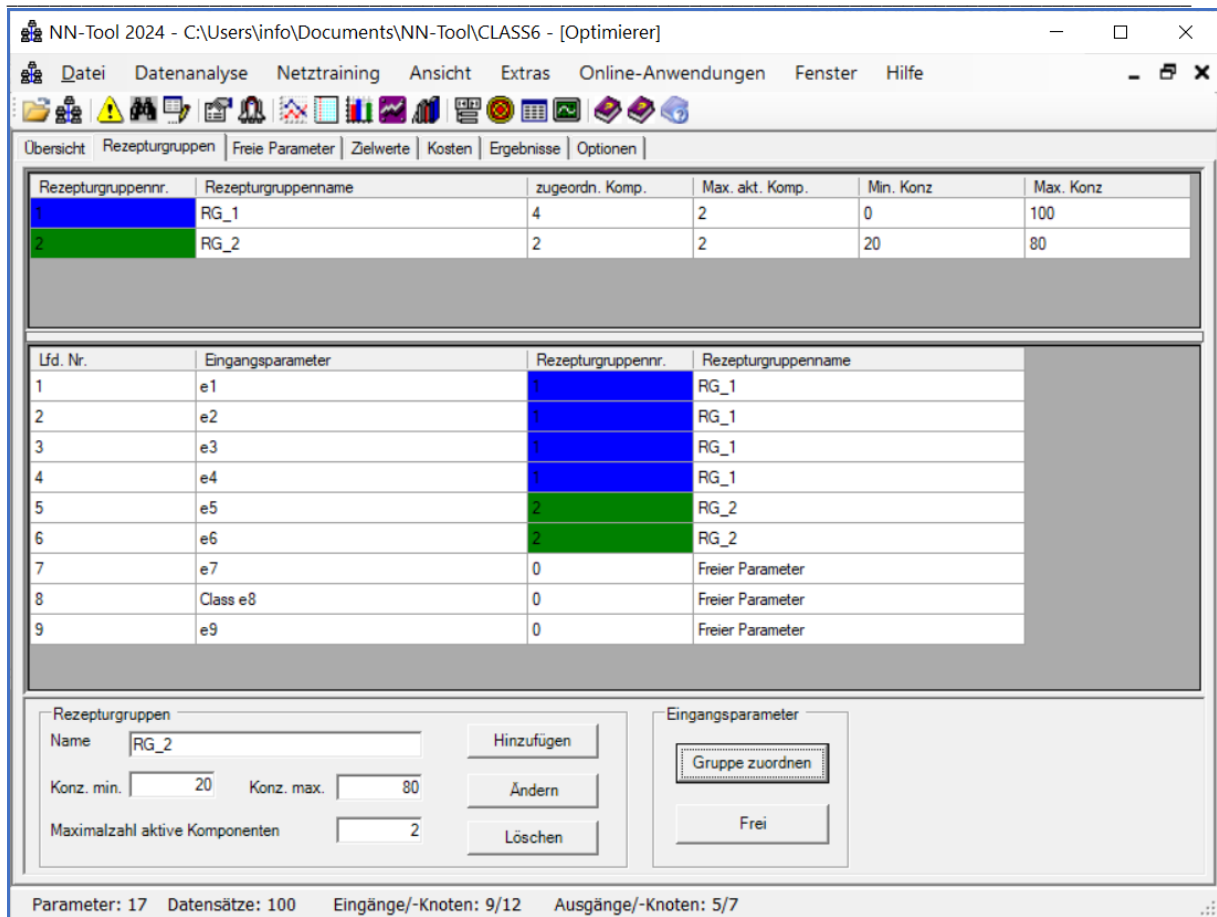
Speichern Laden Optimierung starten

Parameter: 17 Datensätze: 100 Eingänge/-Knoten: 9/12 Ausgänge/-Knoten: 5/7

Zunächst müssen wir die zu berücksichtigenden Rezepturgruppen festlegen. Wechseln Sie auf die Registerkarte „**Rezepturgruppen**“.



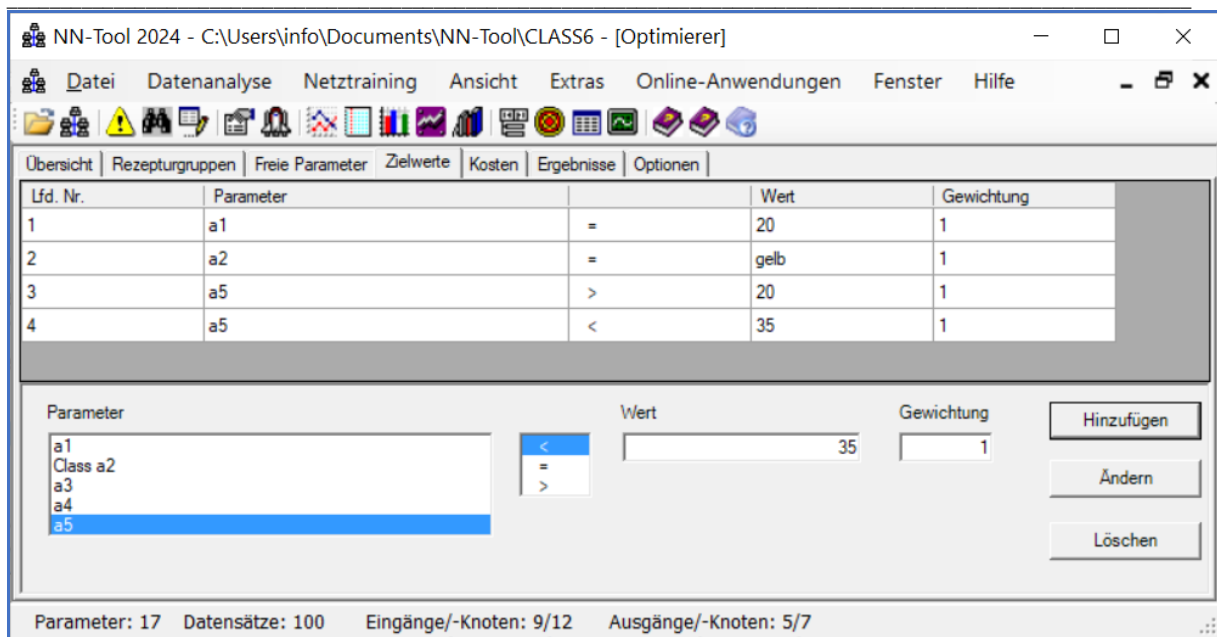
Definieren Sie zwei Rezepturgruppen (RG\_1 und RG\_2). Der Gruppe „RG\_1“ sind die Parameter e1, e2, e3 und e4 zuzuordnen. Der Gruppe „RG\_2“ gehören die Parameter e5 und e6 an (erst die Gruppen definieren, dann die Komponenten der Gruppe zuordnen). Die Gruppe „RG\_1“ soll über den kompletten Bereich von 0 bis 100 Prozent variieren dürfen, aber maximal 2 aktive Parameter (d.h. Komponenten ungleich 0) aufweisen. Dies bedeutet für die Gruppe „RG\_1“, dass von den 4 zugeordneten Komponenten maximal 2 in der berechneten Rezeptur auftauchen dürfen. Die Gruppe „RG\_2“ soll von 20 bis 80% bei 2 aktiven Parametern variieren dürfen. Zuletzt sollte das Fenster wie folgt aussehen:



Wechseln Sie zur Registerkarte „**Übersicht**“. Die Registerkarte zeigt Ihnen an, dass 2 Rezepturgruppen definiert wurden, denen 6 Komponenten angehören. Darüber hinaus gibt es noch 3 sonstige Parameter (e7, e9 sowie der Klassifikator e8). Über die Registerkarte „**Freie Parameter**“ können für die sonstigen Parameter Grenzen definiert werden (z.B. bei Verfahrensoptimierungen). Als Voreinstellungen für die Grenzen werden die Variationsbereiche der Parameter verwendet. Belassen Sie es bei diesen Einstellungen. Spezifizieren Sie nun mittels der Registerkarte „**Zielwerte**“ die folgenden Vorgaben für die Ausgangsparameter:

- a1 = 20
- a2 = „gelb“
- a5 > 20
- a5 < 35

Die Karte „**Zielwerte**“ sollte anschließend wie folgt aussehen:



Durch Einstellung der Gewichtung (positive Zahl!) kann der unterschiedlichen Bedeutung der einzelnen Optimierungsziele Rechnung getragen werden.

Da bei der Rezepturoptimierung die Kosten berücksichtigt werden sollen (unterschiedliche Komponenten verursachen unterschiedliche Kosten), ist zunächst auf die Registerkarte „Kosten“ zu wechseln und eine Kostendatei zu erstellen. Definieren Sie die folgenden Kosten:

- e1 = 10
- e2 = 5
- e3 = 20
- e5 = 30
- Rest 0

Dies bedeutet, dass jedes % der Komponente e1 mit 10 Kosteneinheiten bewertet wird etc. Speichern Sie diese Kosten in der Datei class6.cos ab. Die Registerkarte sollte nun wie folgt aussehen:

Parameter	Kosten
e1	10
e2	5
e3	20
e4	0
e5	30
e6	0
e7	0
e8 suess	0
e8 sauer	0
e8 salzig	0
e8 bitter	0
e9	0

Kostendatei speichern    Kostendatei laden    C:\Users\info\Documents\NN-Tool\CLASS6.cos

Parameter: 17    Datensätze: 100    Eingänge/-Knoten: 9/12    Ausgänge/-Knoten: 5/7

Wechseln Sie nun zur Registerkarte „**Übersicht**“. Geben Sie für die Kosten den Zielwert 2200 mit der Einstellung „<“ und Gewichtung 1 vor und setzen Sie Kosten mitoptimieren auf „ja“.

Die Registerkarte „Übersicht“ sollte nun wie folgt aussehen:

NN-Tool 2024 - C:\Users\info\Documents\NN-Tool\CLASS6 - [Optimierer]

Datei Datenanalyse Netztraining Ansicht Extras Online-Anwendungen Fenster Hilfe

Übersicht Rezepturgruppen Freie Parameter Zielwerte Kosten Ergebnisse Optionen

Rezepturgruppen

Anzahl  zugeordnete Komponenten  Gesamtkonzentration

sonstige Eingangsparameter

Anzahl  davon Klassifikatoren

Zielwerte

Lfd. Nr.	Parameter		Wert	Gewichtung
1	a1	=	20	1
2	a2	=	gelb	1
3	a5	>	20	1
4	a5	<	35	1

Kosten mitoptimieren

☒ ja Kostendatei

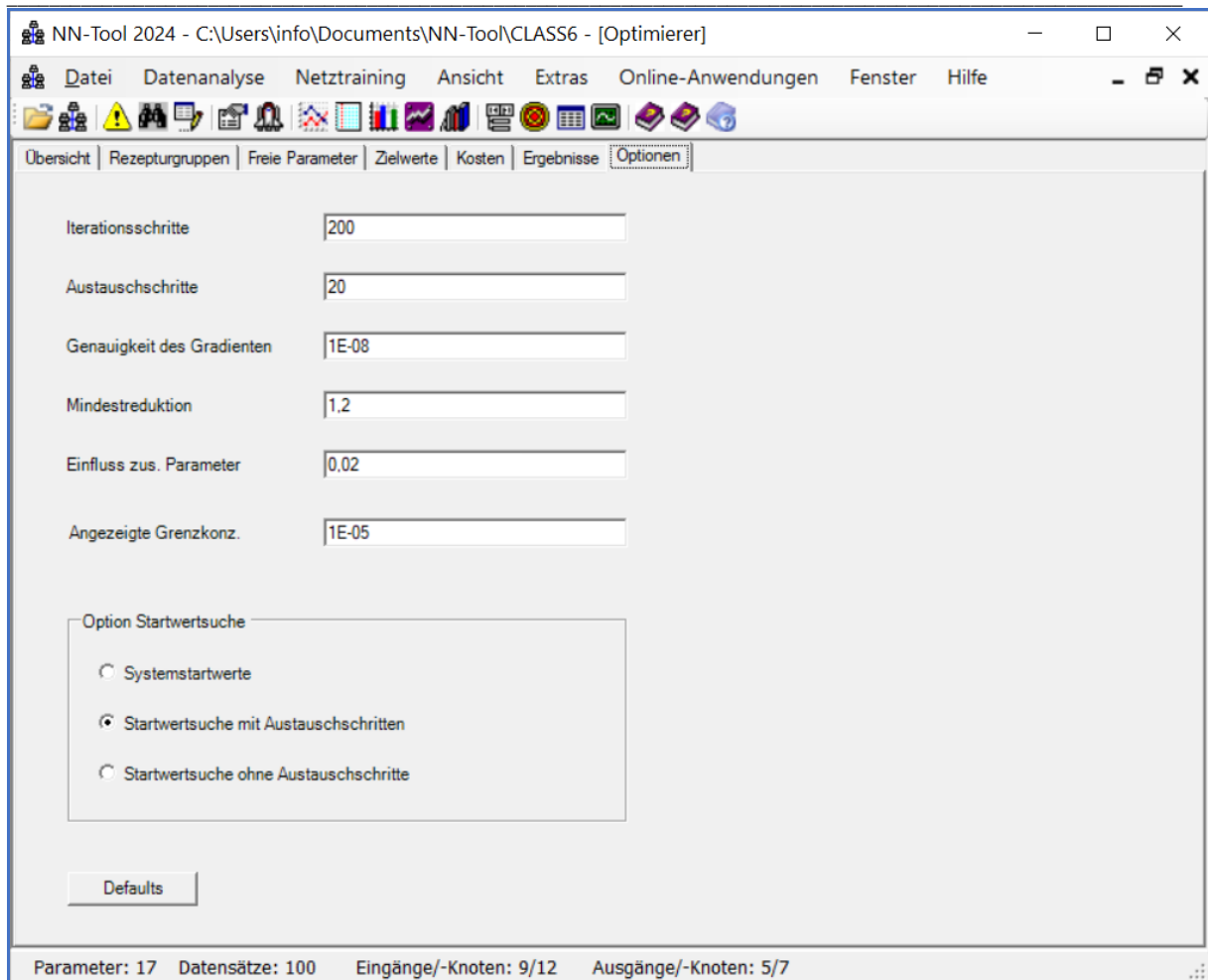
☐ nein  Gewichtung

Parameter: 17 Datensätze: 100 Eingänge/-Knoten: 9/12 Ausgänge/-Knoten: 5/7

Über die Registerkarte **Optionen** können die numerischen Parameter der Optimierung beeinflusst werden. Das Rechenverfahren besteht prinzipiell aus drei Phasen:

1. Ermittlung einer Anfangsrezeptur, die alle Nebenbedingungen erfüllt. Dabei kann von den Systemstartwerten ausgegangen werden (1. Option Startwertsuche) oder es wird die Lerndatendatei nach geeigneten Ausgangsrezepturen durchsucht (Optionen Startwertsuche 2 oder 3).
2. Optimierung der Konzentrationen der Rezeptur mittels eines angepaßten numerischen Verfahrens (auf die Nebenbedingungen projiziertes, asymptotisches CG-Verfahren). Dabei werden sogenannte Iterationsschritte (Gradientenschritte) zur Minimierung der sogenannten Zielfunktion ausgeführt. Je niedriger der Wert der Zielfunktion, desto genauer werden die Vorgaben eingehalten. Typische Anzahl ca. 200.
3. Austauschschritte: Die in der Rezeptur eingesetzten Komponenten werden gegen geeignetere ausgetauscht. Dieser Schritt wird bei Option „Startwertsuche ohne Austauschschritte“ nicht ausgeführt.
4. Die Schritte 2. und 3. werden im Wechsel ausgeführt (außer bei Startwertoption 3).





Iterationsschritte gibt die maximale Anzahl der Gradientenschritte zwischen den Austauschschritten vor. Während der Gradientenschritte werden nur die Konzentrationen der eingesetzten Komponenten verändert. Empfohlener Wertebereich: 50-500 (höherer Wert - größere Rechenzeit, bessere Ergebnisse)

Austauschschritte, d.h. Anzahl der Operationen bei denen die eingesetzten Komponenten verändert werden.

Empfohlener Wertebereich: 5-50

(höherer Wert - größere Rechenzeit, bessere Ergebnisse)

Genauigkeit des Gradienten. Der Gradient wird durch numerische Differenzenbildung ermittelt. Dies ist um so genauer, je kleiner der Wert ist. Zu kleine Werte führen jedoch zu numerischen Instabilitäten.

Empfohlener Wert: 1.0e-8

Mindestreduktion. Geforderter Reduktionsfaktor der Zielfunktion durch Gradientenschritte. Der Algorithmus bricht ab, wenn nach einem Austauschschritt durch die anschließenden Gradientenschritte keine hinreichende Reduktion der Zielfunktion mehr erreicht werden konnte. Ein Wert von 1,2 bedeutet, daß die Zielfunktion um mindestens den Faktor 1,2 reduziert werden muß, ansonsten soll der Algorithmus abbrechen. Der Wert muß größer 1 sein.

Empfohlener Wertebereich: 1,01 – 2,0

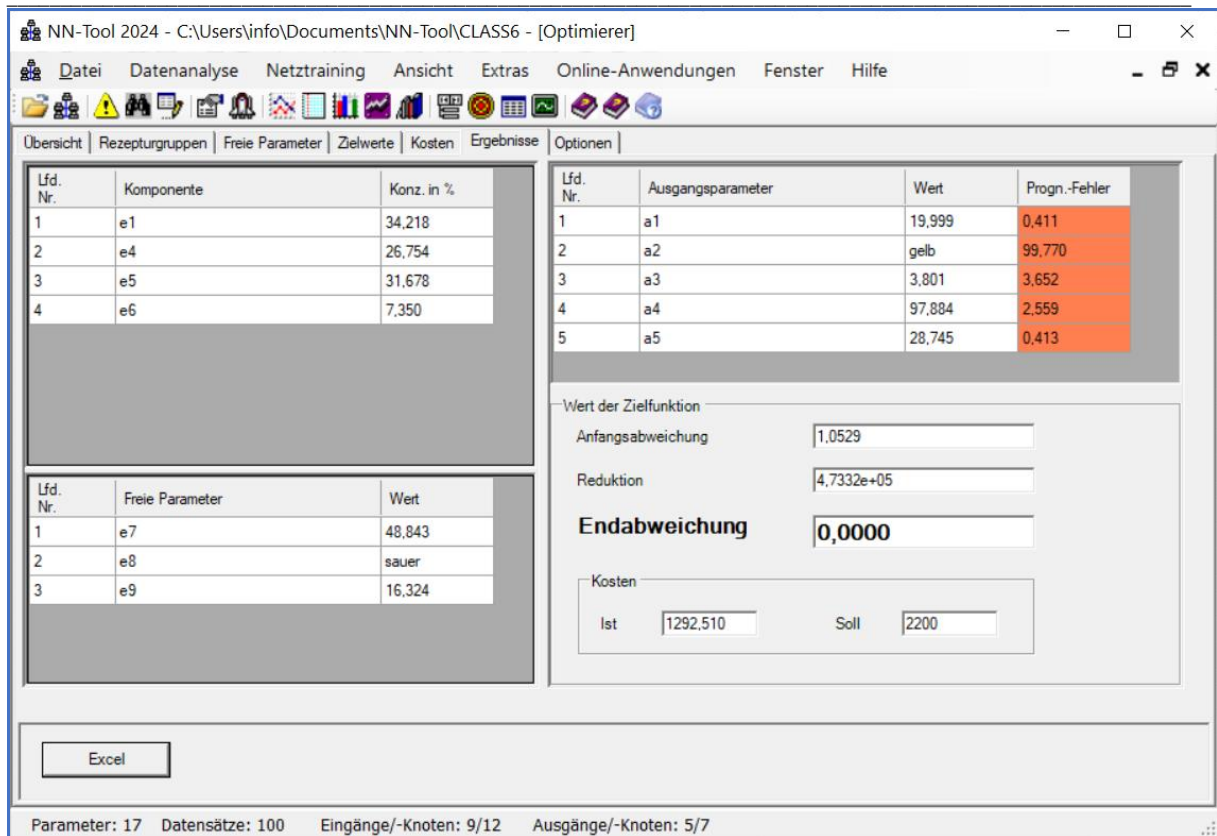
(höherer Wert - kürzere Rechenzeit, schwächere Ergebnisse)

Einfluß zusätzlicher Parameter. Gewichtung der zusätzlichen kontinuierlichen Variablen. Falls neben reinen Rezepturkomponenten, die ja als Konzentrationen nur Werte zwischen 0 und 100 (Angabe in %) annehmen können, zusätzliche kontinuierliche Variable optimiert werden, müssen diese geeignet transformiert (d.h. in ihrem Betrag skaliert) werden, damit sie in die gleiche Größenordnung wie die Komponenten kommen. Der maximal mögliche Wert der transformierten Variable entspricht dem Kehrwert des angegebenen Parameters. Ein Parameter von 0,02 bedeutet, dass die transformierte Variable, unabhängig von ihrem ursprünglichen physikalischen Wert, während der Optimierung nur Werte zwischen 0 und 50 ( $=1/0,02$ ), bzw. -50 und 50 bei Variablen mit unterschiedlichem Vorzeichen, annehmen kann. Im Ergebnis wird die Variable natürlich zurücktransformiert.  
Empfohlener Wertebereich: 0,01 – 0,2  
(kein eindeutiger Einfluß auf die Rechenzeit)

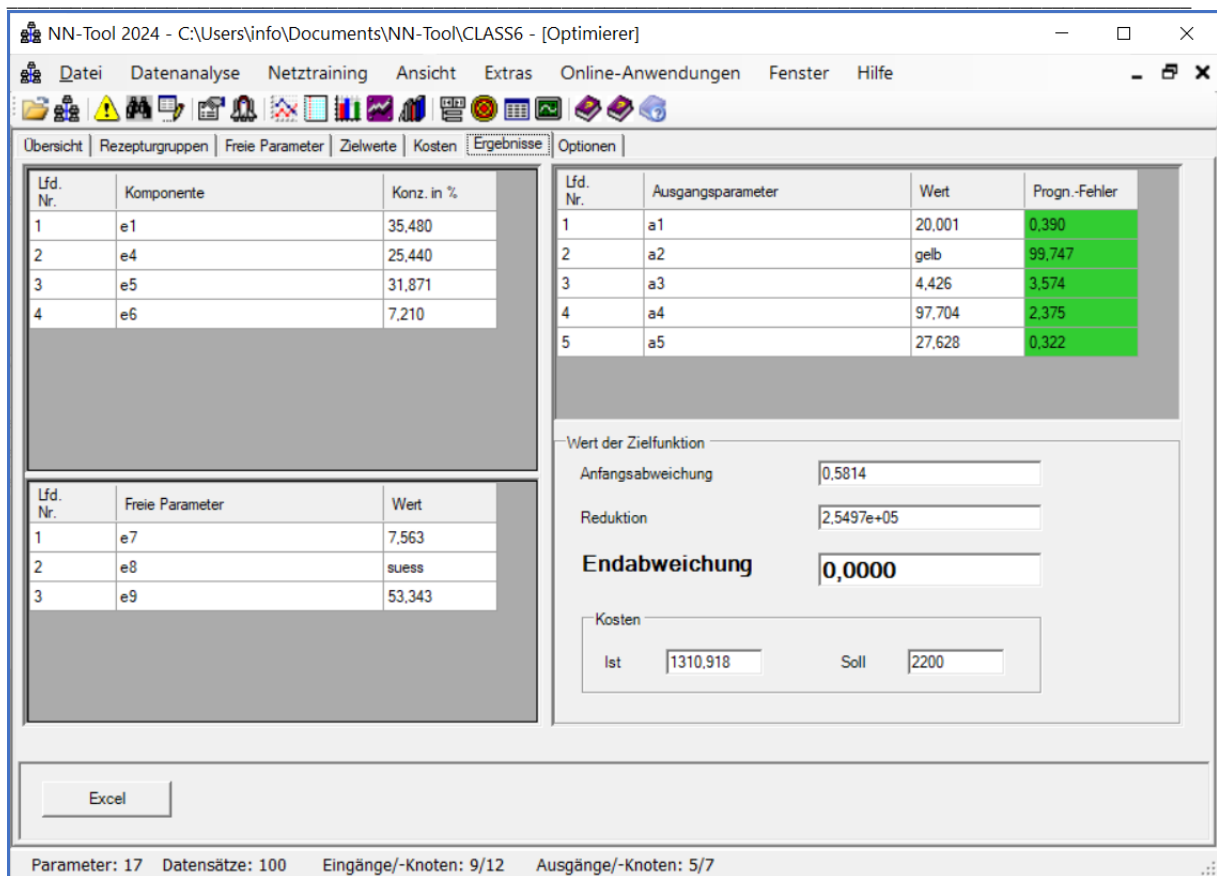
Option Startwertsuche. Es bestehen drei Möglichkeiten zur Festlegung dieser Steuergröße:

1. Der Algorithmus verwendet den in der Datei `nnoptin.dat` mitgegebenen Startvektor. Die Komponenten können ausgetauscht werden.
2. Der Algorithmus sucht in der Datei `<Anwendung>.ptp` eine geeignete Startrezeptur, die möglichst gut zu den Nebenbedingungen und den Zielwerten paßt. Während der Optimierung können die Komponenten ausgetauscht werden.
3. Der Algorithmus sucht in der Datei `<Anwendung>.ptp` eine geeignete Startrezeptur, die möglichst gut zu den Nebenbedingungen und den Zielwerten paßt. Während der Optimierung werden die Komponenten festgehalten. Damit kann das System nur bereits bekannte Rezepturen in den Konzentrationen variieren, aber keine neuen Komponenten einfügen.

**Belassen Sie es bei den Defaulteinstellungen.** Kehren Sie zur Registerkarte „Übersicht“ zurück und speichern Sie sämtliche Einstellung in der Datei **class6.opti**. Drücken Sie nun die Taste **Optimierung starten**. Die Optimierung läuft als eigenständige Anwendung (C-Programm), auch unabhängig von NN-Tool. **Nach Ablauf der Optimierung** wird das folgende Fenster angezeigt (**Hinweis:** Die dargestellten Werte ergeben sich für ein Netz nach dem Netztraining wie in Kapitel 6 beschrieben. Bei Netzen mit anderen Trainingsstrategien ergeben sich im Allgemeinen auch bei der darauf aufbauenden Optimierung Abweichungen):



Links oben stehen die berechneten Werte der eingesetzten Rezepturkomponenten (nur für diese gilt Summe = 100), darunter die Werte für die sonstigen Parameter. Im angegebenen Beispiel konnten die Vorgabewerte für die Ausgangsparameter sowie die Kosten nahezu exakt erreicht werden (für die Kosten galt natürlich  $< 2200$ ). Darauf weisen der Wert 0,0000 für die verbleibende Endabweichung der Zielfunktion sowie der Wert der Reduktion hin. Der Wert Reduktion gibt an, wie stark der ursprüngliche Wert der Zielfunktion reduziert worden ist. Die **rötliche Farbe beim Prognosefehler** weist darauf hin, dass die berechnete Rezeptur relativ weit von Datensätzen aus der Trainingsmenge entfernt liegt, und sich die Fehlerabschätzung daher nur auf wenige benachbarte Beispieldatensätze abstützt. Dies liegt hier vor allem daran, dass der Klassifikator e8 bei der Optimierung als variabel gesetzt wurde, die berechnete Klasse e8 „sauer“ aber in den Trainingsdaten schwächer vertreten ist. Wird die Klasse e8 bei der Optimierung auf die häufigste Klasse „suess“ fixiert, ergibt sich unter sonst gleichen Bedingungen:



Diese Lösung liegt offensichtlich näher an den Datensätzen der Trainingsmenge (Progn.-Fehler grün unterlegt). Aber man kann ja auch mal etwas völlig Neues ausprobieren.

Mittels der Taste **Excel** kann die berechnete Lösung nach Excel übertragen werden.

**Zum Abschluß:** Bitte teilen Sie mir aufgetretene Fehler, auch wenn es sich „nur“ um Bedienfehler handelt, möglichst unverzüglich mit.

**E-Mail:** [info@baermann.de](mailto:info@baermann.de)

**Tel.:** +49 211 7489973 (Anrufbeantworter)

**Nur so kann der „vorletzte Bug“ gefunden werden.**

## Anhang 1: Versuchsplanung

Das Modul **Versuchsplanung** (Hauptmenu Extras) ermöglicht die Prognose einer großen Zahl von automatisch berechneten Datensätzen.

NN-Tool 2024 - C:\Users\info\Documents\NN-Tool\CLASS6 - [Versuchsplanung]

Übersicht | Scanbereich | Zielwerte | Ergebnisse

Lfd. Nr.	Eingangsparameter	Min.	Max.	Schritte	Rezepturkomp.
1	e1	2,0000	96,0000	2	nein
2	e2	1,0000	99,0000	2	nein
3	e3	0,0000	99,0000	2	nein
4	e4	1,0000	99,0000	2	nein
5	e5	1,0000	97,0000	2	nein
6	e6	0,0000	99,0000	2	nein

Versuchsanzahl: 1024

Zielwerte

Lfd. Nr.	Parameter	Wert	Gewichtung

Kosten mit optimieren

☐ ja ☒ nein

Kostendatei:  ...

< 0 = > Gewichtung: 1

Speichern | Laden | Versuchsplan starten

Parameter: 17 Datensätze: 100 Eingänge/-Knoten: 9/12 Ausgänge/-Knoten: 5/7

Nach der Durchrechnung von u.U. Tausenden von Sätzen findet eine Auswahl der Sätze nach den gewählten Zielwerten statt. Eine vom Anwender vorgegebene Anzahl der bezüglich der Zielwerte am besten geeigneten Datensätze wird angezeigt. Zunächst ist der sogenannte **Scanbereich**, auf dem die Untersuchung stattfinden soll, festzulegen. Wechseln Sie auf die Registerkarte **Scanbereich**:

NN-Tool 2024 - C:\Users\info\Documents\NN-Tool\CLASS6 - [Versuchsplanung]

Datei Datenanalyse Netztraining Ansicht Extras Online-Anwendungen Fenster Hilfe

Übersicht **Scanbereich** Zielwerte Ergebnisse

Lfd. Nr.	Eingangsparameter	Min.	Max.	Schritte	Rezepturkomponente
1	e1	2.0000	96.0000	2	nein
2	e2	1.0000	99.0000	2	nein
3	e3	0.0000	99.0000	2	nein
4	e4	1.0000	99.0000	2	nein
5	e5	1.0000	97.0000	2	nein
6	e6	0.0000	99.0000	2	nein
7	e7	0.0000	98.0000	2	nein
8	Class e8	Variabel		4	nein
9	e9	1.0000	99.0000	2	nein

Scanbereich  
Versuchsanzahl  Rezepturkomponenten  Vorschläge

Werte und Schritte setzen

Rezepturkomponenten setzen  
   Gesamtkonzentration

Klassifikator setzen  
 =

Parameter: 17 Datensätze: 100 Eingänge/-Knoten: 9/12 Ausgänge/-Knoten: 5/7

Hier sind für alle Eingangsparameter ein minimaler und maximaler Wert des Scanbereichs festzulegen. Außerdem muß die Zahl der Schritte (Niveaus) dazwischen festgelegt werden. Bei einer Festlegung von beispielsweise 30, 60 und 3 wird der entsprechende Parameter bei 30, 45 und 60 getestet. Letztendlich werden alle Kombinationen prognostiziert und die bezüglich der Zielwerte besten vorgeschlagen. Die Anzahl der Vorschläge kann im entsprechenden Feld eingestellt werden. Klassifikatoren können auf allen Klassen getestet werden ("Variabel") oder es kann eine Klasse vorgegeben werden. Darüberhinaus können Parameter als Rezepturkomponenten definiert werden. In diesem Fall müssen sie sich zur Gesamtkonzentration (i.d.R. 100) aufsummieren. Dies wird über eine Kompensationskomponente realisiert (diese sollte möglichst großen Wertebereich haben). Eine entsprechende Konfiguration könnte z.B. wie folgt aussehen:

NN-Tool 2024 - C:\Users\info\Documents\NN-Tool\CLASS6 - [Versuchsplanung]

Datei Datenanalyse Netztraining Ansicht Extras Online-Anwendungen Fenster Hilfe

Übersicht Scanbereich Zielwerte Ergebnisse

Lfd. Nr.	Eingangsparameter	Min.	Max.	Schritte	Rezepturkomponente
1	e1	2,0000	96,0000	4	ja
2	e2	1,0000	99,0000	4	ja
3	e3	0,0000	99,0000	4	ja
4	e4	1,0000	99,0000	1	Kompensationsk.
5	e5	1,0000	97,0000	4	ja
6	e6	0,0000	99,0000	4	nein
7	e7	0,0000	98,0000	4	nein
8	Class e8	Variabel		4	nein
9	e9	1,0000	99,0000	4	nein

Scanbereich  
Versuchsanzahl: 65536 Rezepturkomponenten: 5 Vorschläge: 100

Werte und Schritte setzen  
4

Rezepturkomponenten setzen  



Gesamtkonzentration: 100

Klassifikator setzen  
 =

Parameter: 17 Datensätze: 100 Eingänge/-Knoten: 9/12 Ausgänge/-Knoten: 5/7

Hier werden alle Eingänge auf 4 Niveaus getestet mit Ausnahme der Kompensationskomponente, die natürlich immer so nachgestellt werden muss, dass die Rezepturbedingung „Summe gleich 100%“ erfüllt ist. Insgesamt werden 65536 Varianten durchgerechnet und davon die bzgl. Zielwerten und Kosten besten 100 Datensätze angezeigt.

Die Vorgabe von Zielwerten und Kosten entspricht der Vorgehensweise bei der komplexen Optimierung. Eine ggf. benötigte Kostendatei ist mittels des Optimierungsmoduls zu erstellen.

**Hinweis:** Beachten Sie, dass bei der Definition von Rezepturkomponenten die Möglichkeit besteht, inkompatible Randbedingungen zu spezifizieren. In diesem Fall kann natürlich kein passender Datensatz gefunden werden.

## Anhang 2: Automatische Dokumentation

Dieses Feature im Hauptmenu „**Netztraining**“ ermöglicht es, vollautomatisch sämtliche für ein Netz relevanten Informationen zu erzeugen und in einer Excel-Mappe abzuspeichern. Dieses schließt Grafiken wie Scatter- und Verlaufplot mit ein.



## Anhang 3: Batchdatei

Neben der üblichen grafisch interaktiven Bedienung, läßt sich NN-Tool auch über eine sogenannte Batchdatei gewissermaßen fernsteuern (entspricht Macrorecorder). Dies ist besonders dann von Interesse, wenn die immer gleiche Netzstruktur mit immer neuen Datensätzen gelernt werden soll. Eine andere Möglichkeit ist die Integration von NN-Tool in userspezifische Anwendungen. Eine entsprechende Datei hat die Endung .cob und läßt sich automatisch nach dem Lernvorgang erstellen. Durch Doppelclick auf eine .cob-Datei führt NN-Tool dann die gesamte Modellierung vollautomatisch durch. Im Hauptmenu „**Extras**“, Menüpunkt „**Batchdatei erzeugen**“

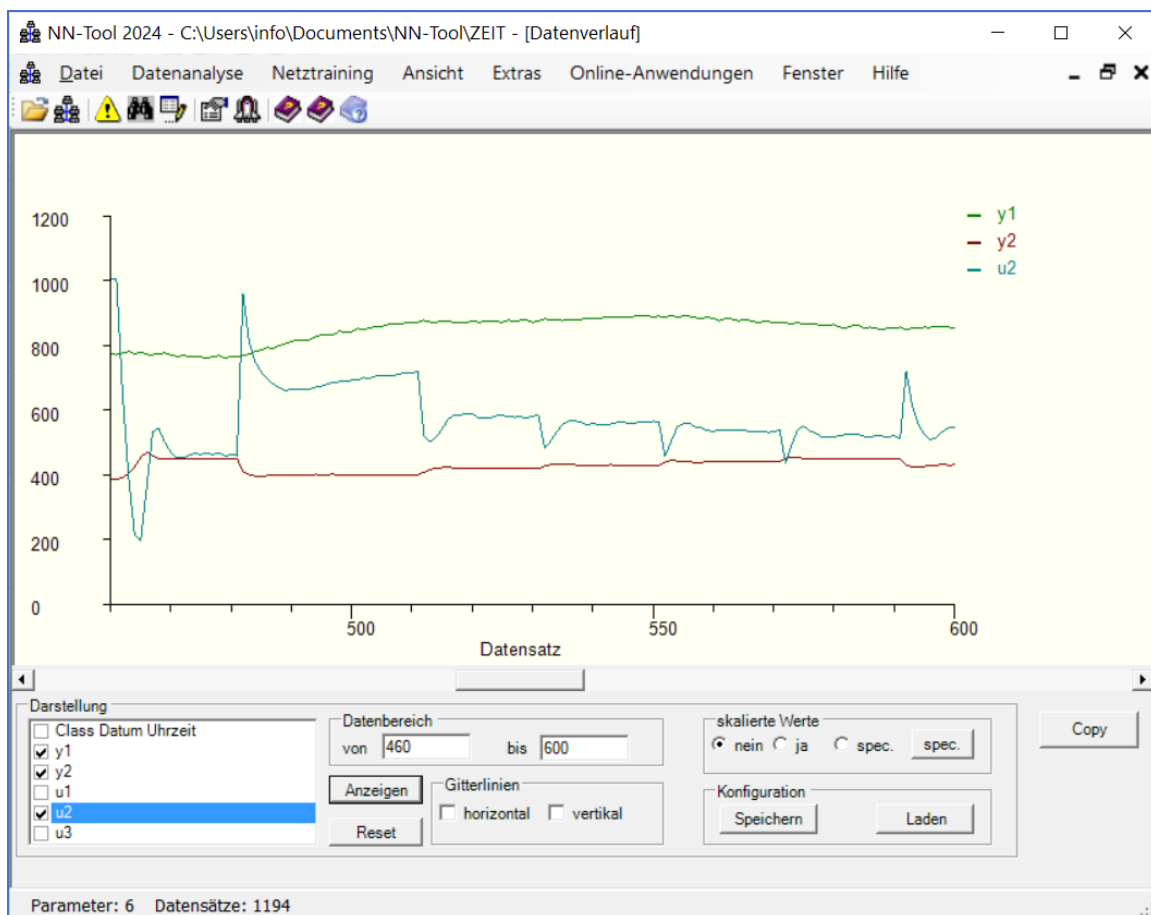
**Hinweis:** Dieses Feature unterstützt z.Z. nur statische Netze (d.h. keine Zeitreihen). Diese können jedoch extern geshiftet werden.

## Anhang 4: Spezielle Funktionen zur Datenanalyse und Vorverarbeitung

In diesem Kapitel werden zunächst die **speziellen Funktionen im Menu Datenanalyse** und die Funktion Histogramm (Start aus Fenster „Datenanalyse“) beschrieben. Die Funktionen „Datenverlauf“, „Histogramm“, „Cross-Correlations“ und „Optimale Eingänge“ dienen der genaueren Untersuchung einzelner Parameter, der automatischen Totzeitbestimmung sowie der Bestimmung optimaler Eingangsparameter bei Netzen mit sehr vielen Parametern. Anschließend werden einige Funktionen zur Datenvorverarbeitung vorgestellt, die vor allem (aber nicht ausschließlich) im Zusammenhang mit Zeitreihen von Interesse sind.

### A. Datenverlauf

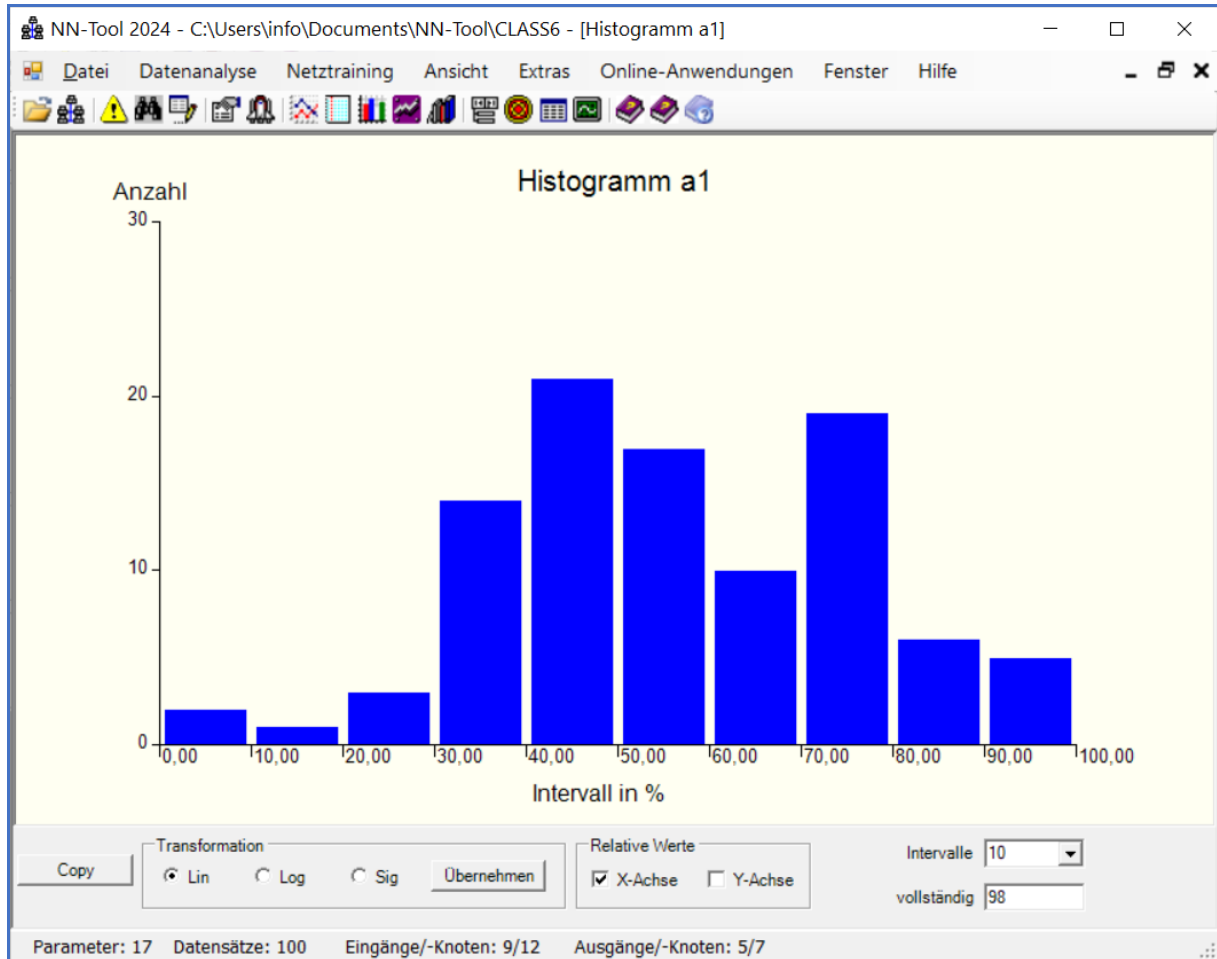
Mit der Form Datenverlauf können mehrere ausgewählte Parameter gleichzeitig über der Datensatznummer dargestellt werden.



Falls verschiedene Parameter unterschiedliche Größenordnung haben, empfiehlt sich die Option „skalierte Werte“ oder eine spezifische Skalierung (Option „spec“). Mittels der linken Maustaste können einzelne Grafikpunkte für Detailinformationen angeklickt werden. Durch Drücken und Ziehen mit der rechten Maustaste kann in die Grafik hineingezoomt werden. Mit der mittleren Maustaste wird der Zoom rückgängig gemacht und alle vorhandenen Datensätze werden angezeigt.

## B. Histogramm

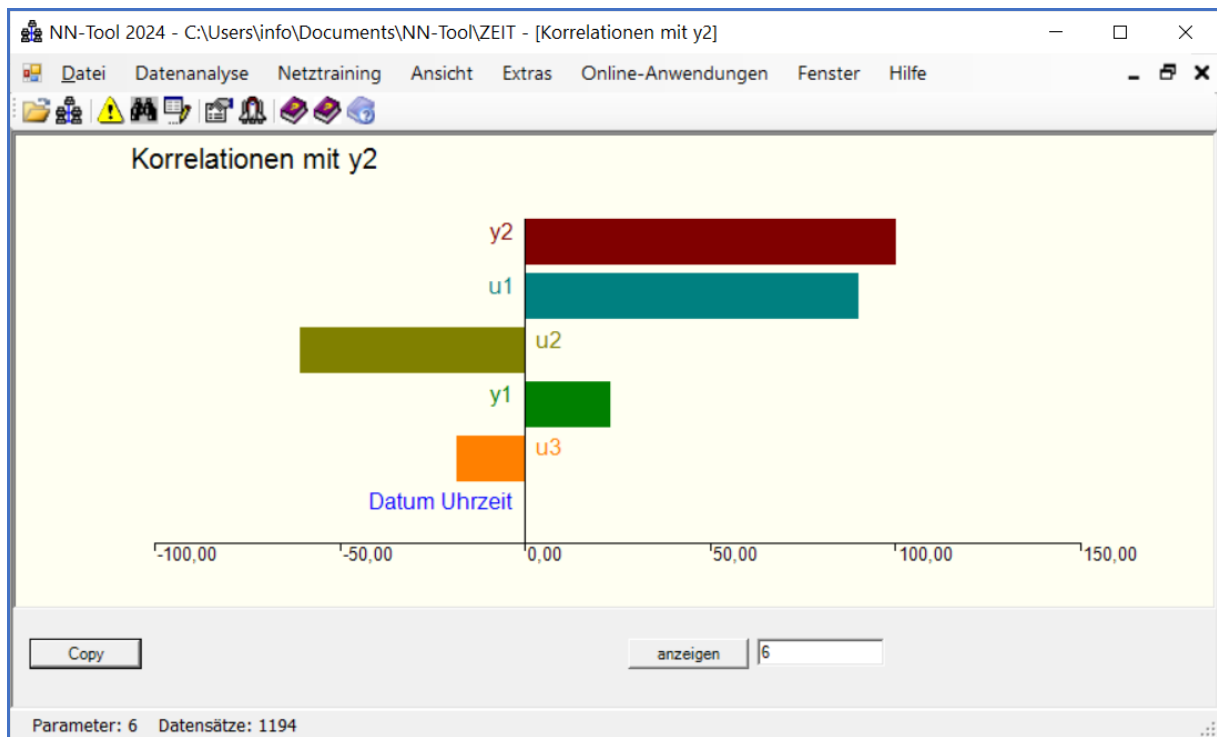
Diese Form stellt die Häufigkeitsverteilung (Histogramm) ausgewählter Parameter dar (Start aus **Fenster Datenanalyse** durch Doppelclick auf einen bestimmten Parameter). Es wird der prozentuale Anteil der Datensätze, für den ein bestimmter Parameter in ein bestimmtes Teilintervall fällt, über der Position des Intervalls dargestellt. Die Anzahl der Teilintervalle kann dabei vorgegeben werden.



**Hauptanwendungsbereich:** Idealerweise sollten Parameter möglichst gleichmäßig über ihren Wertebereich („Range“), d.h. das Intervall vom minimalen zum maximalen Wert, verteilt sein. D.h., das Histogramm sollte im Wesentlichen konstant auf dem optimalen Wert (100% geteilt durch die Anzahl Intervalle) liegen (sogenannte Gleichverteilung, ist im oberen Bild offenbar nicht gegeben). Dies ist insbesondere für eine gute Lernbarkeit von Ausgangsparametern wichtig. Falls einzelne Parameter (insbesondere Ausgangsparameter) stark von der Gleichverteilung abweichen, ist eine geeignete Transformation (z.B. Übergang zum Logarithmus dieser Größe) in Erwägung zu ziehen.

## C. Cross-Correlations

Diese Form zeigt zu einem im unteren Bereich ausgewählten Parameter die Korrelationen aller übrigen Parameter an. Das Ergebnis ist auch grafisch verfügbar:



Dabei wird zunächst der Korrelationskoeffizient jedes Parameters mit jedem anderen berechnet („**Korrelationsmatrix**“). Bei  $m$  Parametern und  $n$  Datensätzen ist der Rechenaufwand proportional zu  $n * m^2$ , d.h. im Allgemeinen hoch.

Nach der Berechnung lassen sich dann zu einem gewählten Parameter die Korrelationen sämtlicher anderen Parameter anzeigen. Über die Registerkarte „Korrelationsmatrix“ kann die gesamte Korrelationsmatrix dargestellt und ggf. nach Excel übertragen werden.

**Hauptanwendungsbereich:** Der Korrelationskoeffizient ist ein statistisches Maß dafür, ob zwei Parameter viel oder wenig miteinander zu tun haben. Ein Korrelationskoeffizient von +100% bedeutet, daß die beiden Parameter vollständig linear abhängig sind und daß der Anstieg des einen immer auch einen Anstieg des anderen zur Folge hat. Bei -100% sind die Parameter ebenfalls vollständig linear abhängig, jedoch sinkt der eine, wenn der andere steigt und umgekehrt. Ein Korrelationskoeffizient nahe 0 weist daraufhin, daß zumindest keine lineare Abhängigkeit zwischen den Parametern existiert. Diese Informationen können zur Auswahl von Eingangsparametern verwendet werden. Bei zwei hochkorrelierten Eingangsparametern (d.h. Betrag des Korrelationskoeffizienten nahe 100%) kann z.B. auf einen der beiden verzichtet werden, da der andere bereits die wesentliche Information bereitstellt. Genauer kann dies mit der Funktionalität „Optimale Eingänge“ analysiert werden.

## D. Optimale Eingänge

Die Funktionalität „Optimale Eingänge“ setzt voraus, dass die Eingangs- und Ausgangsparameter bereits festgelegt wurden. Dann liefert die folgende Form

Eingangsparameter	AusCorr.	EinCorr.	Anzahl	Bewertung
Komp6	42,33	0,00	500	142,33
Komp4	39,95	0,12	500	139,83
Komp3	38,48	0,14	500	138,34
Komp2	27,74	0,24	500	127,50
Komp5	24,00	41,67	500	82,33
Komp1	15,80	61,60	498	53,80

☒ Eig1  
☒ Eig2

Gewichtung  
 Ausgangskorrelation:   
 Eingangskorrelation:   
 Anzahl:   
 Bewertung Eingangskorrelation: ☒ max ☐ mittel

anzeigen  
 Grafik  
 Passiv setzen

Parameter: 9   Datensätze: 500   Eingänge/-Knoten: 6/6   Ausgänge/-Knoten: 2/2

für jeden Eingangsparameter eine Bewertung, wie wichtig dieser Parameter zur Prognose der im unteren Bereich ausgewählten Ausgangsparameter ist. Dazu wird zunächst die Korrelationsmatrix berechnet (s.o.). Der Eingangsparameter mit der höchsten Bewertung ist für die Prognose am wichtigsten. In die Bewertung gehen drei Kriterien ein:

1. **Ausgangskorrelation.**: mittlere absolute Korrelation des betreffenden Parameters zu den Ausgängen. Sollte möglichst hoch sein.
2. **Eingangskorrelation**: maximale oder mittlere absolute Korrelation eines Eingangs zu denjenigen anderen Eingängen, die in der Bewertung bzgl. der anderen beiden Kriterien weiter oben stehen. Hintergrund: Eingangsparameter werden nacheinander nach Prognoseeignung ausgewählt. Ein weiterer hinzukommender Eingangsparameter sollte möglichst wenig mit den bereits ausgewählten zu tun haben, sondern möglichst unabhängige weitere Informationen bereitstellen. Aus diesem Grund sollte die EinCorr. eines Eingangsparameters möglichst niedrig sein. Sie geht deshalb negativ in die Gesamtbewertung ein.
3. **Anzahl**: Anzahl des Auftretens des Parameters in den Datensätzen. Ein geeigneter Eingangsparameter sollte in möglichst vielen Datensätzen vorkommen. Ein Vorkommen in allen Datensätzen ergibt eine Anzahlbewertung von 100, ein Vorkommen in 20% der Datensätze eine Anzahlbewertung von 20.

Die drei Kriterien werden mittels der angegebenen Gewichtungsfaktoren gewichtet und ergeben dann die Gesamtbewertung. Die Gesamtbewertung kann auch negativ sein. Ein Eingang mit

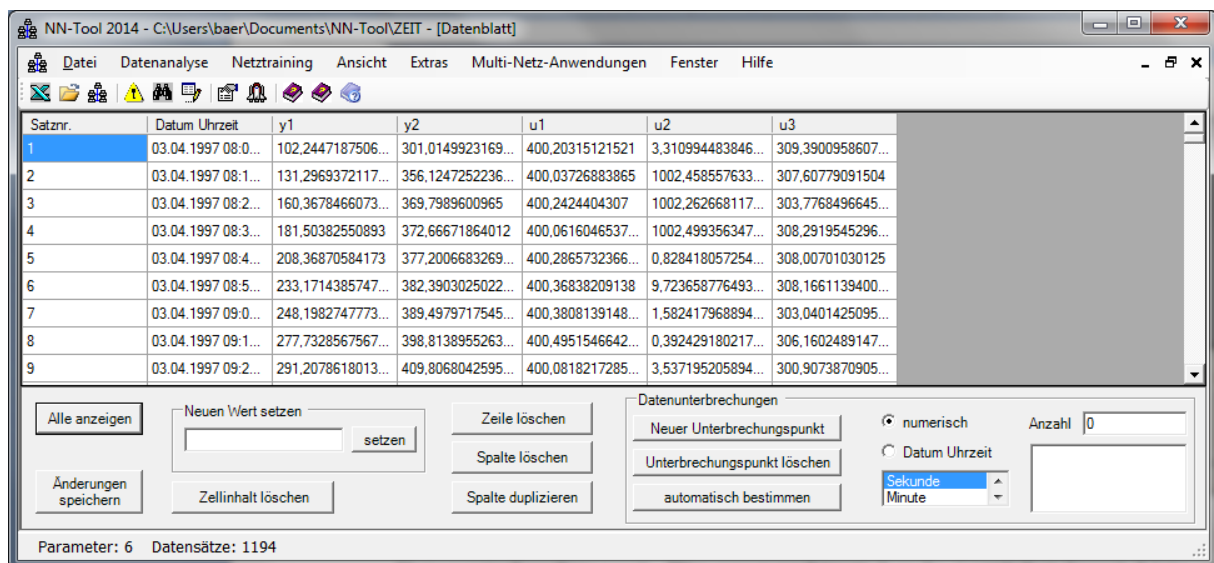
einer Gesamtbewertung von -30 ist weniger geeignet als einer mit -10 oder +5. Die Taste „**Anzeigen**“ ermöglicht eine Neuberechnung nach Änderungen der gewählten Ausgangsparameter oder Änderungen der Gewichtungsfaktoren. „**Grafik**“ liefert ein entsprechendes Diagramm. Mit der Taste „**Passiv setzen**“ besteht die Möglichkeit jeden Eingangsparameter passiv zu setzen, der nicht eine gewisse Mindestbewertung schafft.

**Hauptanwendungsbereich:** Bei vielen Anwendungen im Bereich der Prozeßmodellierung hat man es mit einer großen Zahl von Eingangsparametern zu tun, die untereinander stark korreliert sind (Bei Rezepturprognosen ist dies meist nicht so relevant). Eine Anlage ist beispielsweise mit einer Vielzahl von Druck- oder Temperatursensoren ausgestattet, die alle Ähnliches messen. Dies trifft insbesondere auch auf Zeitreihen zu, bei denen ein und derselbe Parameter zu unterschiedlichen Totzeiten auftritt. Die Funktionalität „Optimale Eingänge“ erlaubt es, aus der großen Zahl der Eingangsparameter systematisch diejenigen auszuwählen, die zur Prognose der Ausgänge besonders geeignet und dabei untereinander möglichst unkorreliert sind. Dies kann auch zur automatischen Totzeitbestimmung genutzt werden. Dazu wähle man im Totzeiteneditor alle in Frage kommenden Zeitverschiebungen und bestimme schließlich die geeignetsten. Man beachte jedoch den Rechenzeitanstieg zur Berechnung der Korrelationsmatrix.

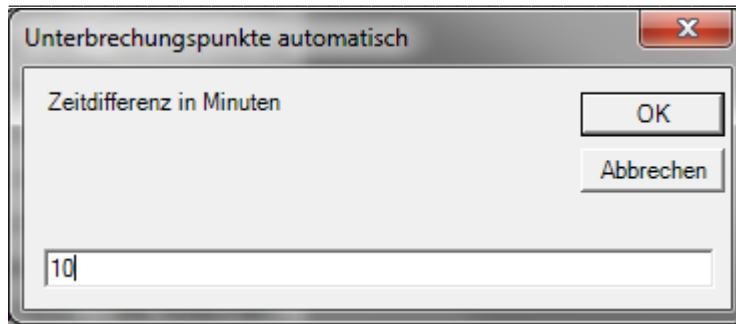
**Wichtiger Hinweis:** Im Allgemeinen wird anstelle dieser Funktionalität die Verwendung des deutlich leistungstärkeren Verfahrens „**Input-Optimierungslauf**“ (siehe Kapitel 6) empfohlen. Der Input-Optimierungslauf berücksichtigt nämlich auch nichtlineare Abhängigkeiten zwischen Ein- und Ausgangsparametern.

## E. Datenunterbrechungen

In Kapitel 6 „Zeitreihen“ hatten wir die Modellierung von Systemen diskutiert, bei denen zwischen den Ein- und Ausgangsgrößen zeitliche Verschiebungen auftreten. Mittels des Menüpunktes Totzeiten im Menu Datenanalyse hatten wir zeitlich verschobene Parameterspalten bereitgestellt. Hierbei haben wir allerdings eine eigentlich unzulässige Vereinfachung vorgenommen. Zu beachten ist nämlich, dass die Datensätze in diesem Beispiel keineswegs kontinuierlich erfasst worden sind, sondern dass zwischenzeitlich Unterbrechungen der Datenreihen durch Anlagenstillstände auftraten. **Nach der Datenanalyse und vor der Zuordnung der Totzeiten** sind für eine genauere Modellierung zunächst diese sogenannten Datenunterbrechungen festzulegen. Wählen Sie im Menu **Datenanalyse** den Menüpunkt **Datenunterbrechungen**:



Es erscheint das bereits bekannte Datenblatt mit der zusätzlichen Funktionalität „Datenunterbrechungen“. Die ganz rechts stehende (zunächst leere) Liste zeigt die bisher bestimmten Datenunterbrechungen anhand der Datensatznummer an. Datenunterbrechungen können manuell über die Tasten **Neuer Unterbrechungspunkt** bzw. **Unterbrechungspunkt löschen** eingegeben werden. Wenn beispielsweise zwischen Datensatz 5 und Satz 6 eine Unterbrechung auftritt, markieren Sie bitte mit der Maus in der Spalte Satznr. die Werte 5 und 6, d.h. den der Unterbrechung vorausgehenden und den nachfolgenden Satz und drücken Sie die Taste **Neuer Unterbrechungspunkt**. Geben Sie verschiedene Unterbrechungspunkte ein und löschen Sie sie anschließend wieder. Effizienter ist es i.a. die Unterbrechungspunkte automatisch anhand von Unregelmäßigkeiten in einer ausgewählten Spalte zu bestimmen. Wählen Sie die Spalte **Datum Uhrzeit** indem Sie eine beliebige Zelle anklicken (dass diese eine Kennung ist, spielt hier keine Rolle) und setzen Sie die Option **numerisch / Datum Uhrzeit** auf Datum Uhrzeit. Da der reguläre Zeittakt bei diesen Daten 10 Minuten beträgt, ist es sinnvoll die Zeiteinheit auf Minuten zu setzen. **Wichtig:** „Minuten“ muss nicht nur sichtbar sondern auch blau unterlegt sein. Drücken Sie dann die Taste **automatisch bestimmen**. Sie werden dann aufgefordert den regulären Zeittakt in der gewählten Zeiteinheit anzugeben. In diesem Beispiel beträgt der Zeittakt 10 Minuten:



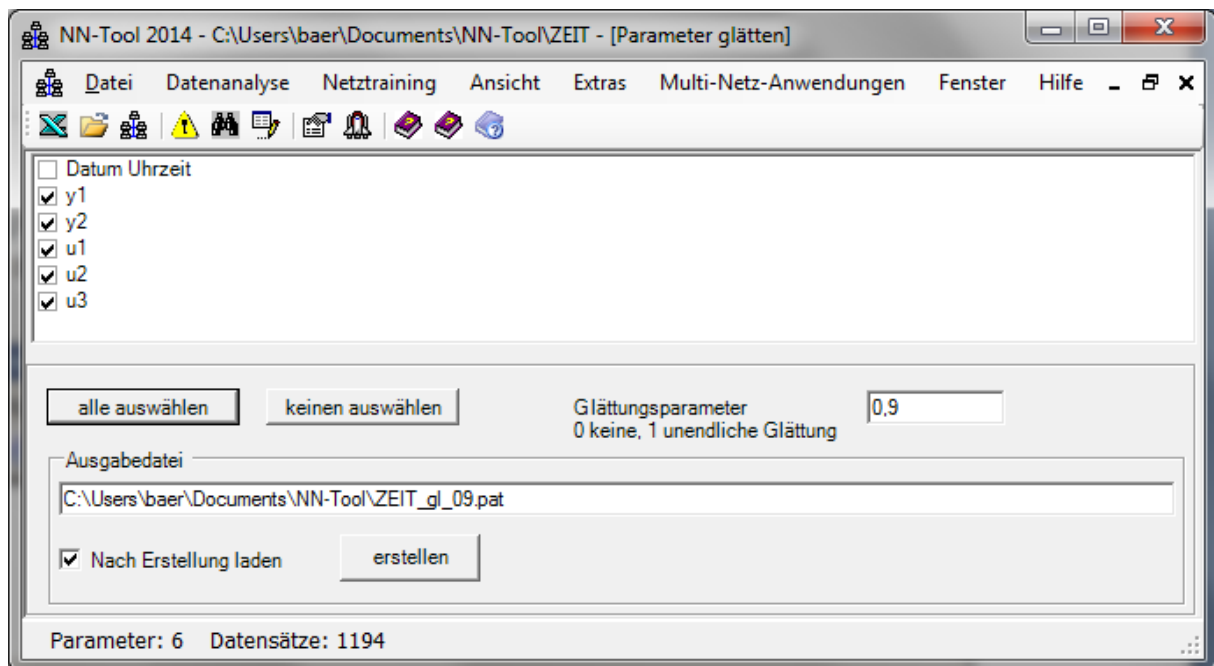
Anschließend werden Sie nach der maximal zulässigen Schwankung des Zeittaktes in % (z.B. bedingt durch manuelle Zeitnahme) gefragt. Geben Sie hier den Wert 30 ein. Dies bedeutet: Wenn zwei aufeinanderfolgende Datensätze nicht zwischen 7 Minuten (=10Minuten - 30%) und 13 Minuten auseinander liegen, wird eine Datenunterbrechung angenommen. Es werden in diesem Beispiel 4 Unterbrechungspunkte gefunden, aufgelistet und farblich markiert.

Setzen Sie nun die Modellierung wie im Kapitel Zeitreihen beschrieben durch Definition der Totzeiten fort. Der Unterschied besteht darin, dass an den Unterbrechungspunkten die Werte der zeitlich verschobenen Größen nicht ermittelt werden. Damit sind die entsprechenden Datensätze unvollständig und werden (korrekterweise) bei der Modellierung nicht berücksichtigt.



## F. Parameter glätten

Diese Funktionalität im Hauptmenu Datenanalyse ermöglicht es, die .pat-Datei um zusätzliche Spalten zu ergänzen und als neue erweiterte Datendatei zu erzeugen.



Diese zusätzlichen Spalten enthalten dann die (zeitlich) geglätteten Originalwerte. Dies macht natürlich in der Regel nur bei Zeitreihen überhaupt einen Sinn. Das Maß der Glättung wird durch den Glättungsparameter, einen Wert zwischen 0 und 1 bestimmt. Die Ausgabedatei enthält sämtliche Originalspalten sowie die geglätteten Versionen der ausgewählten Parameter (Klassifikatoren und Kennungen können nicht geglättet werden).

## **G. Datenfiles verketten**

Mit diesem Menüpunkt im Hauptmenu Datei ist es möglich, mehrere .pat-Files, die dann natürlich exakt die gleichen Parameter enthalten müssen, aneinander zu hängen. Dies ist z.B. bei Anlagenmodellierungen von Interesse, wenn pro Tag eine Datendatei erzeugt wird und man die Daten von mehreren Tagen zu einer Modellierung verwenden möchte.

## **H. Datenfile reduzieren**

Diese Funktionalität im Hauptmenu Datei ermöglicht es, aus einer .pat-Datei nur jeden n-ten Satz auszuwählen und als reduzierte .pat-Datei zu erzeugen. Dies kann bei sehr großen Datenumfängen (viele Tausend Datensätze) mit geringer Variation der aufeinanderfolgenden Datensätze von besonderem Interesse sein. Diese Verdünnung des Datenbestandes ermöglicht es dann, sich zunächst einen schnelleren Überblick über die Vorgehensweise bei der Modellierung zu verschaffen.

## Anhang 5: Ausreisserliste

Diese Komponente ermöglicht die automatische Bestimmung von zweifelhaften Datensätzen. Es lassen sich alle Datensätze auflisten, bei denen der Fehler zwischen prognostiziertem und gemessenem Wert besonders groß ist (Ausreisser). Als Maß für die Größe des Fehlers wird dabei der mittlere Fehler auf dem Testset mit einem Fehlerfaktor (z.B. 2 oder 3) multipliziert. Jeder Datensatz, bei dem der Fehler dann größer als der doppelte oder dreifache mittlere Fehler ist, wird durch Drücken der Taste "Berechnen" bestimmt. Anschließend können die Sätze mit der entsprechenden Taste markiert werden. Zuletzt kann ein neues reduziertes .pat-File erzeugt werden, in welchem die markierten Sätze fehlen. Mit dieser neuen Datendatei läßt sich dann das Netz erneut trainieren.

### **Hinweis:**

Dynamische Daten werden beim Abspeichern in statische Daten transformiert (ist aus Konsistenzgründen notwendig).

## Anhang 6: Validationsets

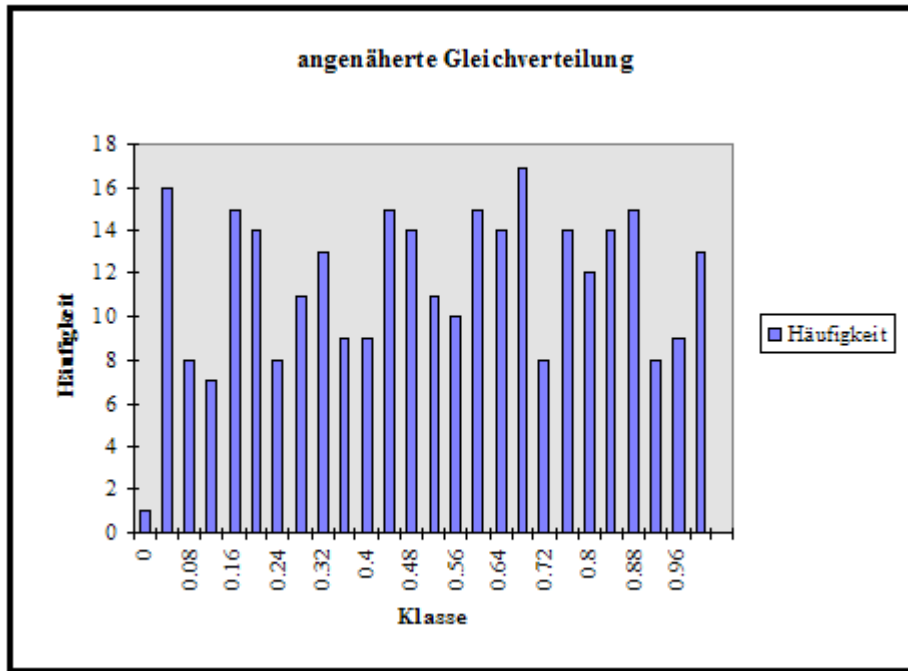
Ein Validationset ist ein unabhängiger Datensatz, der in einer eigenen Datei (.vali-Datei) bereitgestellt wird und vom Netz prognostiziert werden soll. Der Aufbau einer Validationsetdatei muß exakt dem Aufbau einer .pat-datei entsprechen. Zweckmäßigerweise wird man zur Erzeugung einer .vali-Datei auf die von NN-Tool unter Excel bereitgestellte Funktionalität zur Erzeugung von .pat-Dateien zurückgreifen. Ausgeführt wird die Prognose durch Anwählen des Menüpunkts **„Prognose Validationset“** im Menu „Netztraining“.

Der Mechanismus kann auch zur Prognose von neuen Datensätzen herangezogen werden. Die Ausgänge werden dann mit irgendwelchen Werten (z.B. 0 ) vorbelegt (leere Werte sind nicht zulässig). Nach der Prognose können die entsprechenden Einzelwerte über die Taste Excel in eine Excelmappe geschrieben und analysiert werden.

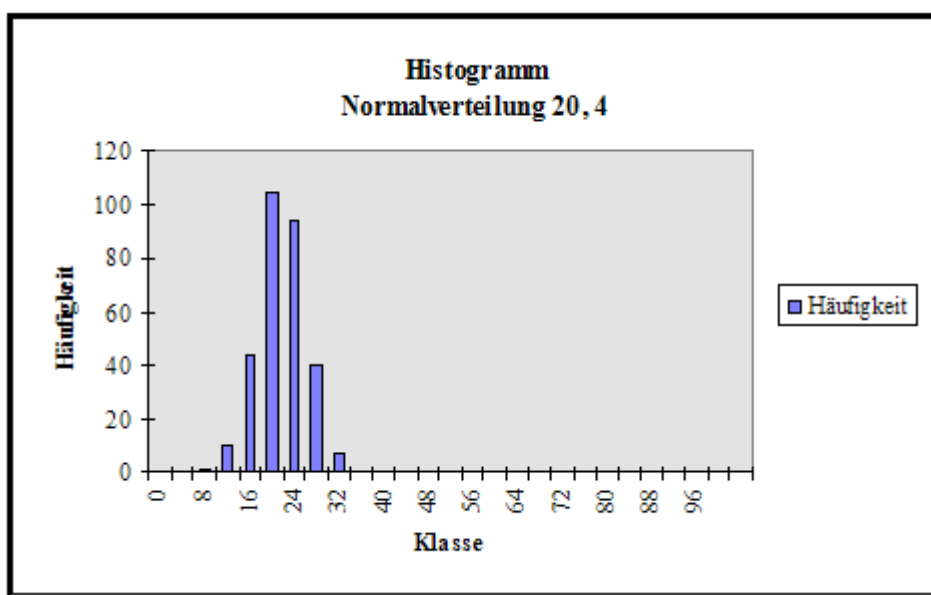
## Anhang 7: Nichtlineare Skalierungstransformationen

### A7a) Bedeutung der statistischen Verteilung von Ein- und Ausgangsgrößen

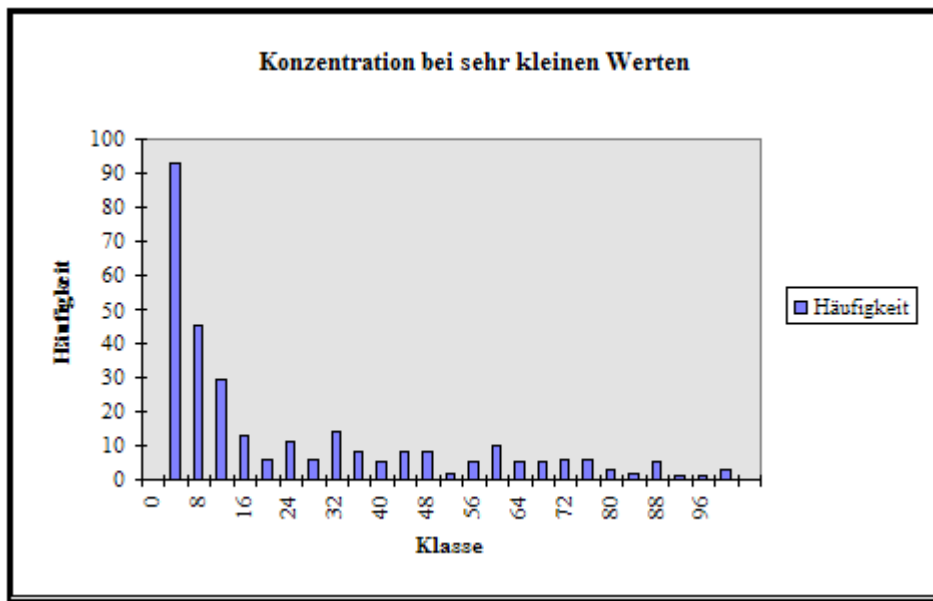
Damit ein neuronales Netz den mathematischen Zusammenhang zwischen verschiedenen Parametern optimal lernen kann, sollten diese Parameter, wenn eben möglich, gleichmäßig über ihren Wertebereich verteilt sein (Abb.1 angenäherte Gleichverteilung).



In vielen Anwendungsfällen sind einzelne Parameter (Meßgrößen) jedoch keineswegs gleichverteilt, sondern die Messungen konzentrieren sich an bestimmten Punkten. Das folgende Diagramm (Abb. 2) zeigt eine Normalverteilung mit Mittelwert 20 und Standardabweichung 4:



Neben der Normalverteilung tritt häufig auch eine Konzentration bei sehr kleinen Werten auf (Abb.3):



Während die Konzentration der Meßpunkte bei Eingangsparametern teilweise noch tolerabel ist (hier kann das NN selbst für eine Entzerrung sorgen), sollten derartige Ausgangsparameter zunächst in neue, gleichmäßiger verteilte Größen, umtransformiert werden. Zur Zeit stehen neben der linearen Transformation (Standardfall) die folgenden beiden Transformationen in NN-Tool zur Verfügung:

1. Sigmoid-Transformation:

$$T_S(x) = \frac{1}{1 + e^{-a(x-m_x)}}$$

2. Logarithmische Transformation:

$$T_L(x) = \frac{\log(x) - \log(\min)}{\log(\max) - \log(\min)}$$

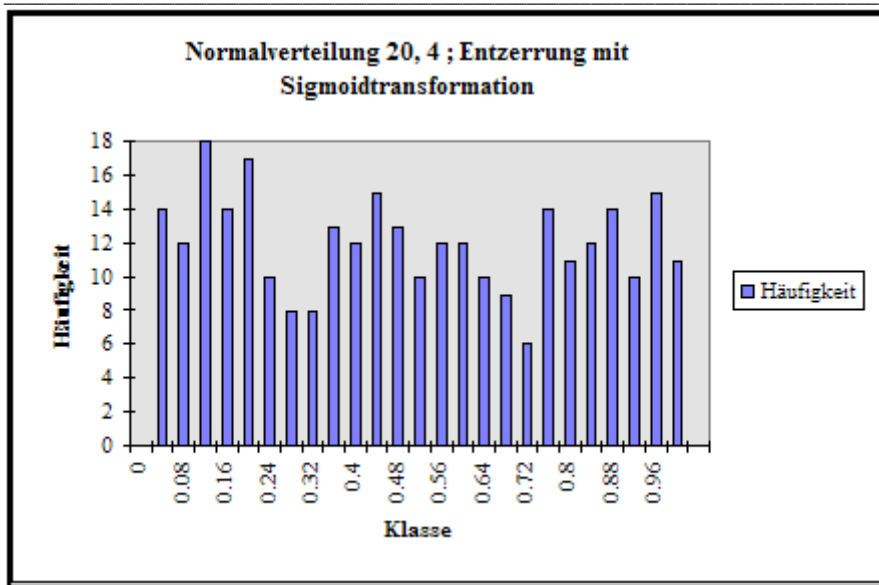
Dabei bedeuten min, max und  $m_x$  die minimalen und maximalen Werte der Größe  $x$  sowie ihr Mittelwert. Für  $a$  gilt:

$$a = \frac{1.6}{\text{Standardabweichung}(x)}$$

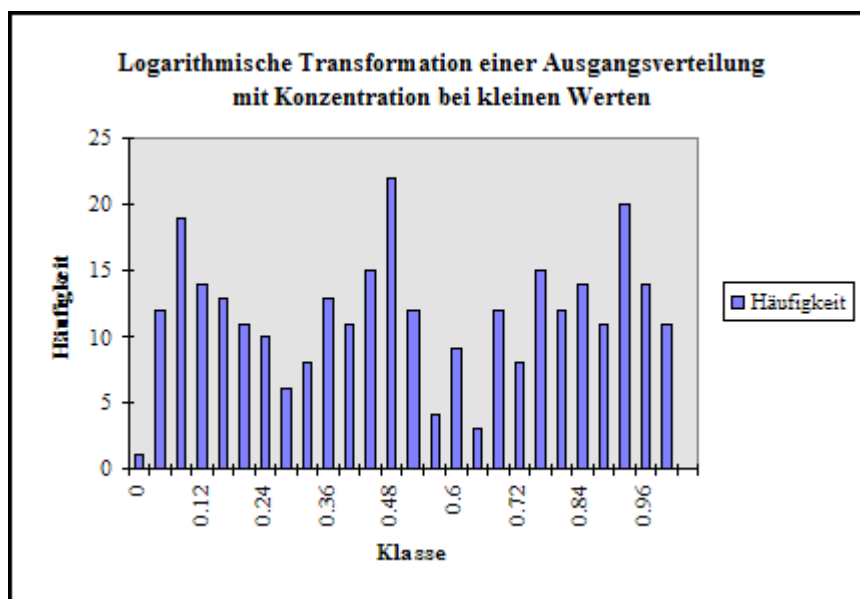
Der Wert von 1.6 läßt sich theoretisch begründen. Die Theorie liefert:

$$\frac{4}{\sqrt{2\pi}} \cong 1.5958$$

Nach Anwendung der Sigmoid-Transformation auf die Normalverteilung der Abb. 2 ergibt sich für die transformierte Größe die folgende Verteilung (Abb. 4):



d.h. die neue Größe ist wesentlich gleichmäßiger über ihren Wertebereich verteilt.  
Für die Verteilung der Abb. 3 ergibt sich nach Anwendung der logarithmischen Transformation (Abb. 5):



Auch diese Größe ist nun gleichmäßiger über ihren Bereich verteilt.

**A7b) Anwendung in NN-Tool**

Die Transformationen können im Datenanalysefenster oder im Fenster Histogramm manuell gesetzt werden. Dabei stellt sich jedoch die Frage, wann welche Transformation angewendet werden sollte.

**Automatische Anwendung:** Mittels der Tasten „Auto-Eingänge“ und „Auto-Ausgänge“ im Fenster Datenanalyse kann NN-Tool die Transformationen selbstständig setzen. Dazu berechnet NN-Tool standardmäßig zwei Kriterien:

$$K_1 = \frac{\text{Max} - \text{Min}}{\text{Stdabw}}$$

$$K_2 = \frac{\text{Mittel} - \text{Min}}{\text{Median} - \text{Min}}$$

Der Median einer Größe  $x$  ist ein Wert, für den 50% der Werte von  $x$  größer und 50% kleiner sind. Der Median ist i.a. vom Mittelwert verschieden (man mache sich das am Beispiel der Vermögensverhältnisse eines Multimillionärs und seiner drei Putzfrauen klar).

NN-Tool setzt die Transformationen, falls eine der folgenden Bedingungen erfüllt ist:

**$K_1 > 5,7$  , dann Sigmoidtransformation anwenden.**

**$K_2 > 2,5$  , dann logarithmische Transformation anwenden**

**Falls für einen Ausgangsparameter eine nichtlineare Transformation gesetzt wurde (manuell oder automatisch), dann kann im Scatterplot – der ja normalerweise die rückskalierten (bzw. rücktransformierten Originalgrößen darstellt – auf die transformierten Werte umgeschaltet werden.**

**Hinweis des Verfassers:** Für diese Option liegen noch nicht viele Erfahrungen vor. Nach ersten Untersuchungen scheint dies bei den Ausgangsparametern eine sehr sinnvolle Option zu sein. Bei den Eingangsparametern ist die Verwendung nichtlinearer Transformationen eher fragwürdig, insbesondere bei Rezeptureigenschaftsbeziehungen.



## Anhang 8: Integration von Vorwissen / Input-Output-Zuordnung

Im Fenster Datenanalyse bieten sich die folgenden Möglichkeiten:

**Registerkarte Input-Output-Zuordnung:** Ermöglicht die Festlegung von Eingangs- zu Ausgangsparametern. Standardmäßig wird jeder aktive Eingangsparameter zur Modellierung jedes Ausgangsparameters herangezogen. Mittels dieser Funktion kann NN-Tool angewiesen werden, bestimmte Eingänge bei bestimmten Ausgängen unberücksichtigt zu lassen oder nur einen rein monotonen Zusammenhang zuzulassen. Falls ein Eingang unberücksichtigt wird, wirkt sich dies entsprechend auf die Anzahl der vollständigen Datensätze eines Ausgangsparameters aus. D.h. ein selten gemessener Eingangsparameter, der für einen bestimmten Ausgang deaktiviert wird, reduziert nicht die Zahl der für diesen Ausgang vollständigen Datensätze.

**Verfügbare Datensätze:** Mit dieser Information in der entsprechenden Spalte des Fenster Datenanalyse wird vorab angezeigt, wieviele vollständige Datensätze beim Lernvorgang für jeden Ausgangsparameter vorhanden sind. Gegebenenfalls sollten einige selten gemessenen Eingangsgrößen passiv gesetzt werden, oder unter Verwendung der Input-Output-Zuordnung bei bestimmten Ausgängen abgeschaltet werden. Durch diese Massnahmen erhöht sich die Zahl vollständiger Datensätze.

## Anhang 9: Registerkarte Lernparameter / Crossvalidation

Mit der Registerkarte **Lernparameter** im Fenster **Datenanalyse** können zusätzliche Einstellungen für den Lernvorgang vorgenommen werden. Diese Einstellungen sind besonders bei kritischen Anwendungen, bei denen das Verhältnis der Zahl der vollständigen Datensätze zur Zahl der Eingangsparameter klein ist (z.B. kleiner 3), von Vorteil. Die wichtigste Massnahme ist sicherlich die Aufteilung der Datensätze. Hier sind besonders die Crossvalidationoptionen für kritische Anwendungen von Bedeutung:

- **Aufteilung der Datensätze:** Bei manchen Anwendungen hängt die Güte des resultierenden Netzmodells empfindlich davon ab, welche Datensätze zum Lernen (Lernset) und welche zum Testen verwendet werden. Dies ist insbesondere der Fall, wenn nur wenige Datensätze zur Verfügung stehen oder die Datensätze untereinander hochkorreliert sind (z.B. bei Zeitreihenanwendungen). Aus diesem Grund stehen die folgenden Optionen für die Aufteilung zur Verfügung:
  - **jeder n-te Satz im Testset / zusammenhängender Testset:** gibt an, wie die Datensätze des Testsets aus den für einen Ausgangsknoten zur Verfügung stehenden Sätzen auszuwählen sind. Für Versuchsreihen ist i.a. die Option: „jeder n-te Satz“ sinnvoller, für Zeitreihen ist wegen der typischerweise hohen Korrelation benachbarter Datensätze die Option „zusammenhängender Testset“ vorzuziehen. NN-Tool wählt die entsprechende Option selbsttätig, d.h. auch hier ist normalerweise keine Änderung notwendig.
  - **spezifische Einteilung:** Ermöglicht die detaillierte Zuordnung jedes Datensatzes zum Lern- oder Testset über die Registerkarte Lernset-/Testseteinteilung. Dabei kann den Datensätzen auch zusätzlich eine Gewichtung mitgegeben werden. Dabei gilt: Gewicht 1 entspricht Standard, 0 = Satz wird nicht berücksichtigt, 2,3,4,.. Satz geht entsprechend oft in die Lern- bzw. Testmenge ein.
  - **Crossvalidation:** Ermöglicht es mit mehrfachen, dynamisch erzeugten Testsets pro Ausgang zu arbeiten. Diese Option verringert die Abhängigkeit der ausgewählten Netzstruktur von einem speziellen Testset. Falls beispielsweise im Feld n-fach Crossvalidation der Wert 5 eingetragen ist, werden für jeden Ausgangsparameter 5 Aufteilungen von Lern- und Testset vorgenommen. Für jede dieser Aufteilungen wird das gesamte Testprogramm bzgl. der inneren Knoten und der Lernschritte automatisch durchgeprüft. Dann wird jeder Ausgangsknoten auf allen Datensätzen mit der optimalen Netzstruktur fertigtrainiert. Zuletzt werden die einzelnen Teilnetze wieder zum Gesamtnetz integriert. **Hinweis:** Eine unter dem Punkt spezifische Einteilung vorgenommene Datengewichtung wird auch bei Crossvalidation berücksichtigt. Für die dynamische Aufteilung der Sets stehen selbst wieder vier verschiedene Strategien zur Verfügung:
    - **Crossvalidation n-fach zyklisch:** Bei dieser Aufteilung und  $n=5$  besteht der erste dynamische Testset aus dem 1,6,11,.. vollständigen Datensatz, der zweite Testset aus dem 2,7,12,.. Datensatz etc. Diese Aufteilung ist i.a. **besonders geeignet für unkorrelierte Versuchsreihen.**
    - **Crossvalidation n-fach blockweise:** Bei dieser Aufteilung und  $n=5$  besteht der erste dynamische Testset aus dem ersten Fünftel der vollständigen Datensätze, der zweite Testset aus dem zweiten Fünftel etc. Diese Aufteilung ist i.a. besonders für **zeitabhängige Daten** („Zeitreihen“, siehe nächstes Beispiel) **optimal** geeignet.

- **Crossvalidation n-fach zufällig:** Bei dieser Aufteilung und  $n=5$  besteht jeder der fünf dynamischen Testsets aus Datensätzen, die mit der unter %-Anteil angegebenen Wahrscheinlichkeit dem Testset zugeordnet wurden. In diesem Modus kann keine endgültige Bewertung der Testsetgüte durchgeführt werden (nur Lernset).
- **Crossvalidation Leave-One-Out:** Im Extremfall "Leave-One-Out" werden genau so viele Testsets gebildet und zur Netzbestimmung herangezogen, wie es vollständige Datensätze zum betreffenden Parameter gibt. Jeder Testset besteht dann nur aus einem Datensatz (dabei ist während des Trainingslaufs keine Korrelationsangabe möglich!). Dieser Betriebsmodus ist der **bei Weitem rechenintensivste**, liefert aber – zumindest bei unkorrelierten Daten – die **höchste Modellgenauigkeit**. Besonders für kleine Datensätze von (hoffentlich) unkorrelierten Versuchsreihen geeignet. Bei Zeitreihen ist diese Option wegen der im Allgemeinen hohen Korrelation der Datensätze untereinander nicht empfehlenswert.

Bei korrelierten Daten ist oft auch die folgende Option von Bedeutung:

- **Fehlerbewertung bei Auswahl in %:** Wie bereits erwähnt, testet NN-Tool während des Lernvorgangs für jeden Ausgangsparameter einzeln die optimale Netzkonfiguration bei verschiedener Anzahl von Lernschritten und inneren Knoten. Zuletzt wird das Netz herangezogen, das die beste Vorhersagegenauigkeit auf dem Testset erzielt. Dazu werden normalerweise die mittleren relativen Fehler der einzelnen Netze verglichen. Mit dieser Option können nun zusätzliche Strafterme auf Netze mit vielen inneren Knoten gelegt werden. Ein Wert von  $x > 0$  bei dieser Option bedeutet, daß der mittlere rel. Fehler eines Teilnetzes mit  $n$  Knoten noch mit einem Faktor

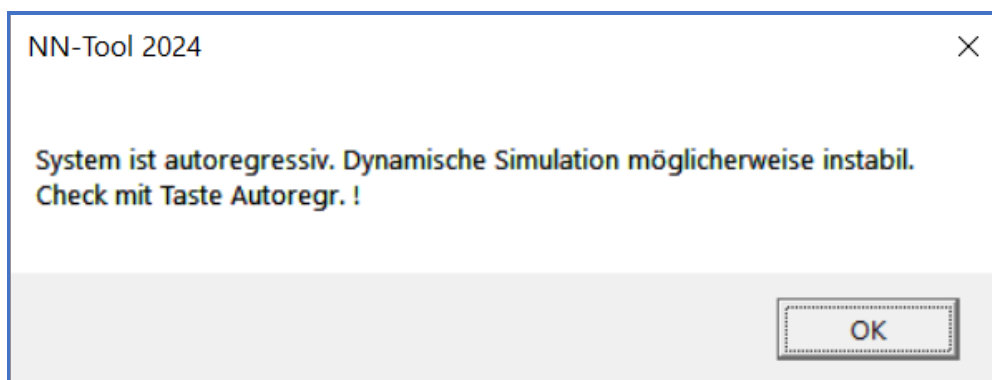
$$1 + (nx / 100)$$

multipliziert wird. Dies liefert dann den bewerteten Fehler. Dieser wird dann zur Netzauswahl herangezogen. Diese Option ist von Bedeutung, wenn von einer hohen Korrelation der Datensätze der Testmenge mit entsprechenden Datensätzen der Lernmenge ausgegangen werden muß. In diesem Fall stellt die Testmenge keine allein ausreichende Bewertungsgrundlage zur Verfügung.

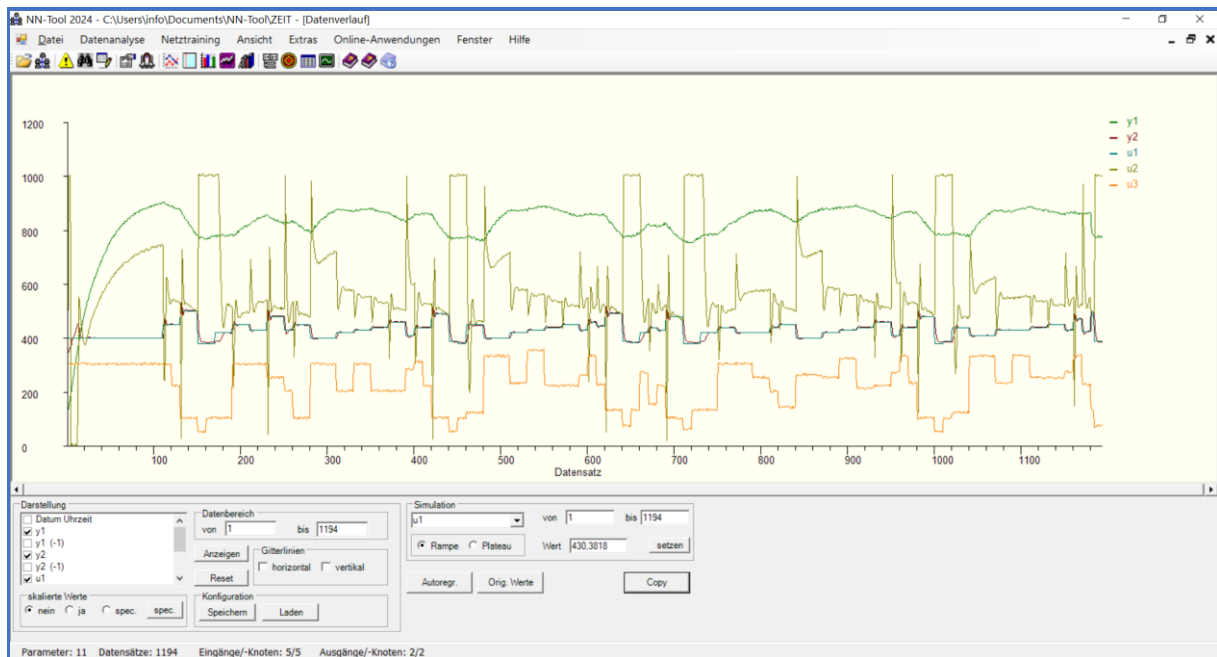
## Anhang 10: Dynamische Simulation

Diese Anwendung im Hauptmenu „Extras“ dient der Simulation von Zeitreihen. Sie ist ganz analog zur Form „Datenverlauf“ (Anhang 4) aufgebaut. Bitte machen Sie sich zunächst damit vertraut. Laden (bzw. erstellen) Sie dann das Netz **zeit.dsc** wie in Kapitel 8 beschrieben. Bei der Anwendung **zeit** handelt es sich um ein sogenanntes autoregressives Modell. **Autoregressiv** bedeutet, dass mindestens eine der Ausgangsgrößen selbst auch Eingangsgröße ist, dann natürlich zu einem früheren Zeitpunkt. D.h. die Modellierung einer Größe beruht (teilweise) auf derselben Größe (daher „Auto“). Das System **zeit** wäre z.B. nicht autoregressiv, wenn die Eingänge nur aus den u-Parametern und die Ausgänge nur aus den y-Parametern bestünden. Autoregressiv ist also nicht zwangsläufig identisch mit dem Auftreten von Totzeiten. Autoregressive Modelle verwenden immer Totzeiten, aber nicht jedes Totzeitenmodell ist autoregressiv.

Warum ist diese Charakterisierung wichtig? Wenn mit Hilfe eines autoregressiven Modells Simulationsrechnungen durchgeführt werden sollen, will man ja in der Regel Vorhersagen der Ausgangsgrößen für bestimmte vorgegebene Verläufe der Eingangsgrößen machen. Falls dabei mehr als nur ein Schritt in die Zukunft gerechnet werden soll, muss das Modell wiederholt angewandt werden. Dazu müssen die im vorhergehenden Prognoseschritt ermittelten Ausgangsgrößen (hier y) im nächsten Schritt als Eingangsgrößen verwendet werden. D.h. ab dem zweiten Schritt basiert die Simulation im Gegensatz zur ursprünglichen Modellbildung nicht mehr ausschließlich auf Meßdaten. Je weiter man in die Zukunft rechnet, desto häufiger werden die berechneten Werte, die ja möglicherweise Modellfehler beinhalten, eingesetzt. Dies bedeutet, dass autoregressive Modelle im mathematischen Sinne dynamische Systeme sind, während nichtautoregressive Modelle einfache Funktionen sind. Insbesondere besteht bei autoregressiven Modellen die Gefahr, dass sich die Modellfehler immer mehr aufschaukeln, je weiter man mit dem Modell in die Zukunft rechnet. In diesem Fall wird die Simulation als instabil bezeichnet. Dieser Effekt kann unterschiedlich stark ausgeprägt auftreten, man sollte sich aber zumindest dieser Möglichkeit bewußt sein. Öffnen Sie nun die Anwendung „Dynamische Simulation“. Sie werden zunächst nach einer Datei gefragt, da die Simulation prinzipiell mit jeder Datendatei durchgeführt werden kann, die dem Format der .pat-Datei entspricht (z.B. wenn neue Messdaten des Prozesses vorliegen). Wählen Sie **zeit.pat**. Falls ein autoregressives System vorliegt, erscheint die folgende Warnmeldung um auf die Möglichkeit einer instabilen Simulation hinzuweisen.

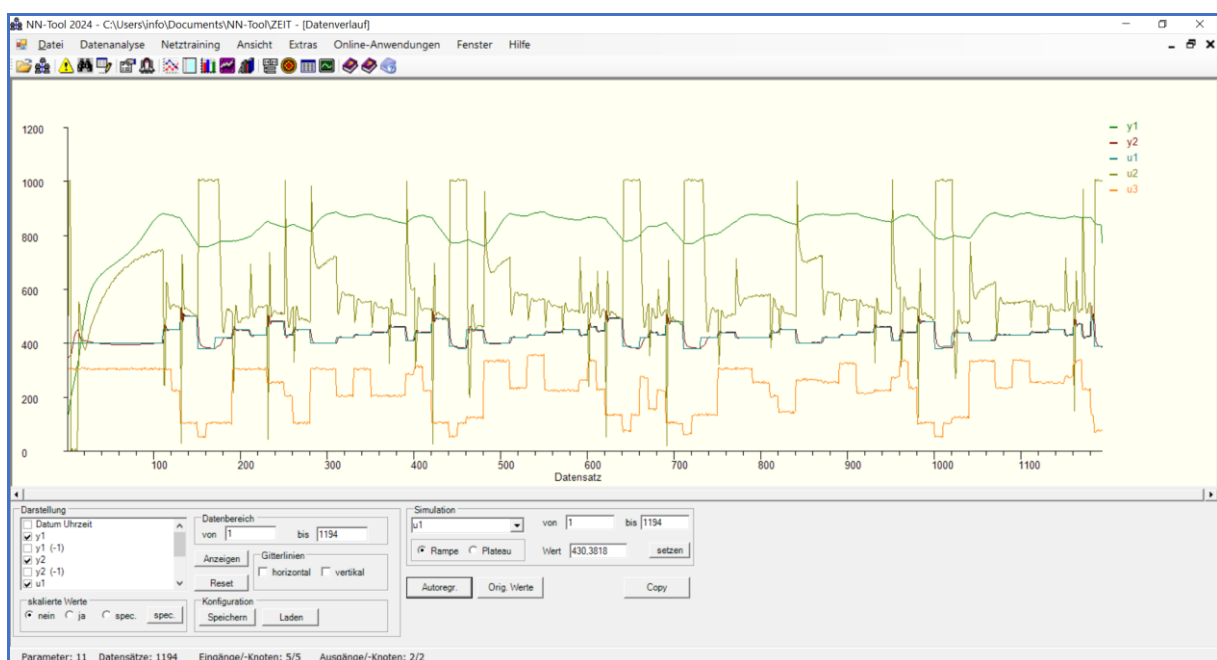


Im Fall von autoregressiven Modellen sollte ein entsprechender Stabilitätstest mit der Taste Autoregr. durchgeführt werden (siehe unten). Nach Bestätigen der Meldung, erscheint das folgende Fenster:

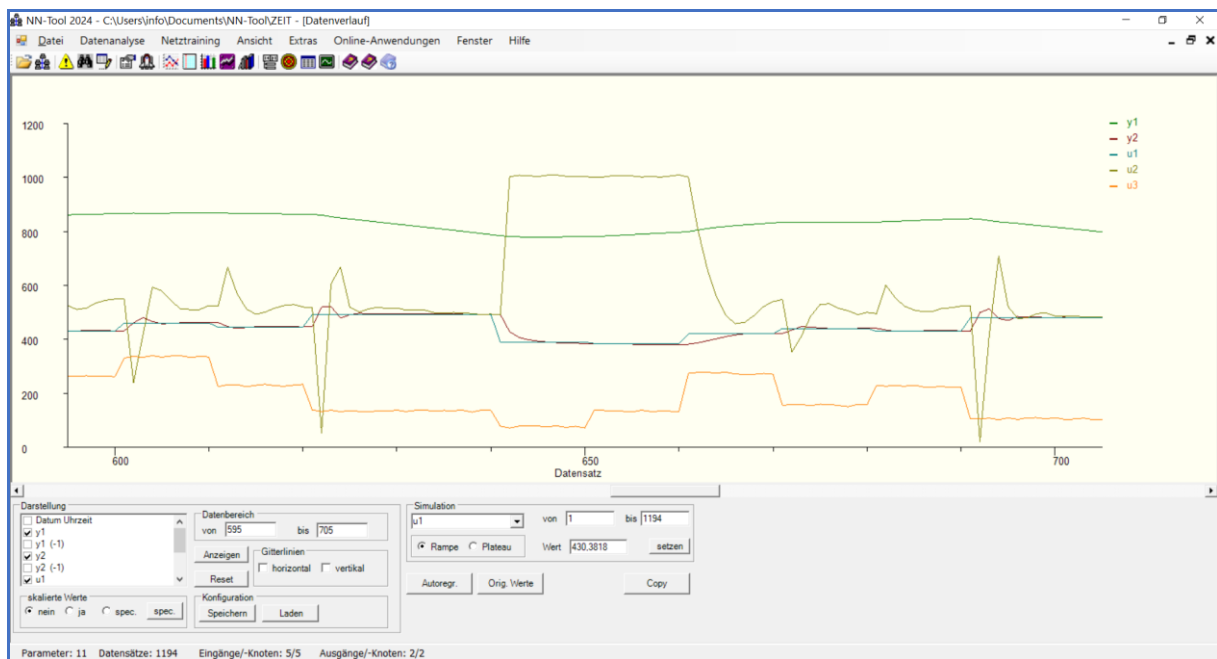


Das Fenster ist mit wenigen Ausnahmen identisch mit dem Fenster Datenverlauf. Bei den Ausgangsgrößen sind jedoch nicht die Meßwerte dargestellt, sondern die auf der Basis der Eingangsgrößen prognostizierten (berechneten) Werte. Diese Werte sind identisch mit den prognostizierten Werten der Verlaufsplots. Beachten Sie, daß diese Berechnung nicht-autoregressiv ist, da die notwendigen Eingangsgrößen hier ja als Meßgrößen komplett verfügbar sind. Solange es eben geht, sollte man natürlich immer nichtautoregressiv rechnen und auf Meßgrößen als Eingangsgrößen zurückgreifen.

Mit der Taste „Autoregr.“ kann nun die komplette Zeitreihe autoregressiv durchsimuliert werden:



Dabei ändert sich der Verlauf von  $y_1$  etwas, insbesondere in den ersten Datensätzen, die dem Anfahrvorgang des Prozesses entsprechen und in den Daten nur einmalig repräsentiert sind. Dies ist übrigens der Grund für die Abweichungen zwischen Lern- und Testset. Der Wert von  $y_2$  ändert sich fast gar nicht. Ggf. sollten die Eingangsgrößen aus der Grafik herausgenommen und die Skalierung auf „spec.“ gesetzt werden um den Effekt besser sehen zu können. Mit der Taste „Orig. Werte“ lässt sich wieder auf die ursprünglichen Daten zurückschalten, d.h. auf originale Eingänge und nichtautoregressiv prognostizierte Ausgänge. Der Vergleich weist hier auf eine auch für längere Simulationen hinreichend stabile Situation hin. Schneiden Sie nun mittels der rechten Maustaste einen Ausschnitt heraus, der den Bereich der Daten zwischen Satz 600 und 700 umfasst.



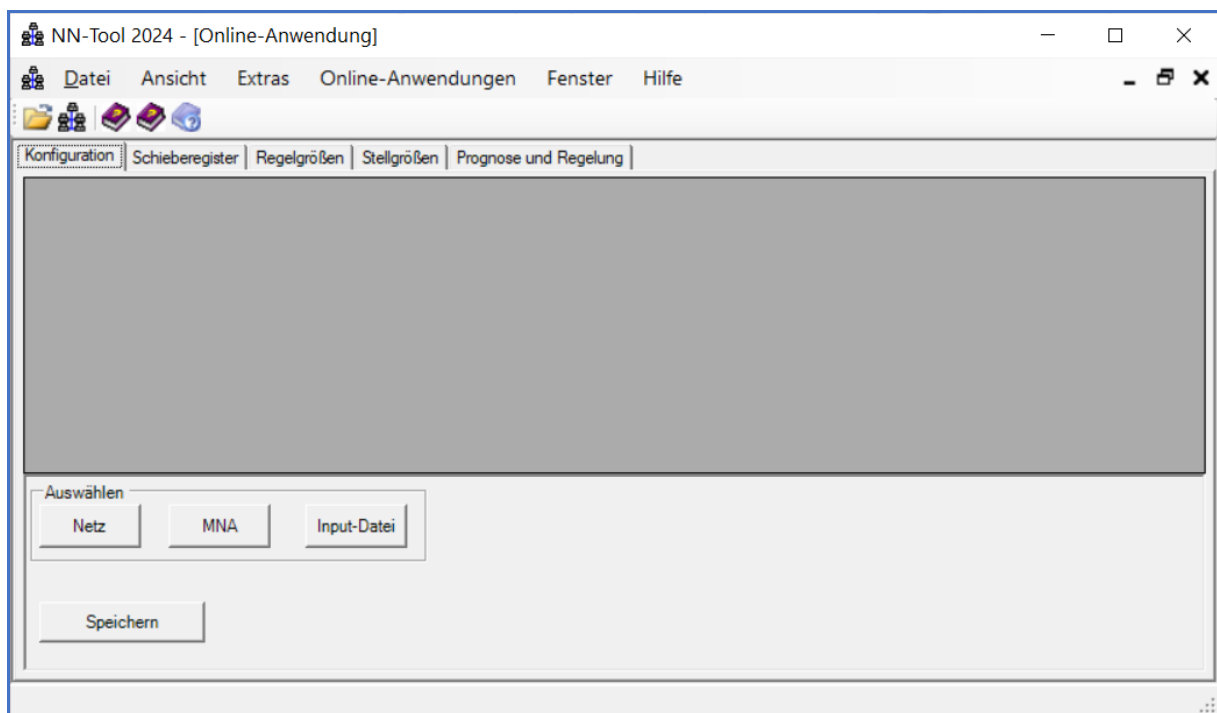
Sie können nun für die Eingangsgrößen des Modells neue Verläufe vorgeben und es wird dann der entsprechende Verlauf der Ausgangsgrößen berechnet. **Hinweis:** die Vorgabe erfolgt immer auf der Basis der Totzeit-0-Werte, selbst wenn diese gar nicht (wie hier) in die Modellierung eingegangen sind. Die entsprechenden verschobenen Werte (z.B.  $u_1$  (-1) ) werden dann automatisch bereitgestellt. Die Vorgabe kann durch Spezifikation eines von-bis-Bereichs und eines Wertes für den ausgewählten Parameter erfolgen. Dabei kann ein plateau- oder rampenförmiger Verlauf gewählt werden. Die Vorgaben können auch über die linke Maustaste erfolgen. Mittels der Taste „Orig. Werte“ können die Vorgaben zurückgesetzt werden.

## Anhang 11: Online-Anwendungen

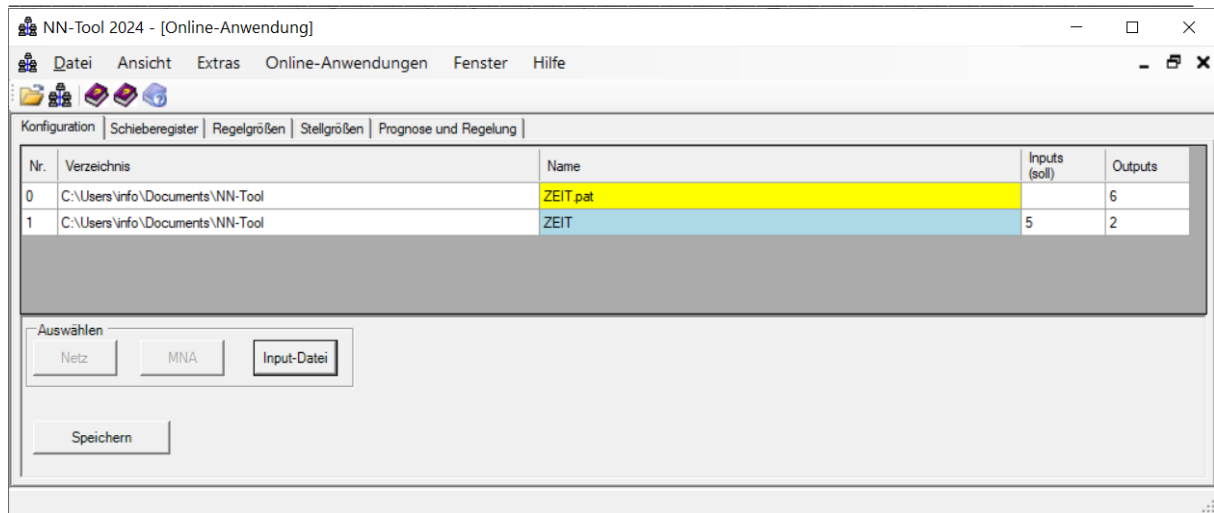
Zur Realisierung von Online-Anwendungen, z.B. Softsensor-Anwendungen oder modellgestützten Regelungen, auf der Basis von NN-Tool Prozessmodellen, steht die zusätzlich lieferbare Softwarekomponente „**NNControlServer**“ (ActiveX-Library) zur Verfügung. Diese Komponente stellt u.a. die folgenden Funktionalitäten zur Verfügung:

- Einbindung in beliebige Programme
- Laden einer entsprechenden NN-Tool Applikation
- Prognose einzelner Datensätze für definierte Zeitintervalle in der Zukunft
- Bereitstellung eines Schieberegisters für Totzeiten
- Glättungsfilter für verrauschte Eingangsdaten
- Offsetkompensation von Modellfehlern, falls zeitversetzte Ausgangsgrößen verfügbar
- Berechnung von Stellgrößen für Regelungsanwendungen („**Model Based Predictive Control**“)

Die Anwendung beruht nicht unmittelbar auf einem NN-Tool Netz, sondern auf einer sogenannten .mna-Datei (siehe auch Anhang 16). Die Konvertierung und Anpassung eines NN-Tool-Netzes für die spezifischen Belange von Online-Anwendungen erfolgt in NN-Tool selbst. In dieser Umgebung kann die Funktionalität dann auch direkt mittels Simulationen getestet werden. Die Vorgehensweise wird im Folgenden am Beispiel der mitgelieferten Anwendung **ZEIT** erläutert (vgl. Kapitel 8. Zeitreihen). Erstellen Sie die Anwendung **ZEIT** wie in Kapitel 8 beschrieben. Aktivieren Sie im Hauptmenu „**Online-Anwendungen**“ den Menüpunkt „**Neue Online-Anwendung**“



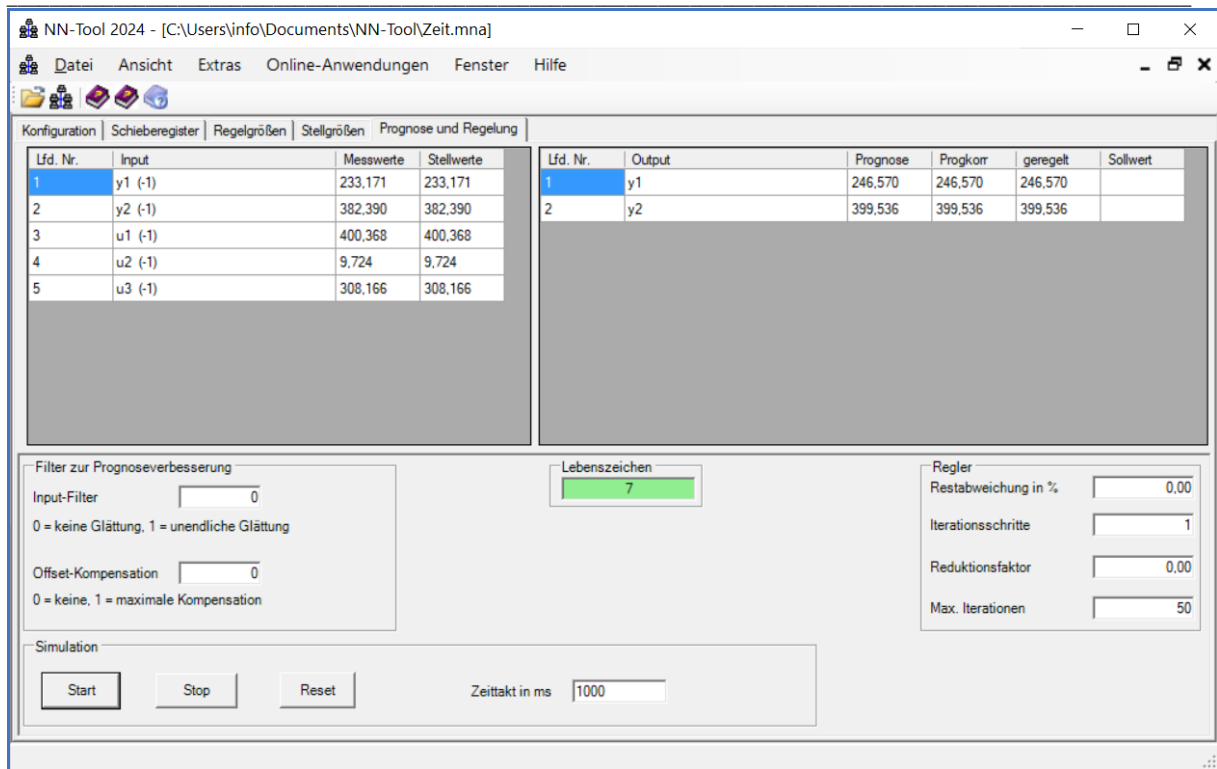
Eine Online-Anwendung basiert standardmäßig auf einer Multinetzanwendung (genauer: ist eine Erweiterung). Liegt diese noch nicht vor, kann aus einem einzelnen NN-Tool-Netz eine neue Multinetzanwendung konfiguriert werden. Betätigen Sie die Taste „**Netz**“, wählen Sie das NN-Tool-Netz „**ZEIT.dsc**“ aus



und drücken Sie die Taste Speichern. Speichern Sie die Anwendung unter dem Namen „**Zeit.mna**“ ab.

Damit ist eine Minimalkonfiguration abgeschlossen und das Netz für eine reine Prognoseanwendung unter dem Namen „**Zeit.mna**“ fertig konfiguriert. Die Konfiguration setzt voraus, dass der an die Online-Anwendung zu übertragende Datensatz exakt das Format der in Zeile „0“ angegebenen Datei („**Inputfile**“) besitzt. Standardmäßig wird dabei das Datenformat der zugehörigen .pat-Datei verwendet. Falls die Anwendung Totzeiten aufweist (wie hier), wird auch ein entsprechendes Schieberegister vorgehalten (siehe Registerkarte „**Schieberegister**“). Wechseln Sie nun zur Registerkarte „**Prognose und Regelung**“ und drücken Sie die „**Start**“-Taste. Damit starten Sie die **Simulation der Online-Anwendung**. Hierbei kommen die Daten nicht aus einer externen Anwendung (deshalb auch „Simulation“), sondern es werden die Datensätze der Datei in Zeile 0 („**Inputfile**“) nacheinander eingelesen, an die Online-Komponente übertragen und die prognostizierten Werte in die Datei „<Anwendung>\_output.txt“ (hier also „**Zeit\_output.txt**“) geschrieben. In der späteren Online-Anwendung findet der Datenaustausch dann selbstverständlich mit dem aufrufenden Programm statt. Die linke Seite des Bildes zeigt unter „**Messwerte**“, diejenigen Werte, die vom aufrufenden Programm an die Komponente geliefert werden. Die Werte auf der rechten Seite mit der Bezeichnung „**Prognosewerte korr.**“ werden von der Komponente zurückgeliefert.

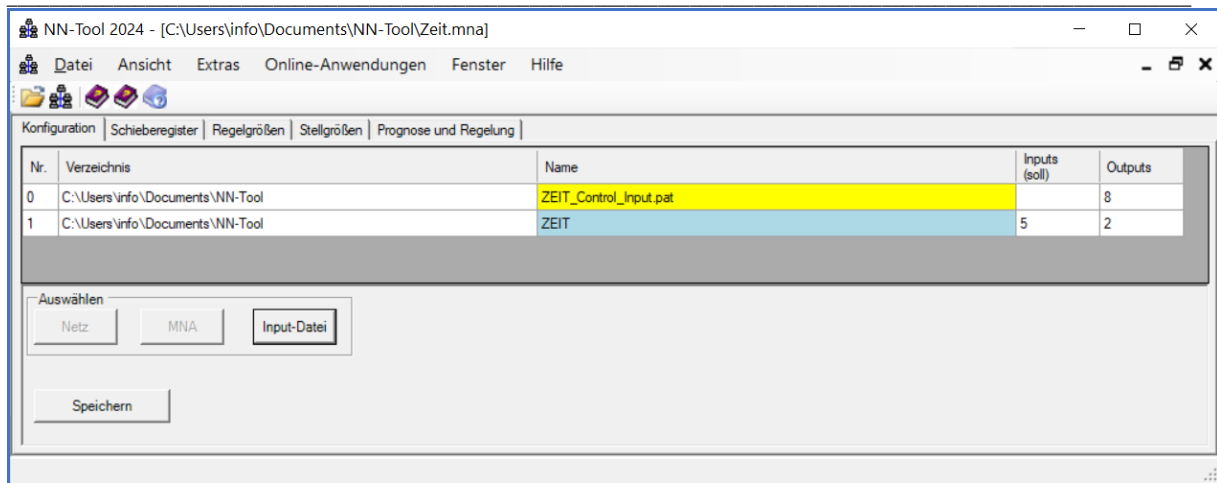




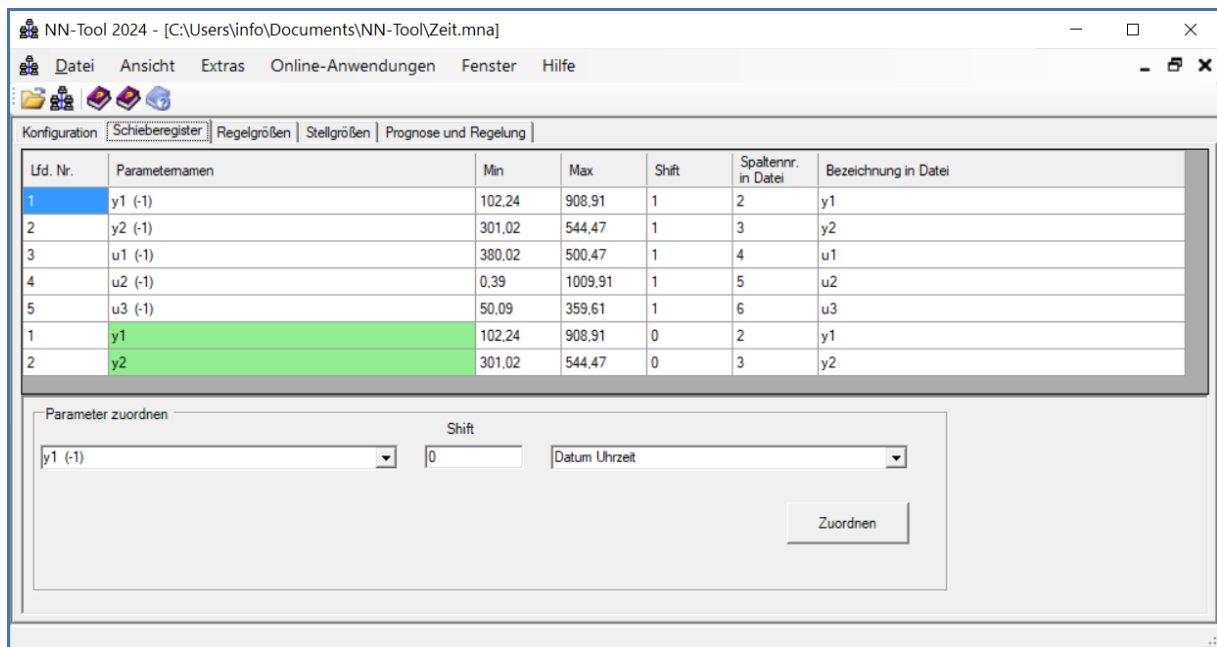
„**Lebenszeichen**“ liefert eine Zahl größer Null (zwischen 1 und 10000), wenn die Datenübertragung und Auswertung erfolgreich war, bzw. einen negativen Wert (Errorcode) sowie ein rotes Feld, falls ein Fehler aufgetreten ist.

Im Folgenden werden wir nun aus der einfachen Prognoseanwendung eine modellgestützte Regelung machen. Drücken Sie die Taste „**Stop**“, anschließend „**Reset**“. Zunächst müssen wir für die Regelung dafür sorgen, dass nun zusätzlich Sollwerte für die Ausgangsgrößen des Netzes bereitgestellt werden. Diese Werte müssen ebenfalls mit den sonstigen Daten an die Online-Komponente übertragen werden. Dazu benötigen wir eine entsprechend erweiterte Inputdatei, welche die Sollwerte bereitstellt. Wechseln Sie zur Registerkarte „**Konfiguration**“ und drücken Sie die Taste „**Input-Datei**“. Wählen Sie die Datei „**ZEIT\_Control\_Input.pat**“ aus dem Verzeichnis "**Dokumente\NN-Tool**" aus. Die Datei verfügt im Vergleich zu „**ZEIT.pat**“ noch über die Spalten „**Soll\_y1**“ und „**Soll\_y2**“.

**Hinweis:** Die Datei verwendet das Komma als Dezimaltrennzeichen. Falls Sie den Dezimalpunkt an Stelle des Dezimalkommas verwenden, müssen Sie in der Datei alle Kommata mit Hilfe eines Editors durch Punkte ersetzen.

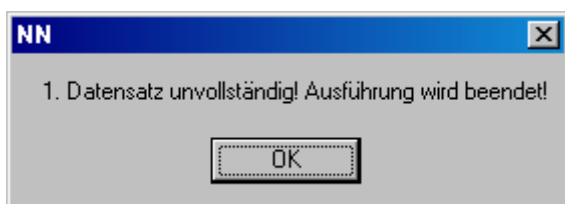


Da dem System jetzt eine neue Eingabedatei zugeordnet wurde, ist nicht automatisch mehr sichergestellt, dass die Eingänge richtig mit Werten beliefert werden. Wechseln Sie deshalb zur Registerkarte „**Schieberegister**“:



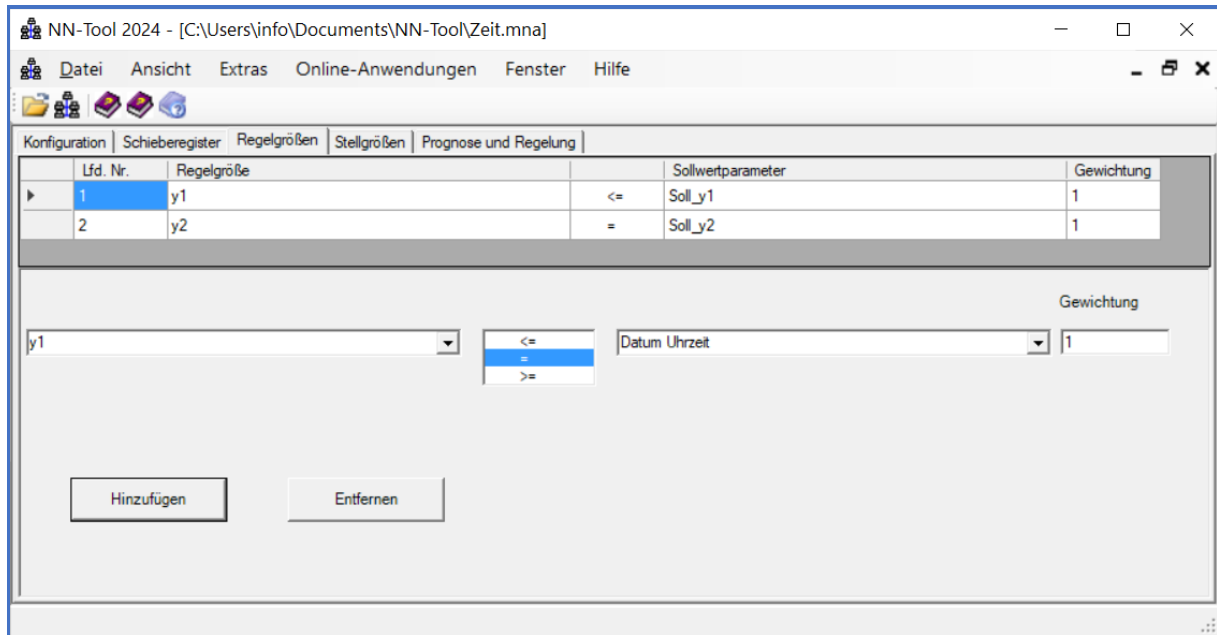
Wie man an den Parameterbezeichnungen und den Shifts (d.h. Zeitverschiebungen) erkennt, wurde die Zuordnung von NN-Tool automatisch richtig vorgenommen.

**Hinweis:** Dies funktioniert keineswegs immer; in diesen Fällen ist die Zuordnung manuell vorzunehmen (siehe folgendes Beispiel). Die Zuordnung kann dann über die beiden Listenfelder und die Taste „Zuordnen“ erfolgen. Falls keine korrekte Zuordnung vorgenommen wird, erfolgt bei der Simulation die Fehlermeldung:

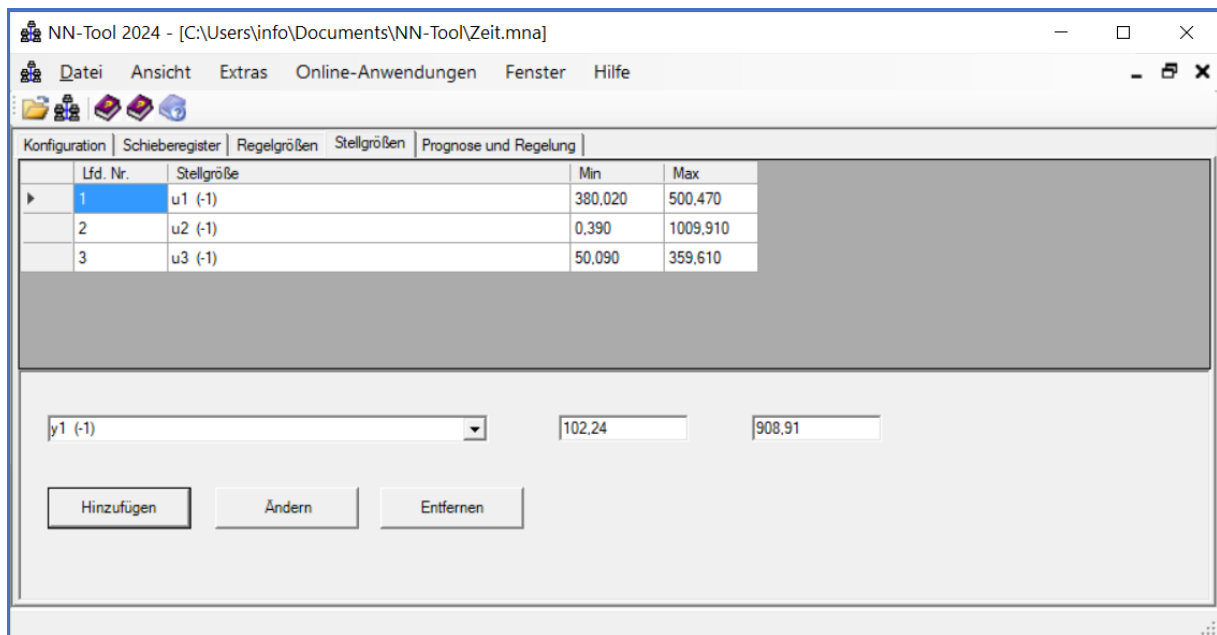


Hier ist die Zuordnung jedoch korrekt.

Wechseln Sie nun auf die Registerkarte „**Regelgrößen**“ und spezifizieren Sie „ $y1 \leq \text{Soll\_y1}$ “ und „ $y2 = \text{Soll\_y2}$ “.



Als nächstes sind die Stellgrößen festzulegen, d.h. die Größen, die von der Regelung zur Erreichung der Regelziele verändert werden dürfen. Wechseln Sie zur Registerkarte „**Stellgrößen**“ und legen Sie die Stellgrößen u1, u2, u3 wie abgebildet fest:

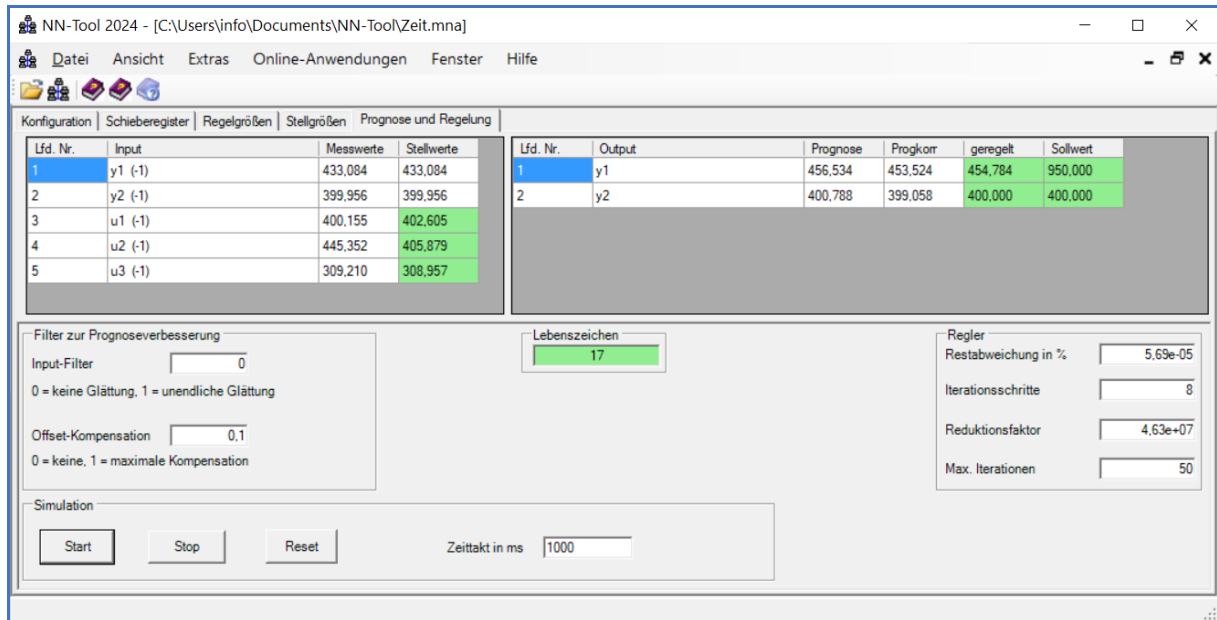


Wechseln Sie zur Registerkarte „**Konfiguration**“ und speichern Sie die Anwendung unter dem Namen **Zeit.mna** ab. Wechseln Sie nun zur Registerkarte „**Prognose und Regelung**“. Hier können Sie die folgenden weiteren **Regleroptionen** einstellen:

- **Inputfilter:** Filter erster Ordnung zur Glättung der Eingangsdaten.
- **Offsetkompensation:** ermöglicht Korrektur des Modellfehlers durch paralleles Auswerten des Modells.

- **Max. Iterationen:** Die Stellgrößen werden durch ein iteratives Verfahren berechnet. Je größer der Wert desto genauer die Rechnung, desto länger dauert es aber auch.

Zusätzlich kann noch der Zeittakt der Simulation in Millisekunden festgelegt werden. Setzen Sie die Offsetkompensation auf den Wert 0,1, bleiben Sie ansonsten bei den Standardeinstellungen, **speichern Sie die Konfiguration ab** und drücken Sie Start.



Im linken Teil werden nun die berechneten Stellgrößen angezeigt (grün unterlegt), im rechten Teil - neben den Prognosewerten des Modells - die korrigierten Prognosewerte, die sich nun auch von den Prognosewerten bedingt durch die Offsetkompensation unterscheiden. In der Spalte „**geregelt**“ werden diejenigen Ausgangswerte angezeigt, die sich nach Meinung des korrigierten Modells im nächsten Zeittakt ergeben, falls die Stellgrößen auf die Anlage aufgeschaltet werden. Zusätzlich werden die folgenden Informationen bereitgestellt, die die Regelgüte beschreiben:

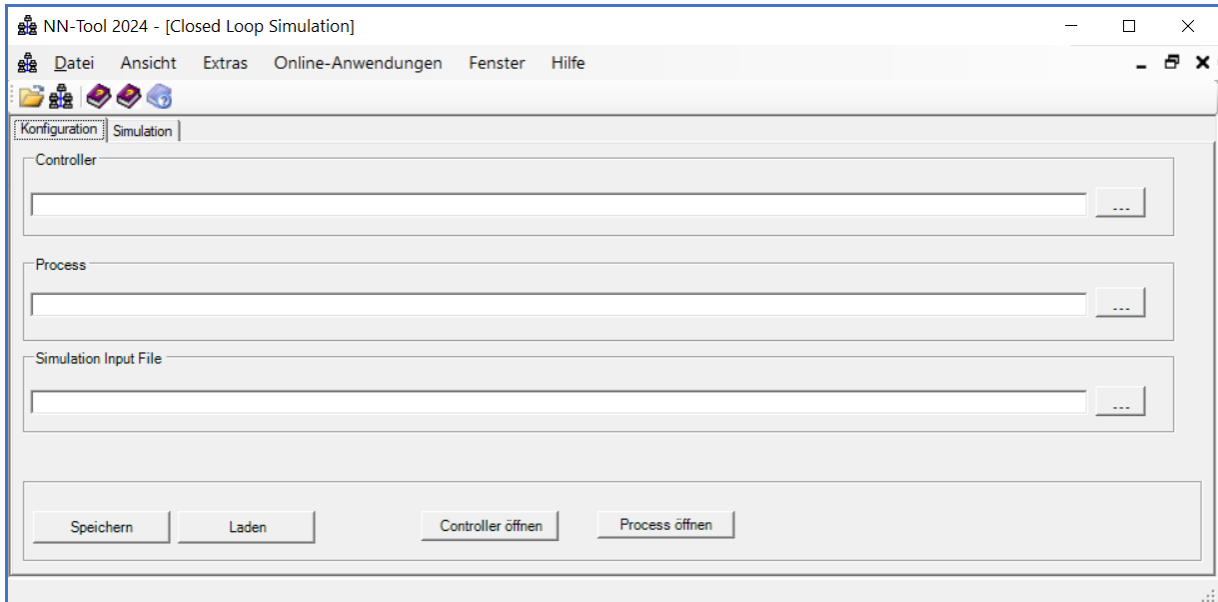
- **Restabweichung in %:** Gibt an, wie genau die Sollwerte erreicht werden konnten. Sollte möglichst klein sein. (Beachten Sie, dass für y1 nur eine  $\leq$  Bedingung gefordert war.)
- **Iterationsschritte:** Zahl der benötigten Rechenschritte
- **Reduktionsfaktor:** Gibt an, um welchen Faktor die ursprüngliche Regelabweichung (d.h. diejenige, bei Verwendung der Messdaten) reduziert werden konnte. Sollte möglichst groß sein.

Die Simulation schreibt nun die Originaldatensätze und zusätzlich die korrigierten Prognosewerte und die Stellgrößen in die Datei „<Anwendung>\_Output.txt“ (hier also „Zeit\_Output.txt“). Diese Datei darf nicht durch andere Anwendungen blockiert werden (ggf. Neustart von NN-Tool, im Extremfall von Windows).

**Hinweis:** Zu dieser NN-Tool-Komponente ist eine **.NET-Runtimebibliothek** verfügbar, um die Anwendung mit externen Applikationen (insbesondere Prozessleitsystemen) zu koppeln (siehe Produktübersicht).

## Anhang 12: Closed Loop Simulation

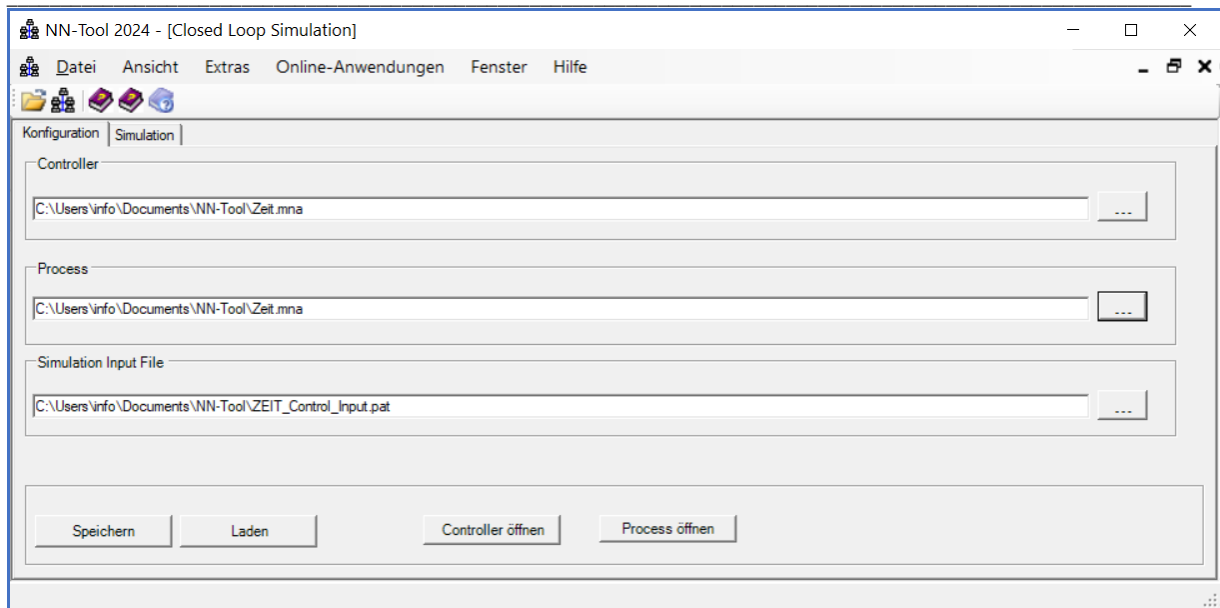
Die Anwendung „**Closed Loop Simulation**“ im Hauptmenu „Online-Anwendungen“ ermöglicht es eine modellgestützte Regelung (siehe vorhergehenden Anhang 11: Online-Anwendungen) in einem geschlossenen Regelkreis zu testen:



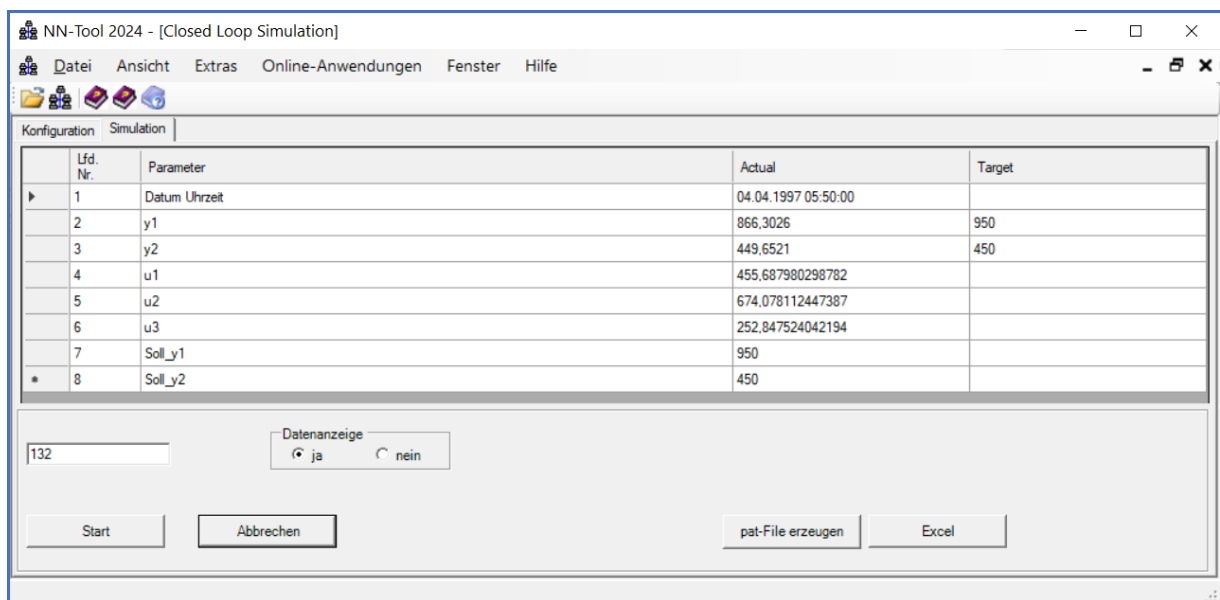
Wählen Sie nun für den Regler („Controller“) die im vorhergehenden Kapitel entworfene modellgestützte Regelung „**zeit.mna**“. Wählen Sie die gleiche Anwendung für die Simulation des Prozesses („Process“).

### Hinweise:

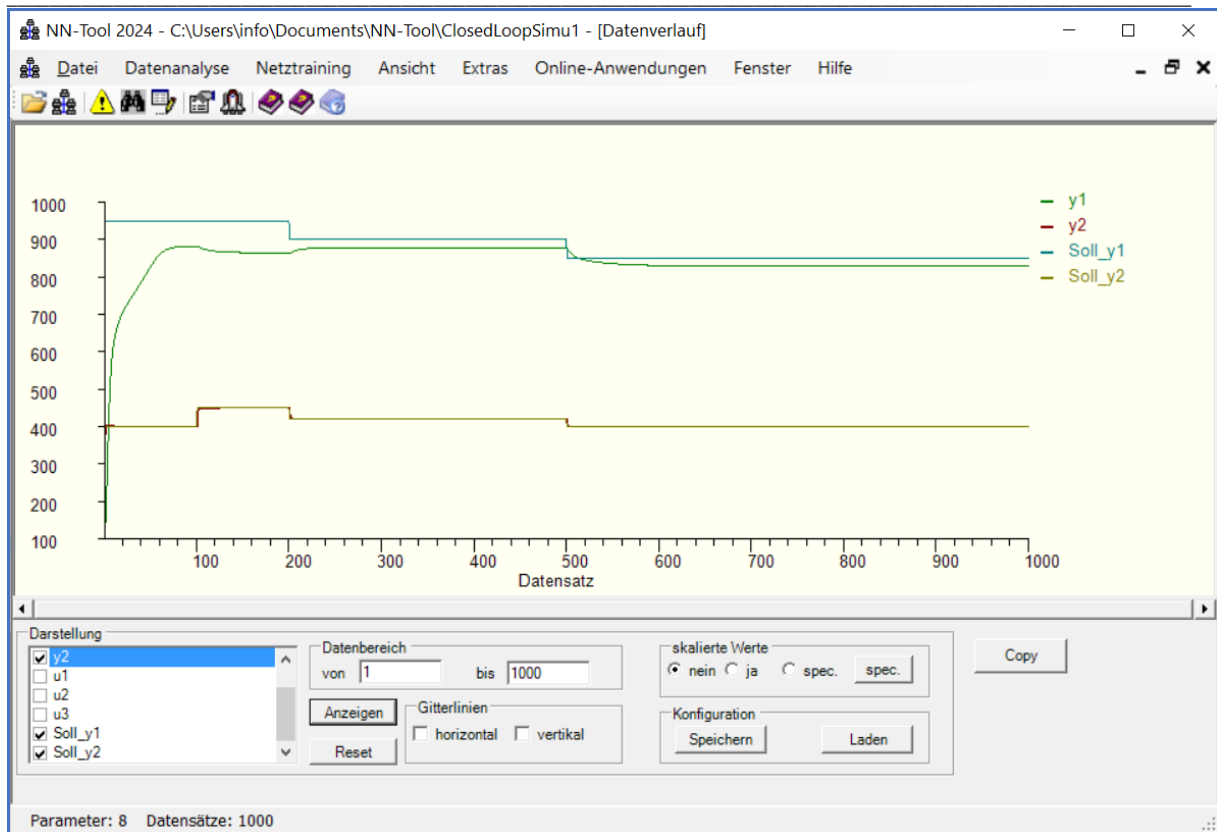
- Im Allgemeinen wird man **unterschiedliche Anwendungen** (basierend auf **verschiedenen Netzen**) für Controller und Process wählen um den Einfluss von Modellfehlern („**Model Plant Mismatch**“) sowie den Effekt der **Offset-Kompensation** studieren zu können. Beide Netze sollten jedoch aus Kompatibilitätsgründen mit der gleichen .pat-Datei erzeugt worden sein.
- Für die .mna-Anwendung für den **Prozess** müssen weder Stell- noch Regelgrößen spezifiziert werden.
- Als Basis für die Simulation („Simulation Input File“) wird standardmäßig die Inputdatei der Controller-Anwendung verwendet.



Speichern Sie die Konfiguration ab, wechseln Sie zur Registerkarte „Simulation“ und drücken Sie den Button „Start“.



Mittels der Taste „Excel“ können sämtliche Ergebnisse in eine Excel-Mappe übertragen werden. Effizienter ist jedoch die Benutzung der Taste „pat-File erzeugen“. Die Simulationsergebnisse werden als .pat-File abgespeichert, der anschließend geladen und analysiert wird:



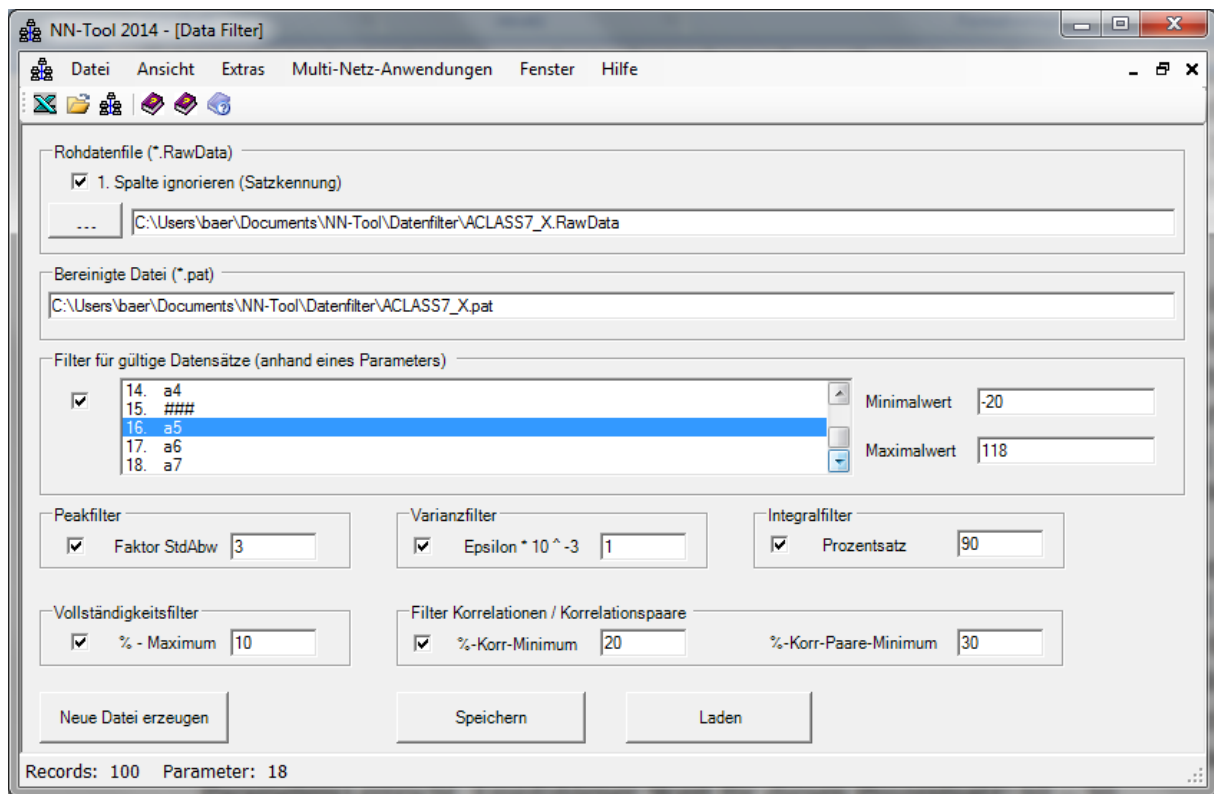
Man erkennt praktisch perfektes Regelverhalten, was aber auch kein Wunder ist, da das Reglermodell exakt mit dem Prozess übereinstimmt (Beachte:  $y1 \leq Soll\_y1$ ). Hier würde im Übrigen auch keine Offset-Kompensation benötigt.

Variieren Sie nun Regler- und Prozessmodell um die Effekte von „Model-Plant-Mismatch“ zu studieren.

**Wichtiger Hinweis: Z.Z. ist die Closed Loop Simulation auf Modelle beschränkt, bei denen die Inputs genau einen Zeittakt vor den Outputs liegen, d.h. Inputs auf Takt „-1“, Outputs auf „0“ (d.h. Outputs ohne Verschiebung).** Ggf. müssen die Rohdaten unter Excel geeignet gegeneinander verschoben werden.

## Anhang 13: Datenfile filtern

Diese Komponente im Hauptmenu Datei ermöglicht es, aus einer stark durch Mess-, Sensor- und Eingabefehler belasteten Rohdatendatei eine bereinigte Datei zu erstellen. Die Rohdatendatei muss dabei dem Format einer .pat-Datei entsprechen, jedoch die Dateierendung „**RawData**“ tragen. Die bereinigte Datei ist dann vom Typ „.pat“ und kann unmittelbar weiterverarbeitet werden.



Die Komponente führt die folgenden Bereinigungs- und Filterschritte in der angegebenen Reihenfolge durch. Dabei kann die erste Datenspalte, die in vielen Fällen eine Satznummer (z.B. Datum/Uhrzeit) enthält, von der Filterung ausgenommen werden:

- Alle nichtnumerischen Einträge werden zunächst entfernt, d.h. durch einen leeren Wert ersetzt.
- **Filter für gültige Datensätze:** Eliminiert **Datensätze**. Anhand eines ausgewählten Parameters (hier Parameter „a5“) können unzulässige Betriebs- oder Prozesszustände komplett entfernt werden. D.h. jeder **Datensatz**, bei dem der ausgewählte Parameter einen Wert außerhalb des durch den gesetzten Minimalwert und Maximalwert definierten Bereichs annimmt, wird durch lauter Leerwerte in allen Parametern ersetzt. Dies gilt auch, falls der ausgewählte Parameter im aktuellen Datensatz gar nicht vorhanden ist. Im dargestellten Beispiel macht es ggf. Sinn den angezeigten Minimalwert des Parameters „a5“ auf einen Wert von 42 zu ändern, da andernfalls ohnehin kein zulässiger Betriebspunkt vorliegt.
- **Peakfilter:** Eliminiert **Ausreisserwerte**. Jeder **Datenwert**, der um mehr als x Standardabweichungen vom Mittelwert des gegebenen Parameters abweicht, wird durch einen leeren Wert ersetzt (entfernt). Der Faktor x kann vorgegeben werden. Empfohlener Bereich 3 – 4.



- **Varianzfilter:** Eliminiert komplette **Parameter** (Datenspalten). Jeder Parameter, der eine zu geringe Varianz in seinen Werten aufweist, wird komplett entfernt. Als Kriterium gilt, dass die Standardabweichung mindestens  $\text{Epsilon} \times \text{Betrag des Mittelwerts}$  sein muss. Ein Wert für Epsilon kann vorgegeben werden. Empfohlener Bereich 1 – 10. Der Wert wird dann automatisch mit  $10^{-3}$  multipliziert. D.h. ein Vorgabewert von 2 besagt, dass alle Parameter eliminiert werden, bei denen die Standardabweichung nicht einmal 2 Promille des Mittelwerts erreicht (genauer: Betrag des Mittelwerts erreicht).
- **Integralfilter:** Eliminiert komplette **Parameter** (Datenspalten). Der Filter eliminiert alle integralen Größen des vorliegenden Datensatzes. Ein Parameter wird als integrale Größe angesehen, falls die Anzahl der Wertänderungen in positive Richtung die Anzahl der Änderungen in negative Richtung weit übersteigt (bzw. umgekehrt). Falls mehr als x Prozent aller auftretenden Wertänderungen nur in eine Richtung zeigen, wird der Parameter als integrale Größe angesehen und eliminiert. Empfohlener Wert für diesen Prozentsatz: 90.
- **Vollständigkeitsfilter:** Eliminiert komplette **Parameter** (Datenspalten). Der Filter eliminiert alle Größen des vorliegenden Datensatzes, die nicht hinreichend oft mit Werten belegt sind (d.h. zu viele fehlende Messwerte aufweisen). Zunächst wird dazu bestimmt, wie viel vollständige Werte derjenige Parameter aufweist, der am häufigsten gemessen worden ist. Ein Parameter wird dann eliminiert, wenn er nicht einmal x Prozent dieses Wertes (d.h. der Anzahl der Werte des am häufigsten gemessenen Parameters) erreicht. Empfohlener Wert für diesen Prozentsatz: 10 – 30.
- **Filter Korrelationen / Korrelationspaare:** Eliminiert komplette **Parameter** (Datenspalten). Der Filter eliminiert alle Größen des vorliegenden Datensatzes, die entweder eine zu geringe Korrelation mit allen anderen Parametern aufweisen (genauer Betrag des Korrelationskoeffizienten) oder bei denen zu wenige mit anderen Parametern gemeinsame Datensätze zur Berechnung der Korrelation („Korrelationspaare“) vorliegen. Das erste Kriterium „%-Korr-Minimum“ verlangt, dass der Betrag des Korrelationskoeffizienten mit mindestens einem anderen Parameter einen Wert von mindestens x Prozent aufweist. Empfohlener Wert 10 – 30. Das zweite Kriterium „%-Korr-Paare-Minimum“ verlangt, dass mindestens x-Prozent mal so viele Korrelationspaare vorliegen wie bei dem Parameter, der die meisten Korrelationspaare aufweist. Empfohlener Wert 20 – 50.
- **Abschlussfilter:** Wird immer ausgeführt. Eliminiert komplette **Parameter** (Datenspalten). Der Filter eliminiert alle Größen des vorliegenden Datensatzes, die keine Messwerte (mehr) aufweisen oder immer nur den gleichen Messwert aufweisen.

#### Hinweise:

- Es werden nur die aktivierten Filter durchgeführt (Ausnahme Abschlussfilter). Jeder Filter wirkt auf die Werte, welche die vorangegangenen Filteroperationen übriggelassen haben. Die Filteroperationen werden in der oben angegebenen Reihenfolge durchgeführt.
- Die Funktionalität ist derzeit auf Anwendungen mit rein numerischen Daten beschränkt.
- Die Filterung wird durch Drücken der Taste „Neue Datei erzeugen“ gestartet. Nach Fertigstellung öffnet NN-Tool die bereinigte .pat-Datei.

**Achtung:** In der anschließend neu erstellten und geöffneten .pat-Datei sind Kennungen für die erste Spalte erneut (manuell) zu setzen!

## Anhang 14: Einsatz Neuronaler Netze unter Excel / NN-Tool Excel Add-In

Sie können unter Verwendung des „NN-Tool Excel-Add-Ins“ (**separat lieferbare Zusatzkomponente**) neuronale Netze, die Sie mit NN-Tool erstellt haben, in Excel-Mappen einsetzen. Die Netze verhalten sich dann wie eine Excel-Standard-Funktion (z.B. die Summenfunktion).

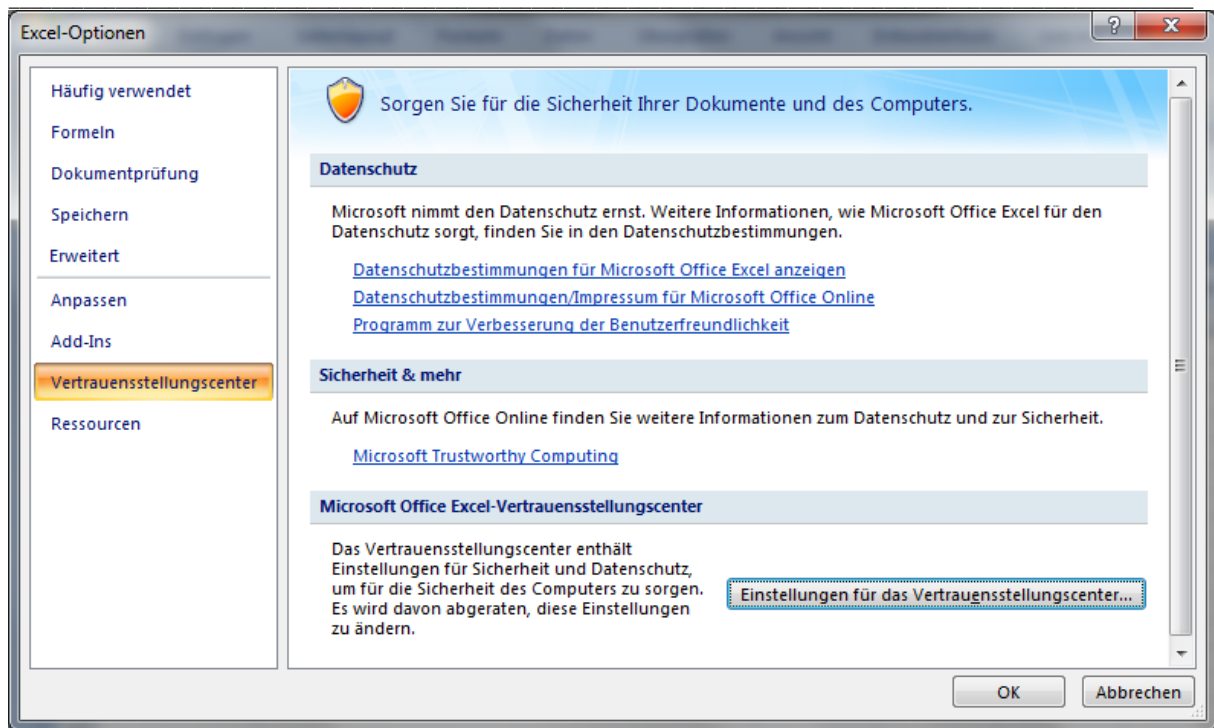
Das Excel-Add-In für NN-Tool kann unabhängig von einer auf der betreffenden Maschine installierten NN-Tool Version genutzt werden. **Bequemer ist jedoch die Erstellung und Verwaltung von entsprechenden Excel-Mappen aus NN-Tool heraus (für Excel-Versionen ab Excel 2007).** Diese Vorgehensweise wird im Folgenden beschrieben.

Starten Sie NN-Tool und laden Sie das fertiggestellte Netz „Test“. Wählen Sie nun im Menu „Extras“ den Menüpunkt „Excel-Add-In Anwendung erstellen“. Dabei versucht NN-Tool die folgenden beiden Operationen:

1. Installation des Excel-Add-Ins „Nnx\_V11“ (Dateiname nnx\_V11.xla) in Excel. Dieses Add-In stellt die notwendigen Funktionen zur Nutzung von NN-Tool Netzen in Excel bereit.
2. Erzeugung einer neuen Excel-Mappe „loadnet.xls“ im Excel-Startverzeichnis. Diese Mappe weist Excel an, die aufgelisteten Netze zu laden und für die Verwendung in Excel-Mappen vorzuhalten.

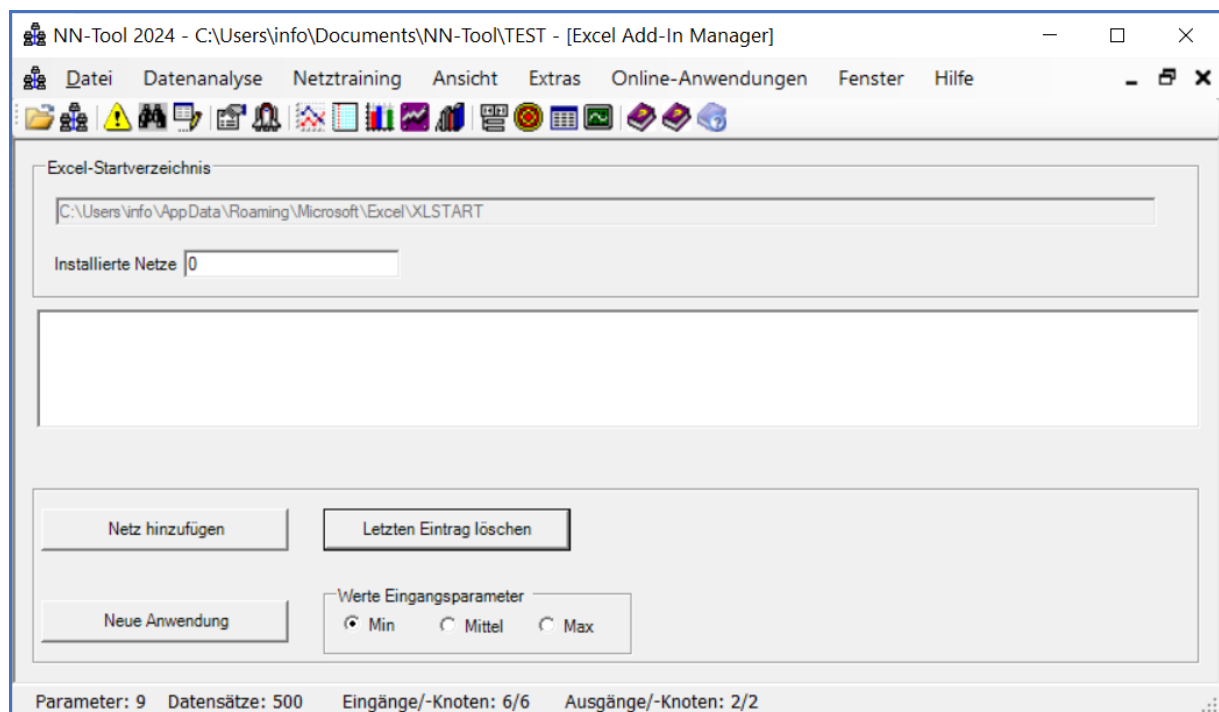
**Falls die beiden vorgenannten Operationen scheitern**, wurde das Excel-Add-In entweder nicht lizenziert oder es liegen nicht ausreichende Berechtigungen vor. Im ersten Fall kontaktieren Sie bitte Baermann Software.

Im zweiten Fall führen Sie die Installation bitte mit Administratorrechten durch. Zusätzlich sind unter Excel ggf. noch Einstellungen vorzunehmen. Gehen Sie wie folgt vor:  
Öffnen Sie bitte das Excel Vertrauensstellungscenter unter Excel/Optionen:



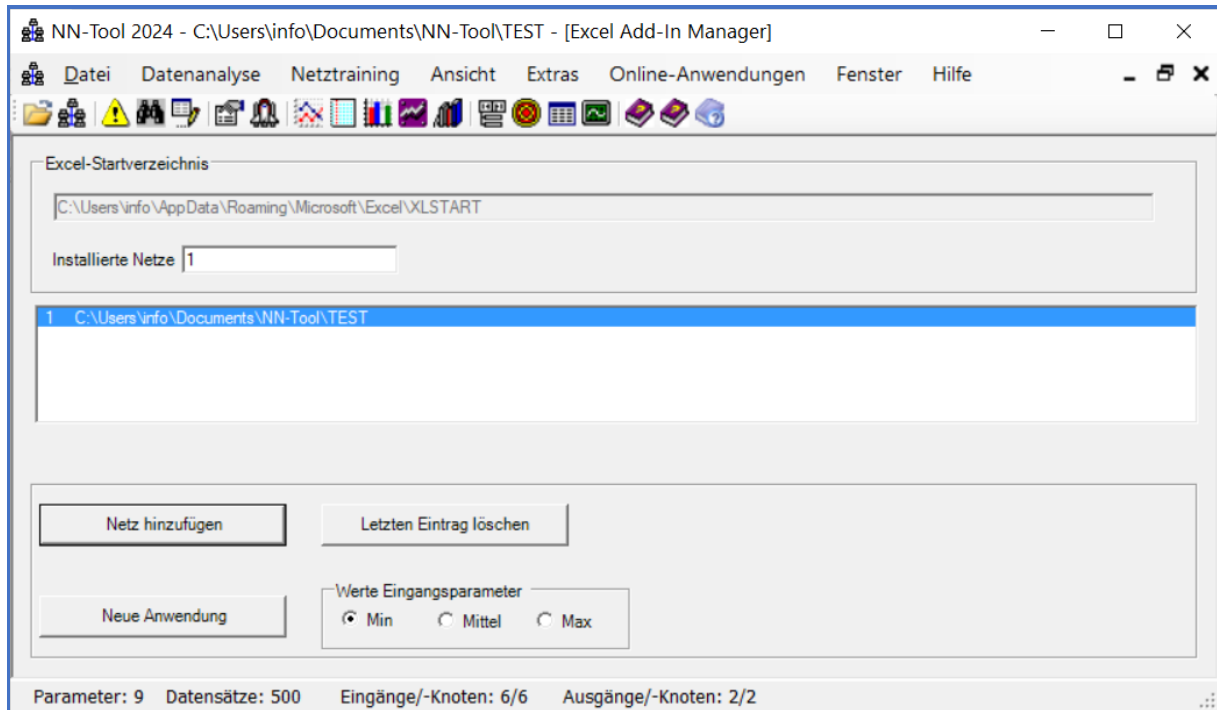
Wählen Sie „Einstellungen für das Vertrauensstellungscenter“ und stellen Sie die Einstellungen für „Add-Ins“, „ActiveX-Einstellungen“ und „Einstellungen für „Makros“ weniger restriktiv ein.

**Sind die beiden Operationen erfolgreich** erscheint das folgende Fenster



Ganz oben wird das Excel-Startverzeichnis angezeigt. Dieser Ordner ist im Windows-Explorer möglicherweise nicht sichtbar, da es sich um einen sogenannten „versteckten“ Ordner handelt. Über die Einstellung der Ordneroptionen kann das Verzeichnis sichtbar gemacht werden. In diesen Ordner kopiert die Anwendung die Mappe loadnet.xls. Zunächst enthält diese Mappe jedoch keine Netze. Installierte Netze: 0

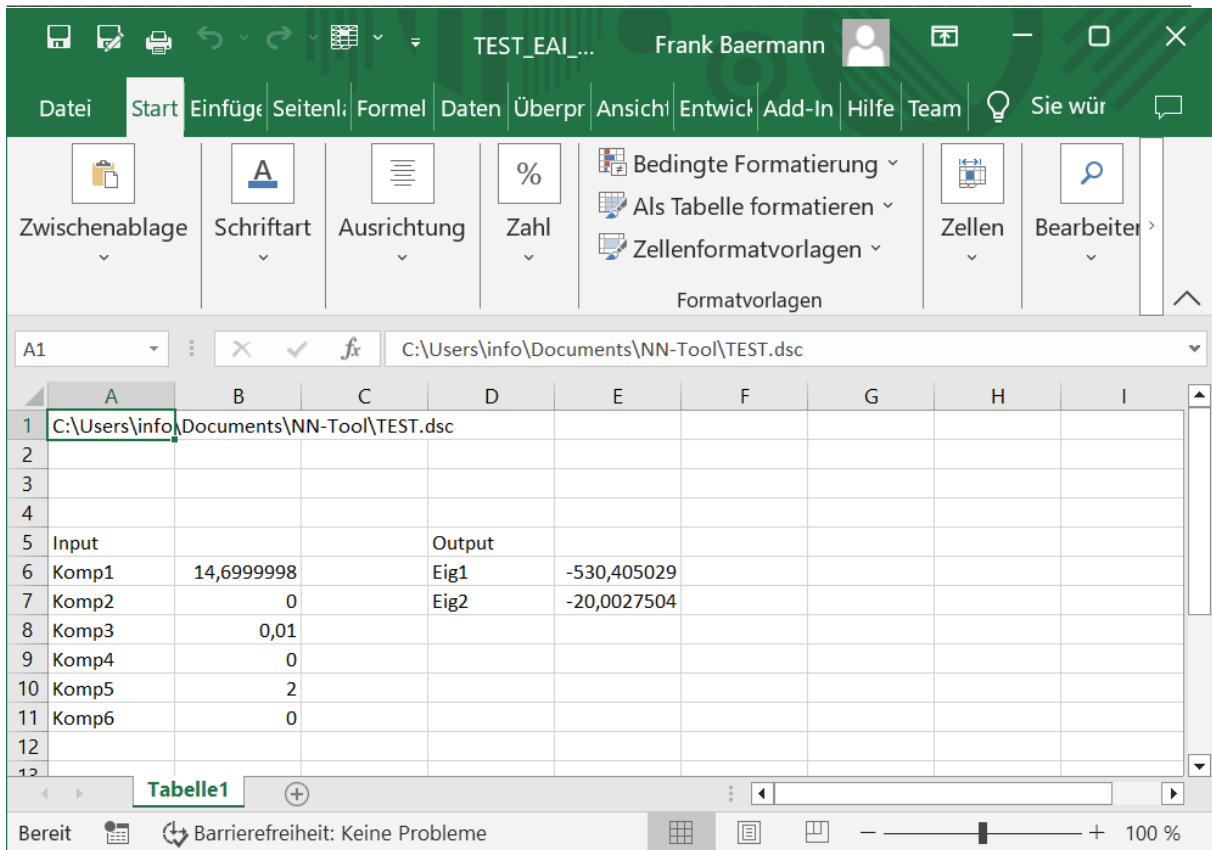
Fügen Sie nun mit der Taste „**Netz hinzufügen**“ das aktuelle Netz zur Liste hinzu.



Damit ist das Netz unter Excel nutzbar.

Wählen Sie nun eines der installierten Netze aus der Liste („in die Liste reinlicken“) aus um eine entsprechende Excel-Anwendung zu erzeugen und drücken Sie die Taste „**Neue Anwendung**“. Es wird eine neue Excel-Mappe erzeugt in der das betreffende Netz ausgewertet werden kann. Die Eingangsparameter werden dabei zunächst auf ihren Min/Max oder Mittelwert gesetzt. Bei Wahl des Minimums ist ein unmittelbarer Vergleich mit der Mischpult-Anwendung möglich (s.u.). Anschließend wird das Fenster geschlossen.

Die erstellte Mappe sieht wie folgt aus:



Zunächst ist das verwendete Netz angegeben. In den Spalten A und B befinden sich die Inputs für das Netz (als Voreinstellung sind hier die Minimalwerte vorgegeben), in der Spalte D die Namen der Outputs, und in der Spalte E werden die Ausgangsgrößen für die aktuellen Inputs direkt berechnet. In der Zelle E6 ist zum Beispiel die folgende Formel hinterlegt:

**=nnx(1;B6:B11;1)**

Dabei sind:

- nnx                                      Formel für Netzauswertung
- Erster Parameter:                      Laufende Nummer des Netzes in der loadnet-Mappe, hier Netz Nummer 1
- Zweiter Parameter:                      Zellbereich der Inputs, hier B6:B11
- Dritter Parameter:                      Nummer des Outputs, der in der aktuellen Zelle angezeigt werden soll

Die Mappe kann nun beliebig erweitert und modifiziert werden. Zur Nutzung einer solchen Mappe ist es auch nicht erforderlich, dass NN-Tool aktiv ist. Darüber hinaus können auf einem Tabellenblatt auch mehrere NN-Tool-Netze verwendet werden (nach Hinzufügen zur Liste der installierten Netze). Falls Sie die Outputzellen kopieren/verschieben, beachten Sie bitte, dass die Bezüge zu den Inputs relative Bezüge sind. Ggf. sollten Sie diese dann auf absolute Bezüge umstellen (Bereich markieren und Taste „F4“ drücken).

**Kontrolle:** Zusätzlich sollten Sie die berechneten Ergebnisse mit dem NN-Tool Mischpult vergleichen:

NN-Tool 2024 - C:\Users\info\Documents\NN-Tool\TEST - [Mischpult]

Datei Datenanalyse Netztraining Ansicht Extras Online-Anwendungen Fenster Hilfe

Mischpult Konfiguration

Eig1		-530,405
Eig2		-20,003

Komp1		14,7
Komp2		0
Komp3		0,01
Komp4		0
Komp5		2
Komp6		0

Prognose mit Fehlerabschätzung

Datensatz nach Excel übertragen

Satz speichern gespeichert 0 Neue Mappe

☒ Single ☐ Multi

Eingabemodus

☒ Schieberegler ☐ Tastatur

Datei

Öffnen Speichern

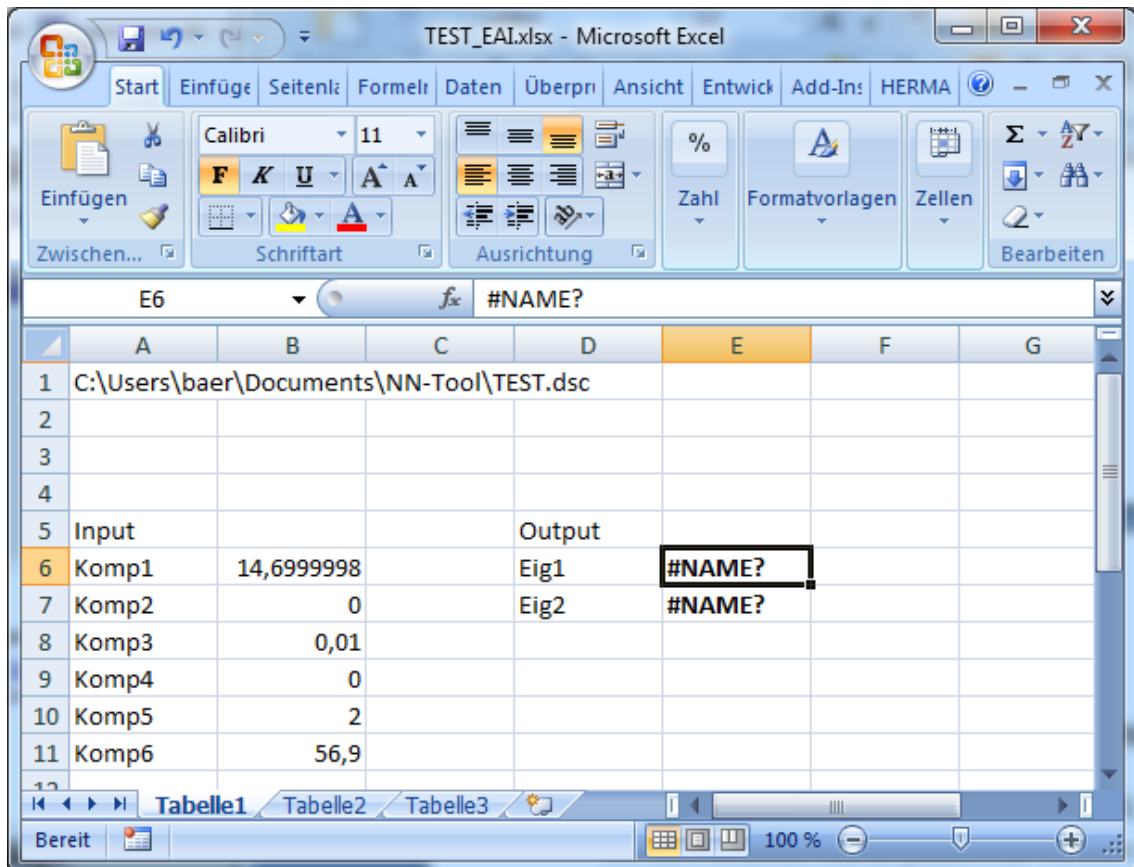
Parameter: 9 Datensätze: 500 Eingänge/-Knoten: 6/6 Ausgänge/-Knoten: 2/2

Wie man sieht stimmen die Werte überein!

### Nutzung unter älteren Excel-Versionen

Die Anwendung kann auch unter Excel 2003, ggf. auch unter älteren Versionen genutzt werden. Getestet wurde jedoch nur Version 2003. Dabei ergeben sich leichte Unterschiede zum bisher beschriebenen.

- Das Vertrauensstellungscenter bzw. die entsprechenden Add-In, ActiveX bzw. Makro-Einstellungen können sich bei älteren Excel-Versionen an anderer Stelle befinden.
- In der erzeugten Excel-Mappe erscheinen nicht direkt die berechneten Ausgangsgrößen in den betreffenden Zellen (hier E6 und E7), sondern es erscheint **#Name?** :



Die Formeln wurden jedoch trotzdem hinterlegt. Klicken Sie nun in die entsprechenden Zellen und drücken Sie die Eingabetaste. Alternativ können Sie auch in der Mappe unter „**Bearbeiten**“ / „**Suchen und Ersetzen**“ alle Gleichheitszeichen „=“ durch Gleichheitszeichen „=“ (ist wirklich so gemeint!) ersetzen. Dies aktiviert die Formeln.

**Hinweis:** Hierbei handelt es sich um einen ganz alten Excel-Bug, der erst in der Version Excel 2007 behoben wurde. Info: <http://support.microsoft.com/kb/291058/de>

## Anhang 15: Hauptkomponentenanalysen / PCATool

Bei bestimmten Anwendungen kann es vorkommen, dass die Eingangsparameter und/oder die Ausgangsparameter nicht unabhängig voneinander sind, sondern dass zwischen ihnen Beziehungen (Gleichungen) bestehen. Derartige Gleichungen reduzieren die Zahl der sogenannten Freiheitsgrade des Systems. Falls ein Problem 10 Eingangsgrößen besitzt zwischen denen 7 Gleichungen bestehen, reduziert sich die Zahl der Freiheitsgrade auf 3, d.h. das Problem besitzt effektiv nur 3 Dimensionen. Falls diese Gleichungen linear sind, lassen sich die genannten Freiheitsgrade über die Hauptkomponentenanalyse ermitteln.

PCATool ist eine eigenständig laufende Anwendung zur Hauptkomponentenanalyse ("Principal Component Analysis" - PCA), die bei der Bayer AG in Leverkusen entwickelt worden ist und zusätzlich bezogen werden kann. Da PCATool auf den gleichen Datenstrukturen wie NN-Tool aufbaut, bietet es für bestimmte Anwendungen ("Multi-Netz-Anwendungen" - MNA) eine sehr nützliche Ergänzung zu NN-Tool. Ein Beispiel finden Sie in der folgenden Dokumentation bei der Beschreibung von Multi-Netz-Anwendungen. PCATool kann jedoch auch völlig unabhängig von NN-Tool eingesetzt werden. Näheres entnehmen Sie bitte der Dokumentation zu PCATool.



## Anhang 16: Multi-Netz-Anwendungen - MNA

Multi-Netz-Anwendungen („MNA“) sind Anwendungen zu deren Beschreibung nicht ein einzelnes neuronales Netz eingesetzt wird, sondern eine Verschaltung mehrerer Netze (oder sogenannter UserModule, siehe Anhang 14). **MNA wendet sich an den erfahrenen User.** Bei der Installation von NN-Tool wird ein MNA-Beispiel mitgeliefert. Es handelt sich um die Dateien **MNA\_Demo\_PCA.pat** und **MNA\_Validation.pat** im Verzeichnis „**Eigene Dokumente \ NN-Tool**“. An diesem Beispiel wird im Folgenden die Vorgehensweise erläutert.

MNA basiert auf fertiggestellten NN-Tool-Netzen. Für die Beispielanwendung sind zunächst einmal die entsprechenden Netze zu erstellen. Bitte laden Sie zunächst die Datei **MNA\_Demo\_PCA.pat**.

**Hinweis:** Die Datei verwendet das Komma als Dezimaltrennzeichen. Falls Sie den Dezimalpunkt an Stelle des Dezimalkommas verwenden, müssen Sie in der Datei alle Kommata mit Hilfe eines Editors durch Punkte ersetzen.

Die Datei enthält 200 Datensätze von 10 Eingangsgrößen  $x_1, \dots, x_{10}$ , gefolgt von 9 Ausgangsgrößen  $y_1, \dots, y_9$ . Es zeigt sich, dass sowohl die Eingangsgrößen als auch die Ausgangsgrößen untereinander nicht unabhängig sind. Aus diesem Grund wurde mit PCATool jeweils eine Hauptkomponentenanalyse für die Eingänge (PCA\_X.1 bis PCA\_X.10) als auch für die Ausgangsgrößen (PCA\_Y.1 bis PCA\_Y.9) durchgeführt und die entsprechenden Werte der ursprünglichen .pat-Datei hinzugefügt. Darüber hinaus sind jeweils nur die ersten 3 Hauptkomponenten von Bedeutung. Wegen der starken Messfehlerbelastung der Datensätze bietet sich nun das folgende Vorgehen an:

1. Bereitstellung eines Netzes, das die 10 X-Werte auf die ersten 3 X-Hauptkomponenten abbildet. Trainieren Sie also ein Netz mit  $x_1, \dots, x_{10}$  als Input und PCA\_X.1, ..., PCA\_X.3 als Output, d.h. 10 Eingangsgrößen, 3 Ausgangsgrößen. Verwenden Sie dabei Crossvalidation 5-fach zyklisch in der Standardoption mit bis zu 8 inneren Knoten. Speichern Sie das Netz unter dem Anwendungsnamen X\_PCAX ab (**Menu "Datei", Menüpunkt "Netz speichern unter"**), d.h. X wird auf seine Hauptkomponenten abgebildet. Wie die folgenden Ergebnisse aus der automatischen Dokumentation zeigen, lassen sich die Hauptkomponenten sehr gut lernen. Das muss ja auch so sein, da es sich nur um durch Messfehler belastete lineare Beziehungen handelt.

Korrel. Testset Info							
Lfd. Nr.	Ausgangsparameter	Korrelation	Rel. Fehler in %	Abs. Fehler	a0-Koeffizient	a1-Koeffizient	Anzahl
1	PCA_X1	0,9999	0,2055	0,0208	-3,24E-03	1,0009	200
2	PCA_X2	0,9999	0,1503	0,0113	-2,86E-03	0,9997	200
3	PCA_X3	0,9998	0,3447	0,0163	-6,83E-04	0,9994	200

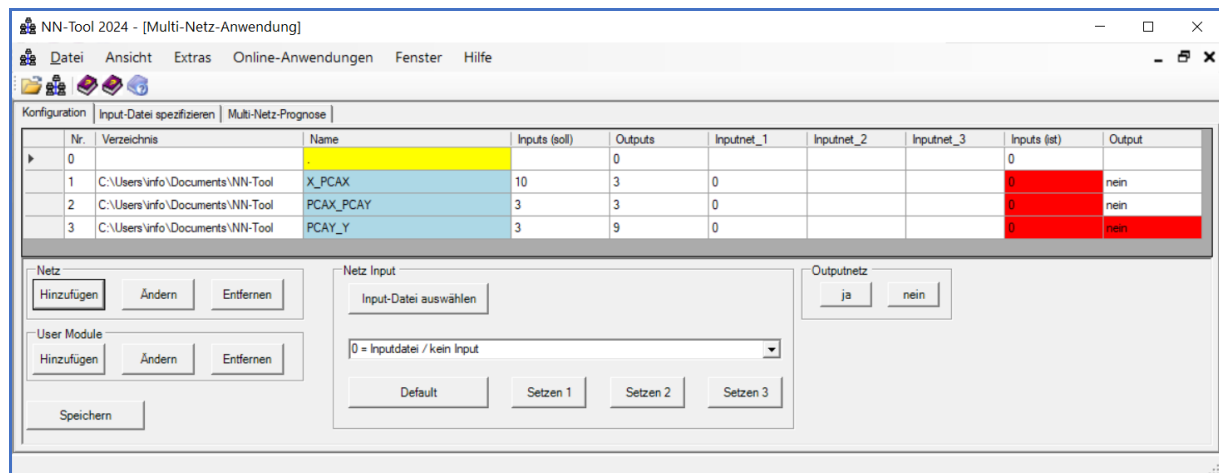
2. Trainieren Sie entsprechend ein Netz mit PCA\_X.1, ..., PCA\_X.3 als Input und PCA\_Y.1 bis PCA\_Y.3 als Output und speichern Sie es unter dem Anwendungsnamen PCAX\_PCAY.

Korrel. Testset Info							
Lfd. Nr.	Ausgangsparameter	Korrelation	Rel. Fehler in %	Abs. Fehler	a0-Koeffizient	a1-Koeffizient	Anzahl
1	PCA_Y1	0,9583	3,0417	0,4733	-0,0704	0,8874	200
2	PCA_Y2	0,9353	4,855	0,3994	0,0541	0,9471	200
3	PCA_Y3	0,9547	4,2282	0,2515	-4,03E-03	0,9592	200

3. Als letztes erstellen Sie ein Netz mit PCA\_Y.1 bis PCA\_Y.3 als Input und y1,...,y9 als Output und speichern es unter dem Namen PCA\_Y ab.

Korrel. Testset Info							
Lfd. Nr.	Ausgangsparameter	Korrelation	Rel. Fehler in %	Abs. Fehler	a0-Koeffizient	a1-Koeffizient	Anzahl
1	y1	0,9907	1,8034	0,0548	0,019	0,9804	200
2	y2	0,9963	1,1766	0,0515	-2,66E-03	0,9926	200
3	y3	0,9946	1,4143	0,0633	-7,30E-03	0,9813	200
4	y4	0,9878	1,2801	0,0861	0,0401	0,9511	200
5	y5	0,9915	1,2201	0,066	-0,0284	0,9647	200
6	y6	0,981	2,1743	0,0861	0,0802	0,9218	200
7	y7	0,9847	2,0068	0,1413	-0,1093	0,9324	200
8	y8	0,9875	1,8647	0,1104	0,0828	0,9438	200
9	y9	0,9964	0,9863	0,0657	5,81E-03	0,9902	200

Diese 3 Netze werden nun mittels MNA hintereinander geschaltet. MNA erlaubt nicht nur lineare Verschaltungen, sondern es sind beliebige Verknüpfungen von Netzen und UserModules erlaubt. Jedes Netz (bzw. UserModule) erhält seinen Input von den Outputs von bis zu drei vorgeschalteten Netzen (Usermodules). Dabei müssen Input und Output der entsprechenden Netze natürlich passen. Eine u.U. notwendige Anpassung der Outputs eines gegebenen Netzes kann durch Zwischenschaltung von passend definierten UserModules erfolgen. Das erste Netz wird entweder über eine Datei mit Eingangsdaten ("Prognose Inputdatei") versorgt oder beim Einsatz als Online-Anwendung werden die Daten von einem Leitsystem übertragen. Öffnen Sie **"Online-Anwendungen"** / **"Neue Multi-Netz-Anwendung"** und fügen Sie mittels der Taste "Hinzufügen" (bei „Netz“) die 3 erstellten Netze hinzu.

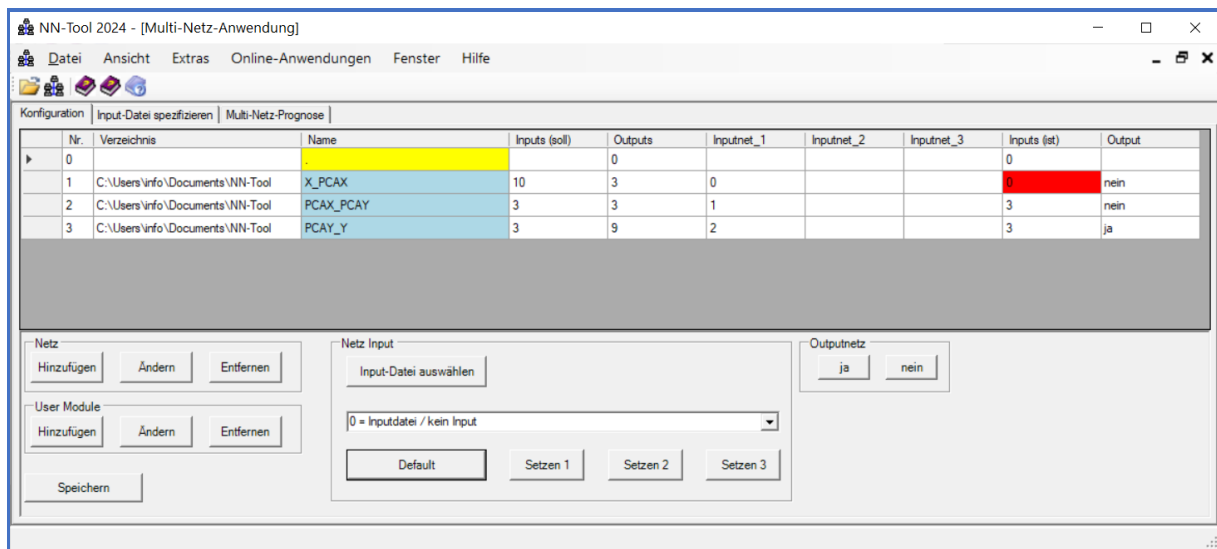


Die roten Felder zeigen an, dass die jeweiligen Eingangsströme („Inputs“) in die drei Netze noch nicht geeignet spezifiziert wurden. Darüber hinaus muss wenigstens ein Ausgangsnetz („Output“) festgelegt werden, üblicherweise das Letzte.

Für jedes Netz (bzw. UserModule) können hier bis zu 3 vorgeschaltete Netze spezifiziert werden (mit „Setzen 1“, „Setzen 2“ etc.) deren Outputs in das betreffende Netz als Input eingehen. Dabei werden die Inputs des gegebenen Netzes der Reihe nach belegt, d.h. zunächst

die Outputs des ersten vorgeschalteten Netzes „**Inputnet\_1**“ als erste Inputquelle, dann die Outputs des zweiten vorgeschalteten Netzes etc. Selbstverständlich muss die Summe der Inputs der vorgeschalteten Netze („**Inputs (ist)**“) mit der Anzahl der Inputs des gegebenen Netzes („**Inputs (soll)**“) übereinstimmen. Für das erste Netz muss stattdessen eine Input-Datei vorgegeben werden und anschließend die entsprechenden Parameter der Datei den Inputs des ersten Netzes zugeordnet werden.

Bei linearen Verschaltungen (wie hier), bei denen jedes Netz seine Inputs nur aus den Outputs des jeweils vorhergehenden Netzes bekommt, können Sie die Verschaltung nun mittels der Taste "**Default**" vornehmen.



Also Netz 2 „PCAX\_PCAY“ bekommt seinen Input von Netz 1 („X\_PCAX“) etc.

Spezifizieren Sie dann anschliessend noch die Datei **MNA\_Validation.pat** als Input-Datei durch Drücken der Taste „Input-Datei auswählen“.

**Hinweis:** Die Datei verwendet das Komma als Dezimaltrennzeichen. Falls Sie den Dezimalpunkt an Stelle des Dezimalkommas verwenden, müssen Sie in der Datei alle Kommata mit Hilfe eines Editors durch Punkte ersetzen.

Es folgt das folgende Fenster:

Configuration: Input-Datei spezifizieren | Multi-Netz-Prognose

Spaltennr.	Bezeichnung in Datei	Typ	Netzbezeichnung
1	Udnnr	K	Kennung
2	x1	K	Kennung
3	x2	K	Kennung
4	x3	K	Kennung
5	x4	K	Kennung
6	x5	K	Kennung
7	x6	K	Kennung
8	x7	K	Kennung
9	x8	K	Kennung
10	x9	K	Kennung
11	x10	K	Kennung
12	y1	K	Kennung
13	y2	K	Kennung
14	y3	K	Kennung
15	y4	K	Kennung
16	y5	K	Kennung
17	y6	K	Kennung

Inputs: soll  ist

Outputs: soll  ist

Spalten typ festlegen

Spaltenwert

Diese Vorgehensweise hier ist flexibler als bei der Option Validationset, da die ausgewählte Datei keinen fest vorgeschriebenen Aufbau haben muss. Dafür müssen Sie nun noch die entsprechenden Spalten der Datei als Input (X-Werte) und ggf. als Targetoutputs (Y-Werte) festlegen. Auf die Spezifikation von Targets kann auch verzichtet werden, wenn neue Datensätze einfach nur prognostiziert werden sollen. Targets können z.Z. nur für das Netz mit der höchsten Netznr. festgelegt werden. Auf Wunsch lassen sich jedoch auch die Ausgänge der anderen Netze mitprognostizieren und in eine Ausgabedatei oder nach Excel übertragen. Legen Sie die Zuordnung der Spalten der Inputdatei zu den Ein- und Ausgängen der Multinetz-anwendung wie folgt fest:

Spaltennr.	Bezeichnung in Datei	Typ	Netzbezeichnung
1	Lfdnr	K	Kennung
2	x1	I 1	x1
3	x2	I 2	x2
4	x3	I 3	x3
5	x4	I 4	x4
6	x5	I 5	x5
7	x6	I 6	x6
8	x7	I 7	x7
9	x8	I 8	x8
10	x9	I 9	x9
11	x10	I 10	x10
12	y1	O 1	y1
13	y2	O 2	y2
14	y3	O 3	y3
15	y4	O 4	y4
16	y5	O 5	y5
17	y6	O 6	y6
18	y7	O 7	y7
19	y8	O 8	y8
20	y9	O 9	y9

Inputs: soll  ist

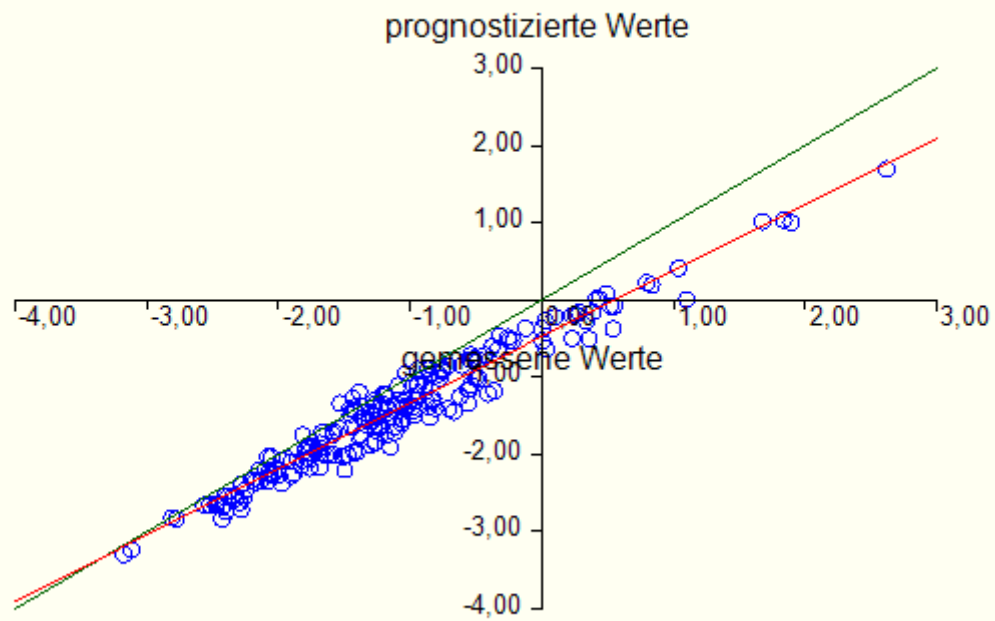
Outputs: soll  ist

Spaltentyp festlegen: Spaltenwert

Wechseln Sie auf die Registerkarte „Konfiguration“ und speichern Sie die komplette Multinetz-anwendung mit der Taste "Speichern" unter dem Namen "X\_PCAX\_PCAY\_Y.mna" ab.

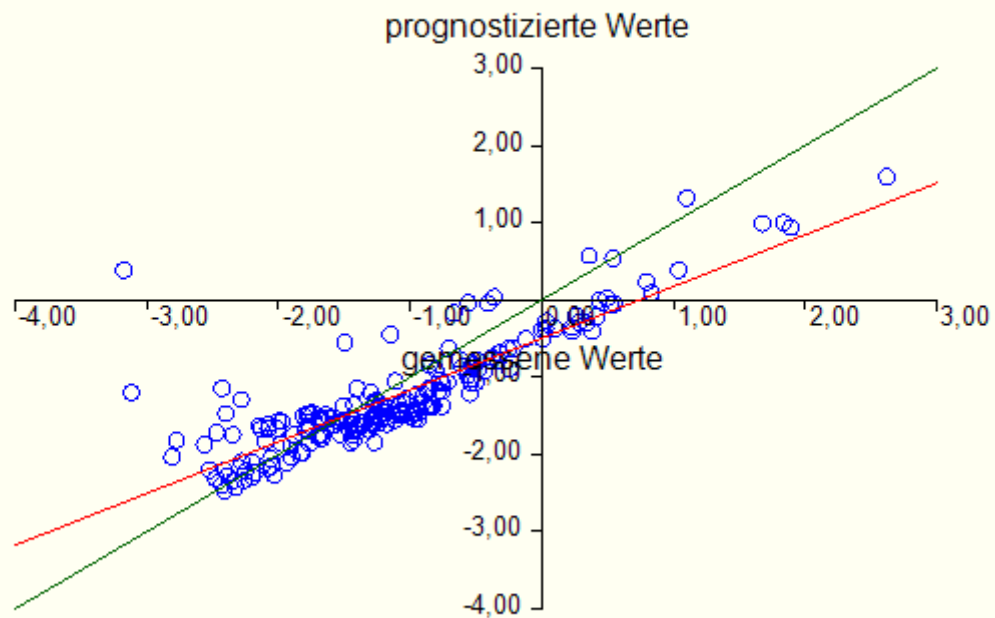
Auf dem Registerblatt „Multi-Netz-Prognose“ können Sie nun eine Prognose auf der Input-Datei durchführen. Dabei zeigt sich, dass die Ergebnisse z.T. deutlich besser sind als bei der direkten Prognose von  $y_1, \dots, y_9$  als Funktion von  $x_1, \dots, x_{10}$ . Für den Parameter  $y_7$  liefert die Multi-Netz-Anwendung das folgende Diagramm:

Validationset, y7,  $R = 0,9733$ ,  $Y = -0,4741 + 0,8561 * X$



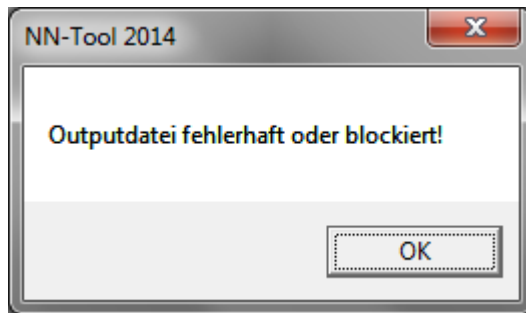
Zum Vergleich die direkte Prognose **auf der Datei MNA\_Validation.pat**:

Validationset, y7,  $R = 0,8604$ ,  $Y = -0,4958 + 0,6698 * X$



Eine fertiggestellte Multi-Netz-Anwendung lässt sich neben der Verwendung für Prognosezwecke insbesondere auch in Online-Anwendungen integrieren (vgl. Anhang 11: Online-Anwendungen).

**Hinweis:** Falls beim Start der Multi-Netz-Prognose die folgende Fehlermeldung



erscheint, benennen sie die Datei im Feld „Ausgabe in Datei“ einfach um.

## Anhang 17: UserModules

Die ausschließliche Verwendung von NN-Tool-Netzen in Multinetzapplikationen ist teilweise etwas unpraktisch. So muss bei einigen Anwendungen zusätzlich ein Netz erstellt werden, nur um bestimmte Werte durchzureichen. Darüber hinaus würde jedes Mal ein neues Netz benötigt, wenn an ein nachfolgendes Netz nur eine Teilmenge der Ausgangsparameter (oder diese in anderer Reihenfolge) weitergegeben werden sollte. UserModules lösen nun diese Probleme, in dem sie die benötigten Funktionalitäten bereitstellen. Dies wird im Folgenden am letzten Beispiel des Anhangs 16 erläutert. Bitte arbeiten Sie zunächst die entsprechenden Beispiele ab.

Im Beispiel des Anhangs 16 wurde eine Multinetz-anwendung erstellt, die ausgehend von X-Werten über die Hauptkomponenten der X-Werte und der Y-Werte schließlich die Y-Werte prognostiziert. Wir wissen jedoch, dass die X-Werte nicht unabhängig voneinander sind. Insbesondere können sie für Prognosezwecke in dafür erstellten Inputdateien nicht beliebig variiert werden. Dagegen sind die 3 X-Hauptkomponenten frei wählbar. Es macht also viel mehr Sinn, die Multi-Netz-Anwendung auf der Basis der 3 X-Hauptkomponenten aufzubauen und daraus die entsprechenden Y-Größen **und zusätzlich die passenden X-Werte** zu prognostizieren. Für den letzten Punkt wird also noch ein Netz PCAX-X benötigt, d.h. PCA\_X.1,...,PCA\_X.3 als Input und x1,...,x10 als Output. Erzeugen Sie zunächst dieses Netz.

Außerdem benötigt die gesamte Anwendung die Möglichkeit eine Komponente PCAX\_PCAX (d.h. PCA\_X.1,...,PCA\_X.3 sowohl als Input als auch als Output) als Inputquelle vorzuschalten. Der Grund ist, dass derzeit nur ein Netz bzw. UserModule auf die Eingangsdatei zugreifen kann. Dieses erste Netz/Modul kann dann aber alle benötigten Parameter bereitstellen, so dass die nachgeschalteten Netze/Module darauf zugreifen können.

Diese Funktionalität soll nun durch ein UserModule bereitgestellt werden. Öffnen Sie die Anwendung „Neues UserModule“ im Hauptmenu „Online-Anwendungen“:



The screenshot shows the 'NN-Tool 2024 - [Usermodules]' window. The menu bar includes 'Datei', 'Ansicht', 'Extras', 'Online-Anwendungen', 'Fenster', and 'Hilfe'. The toolbar contains icons for file operations and help. The 'Definieren' tab is active, with sub-tabs 'Parameter definieren' and 'Modell auswerten'. The main area has input fields for 'Inputs' and 'Outputs', a dropdown for 'Typ' (currently 'UMTyp 0: (affin) lineare Funktion'), and an 'Initialisieren' button. Below this is a large grey rectangular area. At the bottom, there is a 'Koeffizienten setzen' section with a text box containing '1' and a 'setzen' button, and a 'Speichern' button at the very bottom.

UserModules sind entweder (affin) lineare Funktionen von beliebig vielen Inputs auf beliebig viele Outputs (UMTyp 0) oder einer von z.Z. 19 vordefinierten Typen (z.B. UMTyp 10 entspricht „e hoch x“). Um ein UserModule zu definieren, wählen Sie einen der Typen aus, geben bei Typ 0 noch die gewünschte Anzahl der Inputs und Outputs vor und drücken „Initialisieren“. Die von uns gewünschte Funktionalität benötigt 3 Inputs und Outputs:

Definieren | Parameter definieren | Modell auswerten

Inputs: 3  
Outputs: 3  
Typ: UMTyp 0: (affin) lineare Funktion [v] Initialisieren

	1	X_1	X_2	X_3
Y_1	0	1	0	0
Y_2	0	0	1	0
Y_3	0	0	0	1

Koeffizienten setzen  
[1] setzen

Speichern

Dargestellt sind die Koeffizienten der (affin)linearen Abbildung von 3 Inputs (namens X\_1 bis X\_3) auf drei Outputs Y\_1 bis Y\_3. Die Spalte „1“ entspricht der Vorgabe von Konstanten. D.h. die erste Zeile ergibt:  $Y_1 = 0 \cdot 1 + 1 \cdot x_1 + 0 \cdot x_2 + 0 \cdot x_3$

Standardmässig sind alle Koeffizienten = 0 bis auf die Diagonalkoeffizienten (= 1). Mit der Taste „setzen“ können die Koeffizienten beliebig vorgegeben werden. Anschließend kann das Modell an Vorgabewerten auf der Seite „**Modell auswerten**“ getestet werden. Da wir die Werte nur durchreichen wollen, brauchen wir keine weiteren Einstellungen und könnten das Modell direkt speichern. Nachteilig ist, dass die Parameter nur Defaultnamen haben und zusätzlich keine Information über zulässige Minimal- und Maximalwerte vorliegt (sind daher standardmässig 0 bzw. 1). Um diesen Nachteil zu umgehen, kann ein UserModule (vom UMTyp 0) auch direkt aus der NN-Tool Datenanalyse erstellt werden. Dazu benötigen wir eine Datei in der die 3 X-Hauptkomponenten jeweils in 2 Spalten vorkommen. Öffnen Sie zunächst das letzte erstellte Netz „PCAX\_X“ und speichern Sie es zusätzlich unter dem Namen PCAX\_PCAX ab. Öffnen Sie nun **PCAX\_PCAX.pat**. Im geöffneten Datenblatt müssen wir jetzt die 3 Spalten für PCA\_X1 bis PCA\_X3 duplizieren. Klicken sie dafür nacheinander in die Spalten und drücken sie jeweils die Taste „Spalte duplizieren“. Die entsprechenden Spalten werden nun angefügt. Drücken Sie anschließend Änderungen speichern und speichern Sie die Datei unter dem Namen „PCAX\_PCAX\_New“ ab. Führen Sie dann die Datenanalyse aus, setzen Sie alle Parameter auf passiv bis auf die Spalten für PCA\_X1 bis PCA\_X3, die jeweils doppelt vorhanden sind. Setzen sie das erste Auftreten als Eingangs-, das zweite Auftreten als Ausgangsparameter:

NN-Tool 2024 - C:\Users\info\Documents\NN-Tool\PCAX\_PCAX\_New - [Datenanalyse]

Datei Datenanalyse Netztraining Ansicht Extras Online-Anwendungen Fenster Hilfe

Datenanalyse Lernparameter Input-Output-Zuordnung Lernset- / Testseteinteilung

	Parameter	Min	Max	Mittel	Stdabw	Median	vollständig	Transform.	E/A	A/P	verfügbar
18	y7	-3,9572	3,0860	-1,3486	1,1372	-1,4838	200	Lin	E	P	
19	y8	-2,4516	3,4683	1,2891	0,9552	1,3976	200	Lin	E	P	
20	y9	-4,0735	2,5880	0,5087	1,0209	0,5461	200	Lin	E	P	
21	PCA_X1	-5,2630	4,8801	2,6263e-09	2,3441	0,1075	200	Lin	E	A	
22	PCA_X2	-3,7689	3,7378	-1,7695e-10	1,6028	-0,0563	200	Lin	E	A	
23	PCA_X3	-2,4662	2,2677	-1,7765e-09	1,1972	0,0430	200	Lin	E	A	
24	PCA_X4	-1,2933	0,9806	-5,0291e-10	0,4947	0,0437	200	Lin	E	P	
25	PCA_X5	-0,7117	0,7316	2,9686e-11	0,3133	-3,5085e-03	200	Lin	E	P	
26	PCA_X6	-0,6275	0,5593	-7,2469e-10	0,2597	3,9160e-03	200	Lin	E	P	
27	PCA_X7	-0,4291	0,6998	-5,5297e-10	0,2108	-0,0227	200	Lin	E	P	
28	PCA_X8	-0,5256	0,4730	5,5821e-10	0,1971	-0,0187	200	Lin	E	P	
29	PCA_X9	-0,5178	0,4813	-5,1776e-10	0,1883	-6,2116e-04	200	Lin	E	P	
30	PCA_X10	-0,4452	0,4729	2,5175e-10	0,1562	0,0131	200	Lin	E	P	
31	PCA_Y1	-6,2290	9,3325	7,6042e-09	2,4000	-0,2466	200	Lin	E	P	
32	PCA_Y2	-4,5854	3,6414	-1,2433e-09	1,4279	0,0605	200	Lin	E	P	
33	PCA_Y3	-3,0132	2,9351	8,7341e-10	1,0634	-0,0537	200	Lin	E	P	
34	PCA_Y4	-0,7120	0,5686	6,3941e-10	0,1745	4,2906e-03	200	Lin	E	P	
35	PCA_Y5	-0,4768	0,4187	5,0059e-11	0,1597	7,6606e-03	200	Lin	E	P	
36	PCA_Y6	-0,5392	0,3886	2,3048e-10	0,1461	0,0120	200	Lin	E	P	
37	PCA_Y7	-0,6511	0,5060	-2,7241e-10	0,1219	1,4843e-03	200	Lin	E	P	
38	PCA_Y8	-0,4615	0,2909	2,7423e-10	0,1128	7,6095e-04	200	Lin	E	P	
39	PCA_Y9	-0,3701	0,2913	-1,5243e-10	0,1031	1,3661e-03	200	Lin	E	P	
40	PCA_X1	-5,2630	4,8801	2,6263e-09	2,3441	0,1075	200	Lin	A	A	200
41	PCA_X2	-3,7689	3,7378	-1,7695e-10	1,6028	-0,0563	200	Lin	A	A	200
42	PCA_X3	-2,4662	2,2677	-1,7765e-09	1,1972	0,0430	200	Lin	A	A	200

Eingang Aktiv Transformationen nach Anzahl abschalten User-Modul definieren

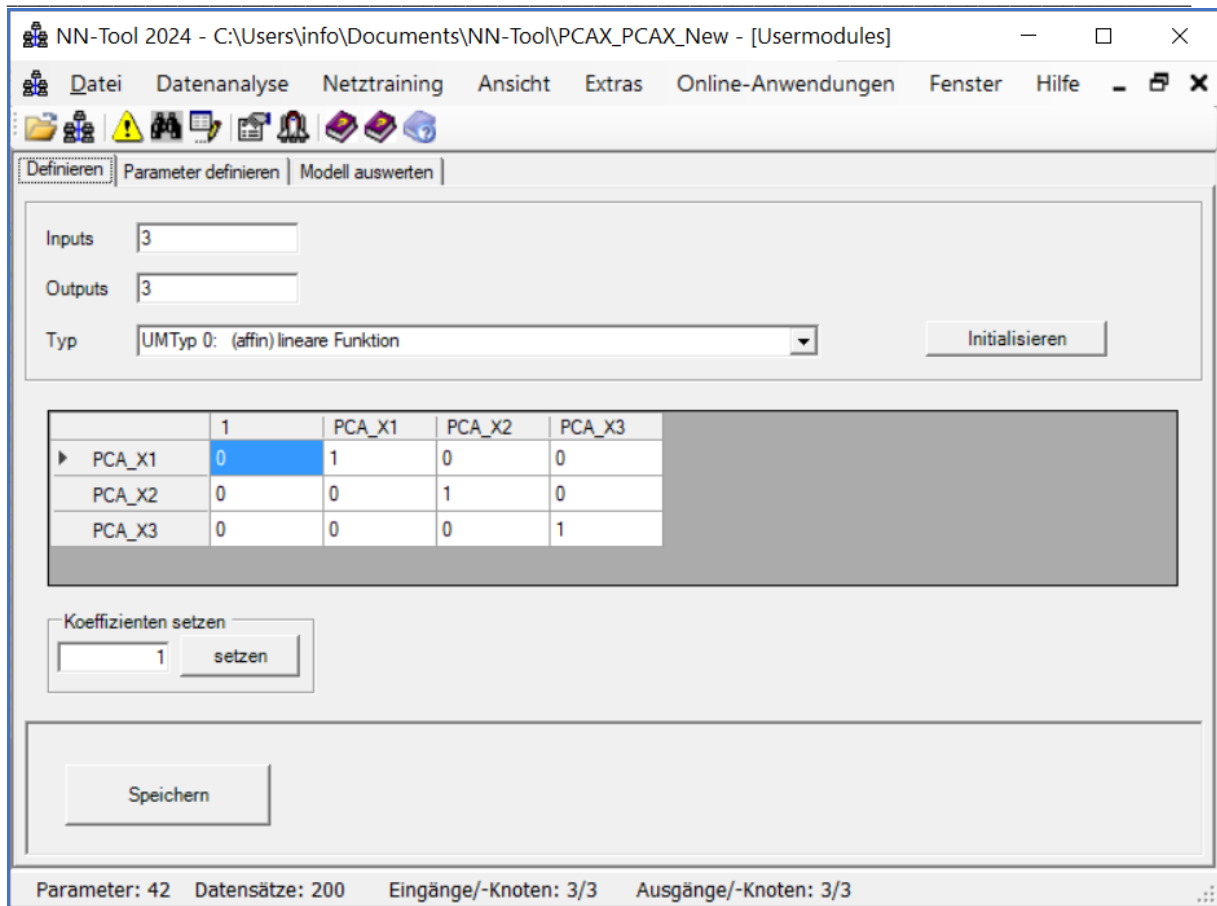
Ausgang Passiv Lin Auto-Eingänge Klassifikator manuell

Log Auto-Ausgänge Histogramm

Sig Alle Lin Datenverlauf

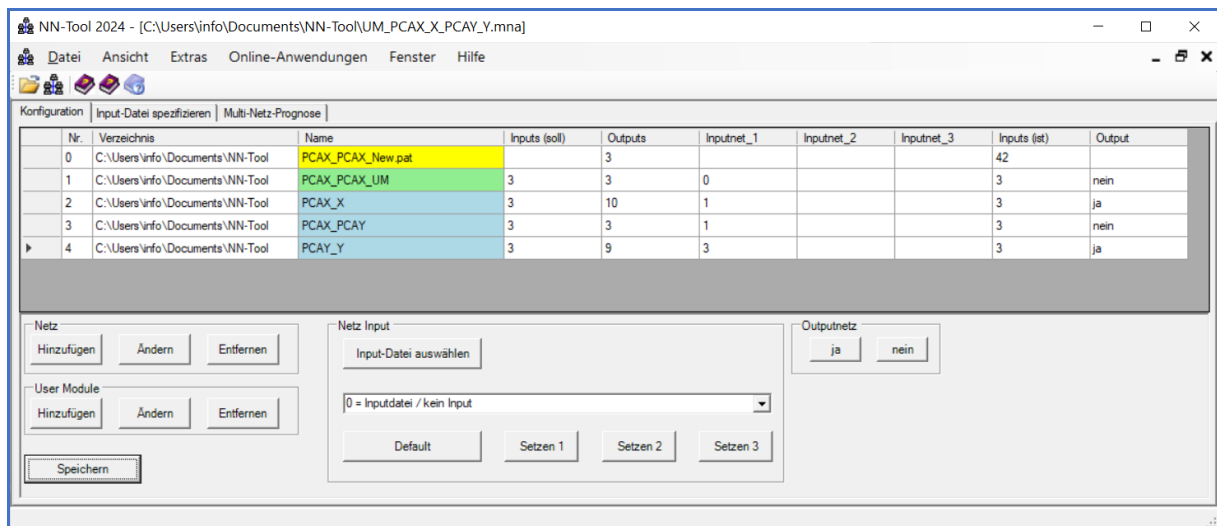
Parameter: 42 Datensätze: 200 Eingänge/-Knoten: 3/3 Ausgänge/-Knoten: 3/3

und drücken Sie nun die Taste „User-Module definieren“



Speichern Sie das Module unter der Bezeichnung „PCAX\_PCAX\_UM“ ab.

Definieren Sie mit diesem Modul jetzt die folgende Multinetz-Anwendung:

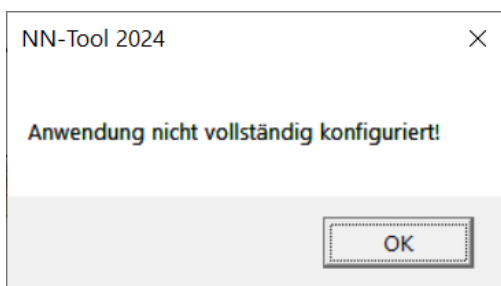


Speichern Sie die gesamte Applikation unter dem Namen „UM\_PCAX\_X\_PCAY\_Y“ ab.

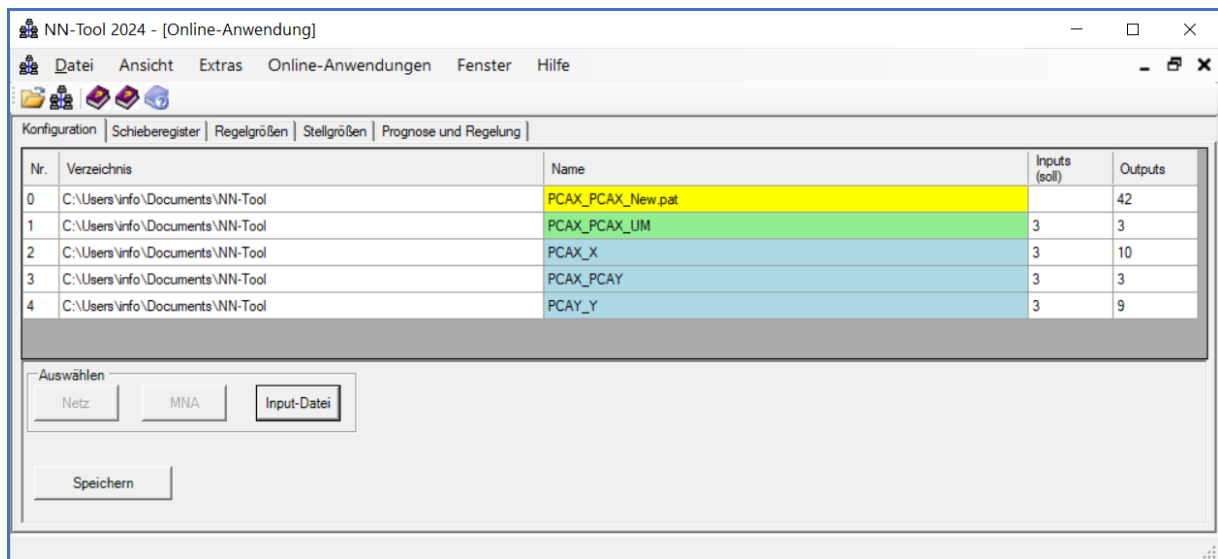
Die gesamte Verschaltung kann nun aus entsprechenden Eingabedateien auf der Basis der unabhängigen Größen PCA\_X1 bis PCA\_X3 sowohl die zugehörigen X-Werte als auch die Y-Werte prognostizieren und z.B. nach Excel oder in eine Ausgabedatei ausgeben. Beachten Sie, dass zu diesem Zweck auch das Netz „PCAX\_X“ als „Output“ gesetzt worden ist.

Darüber hinaus ermöglichen die übrigen UserModule-Typen durch entsprechende Verschaltungen beliebige Transformationen an den Outputs vorgeschalteter Netze/UserModule vorzunehmen und diese transformierten Werte an die nachgeschalteten Netze/UserModule weiterzugeben. In diesem Sinne ist auch die derzeitige Beschränkung auf maximal 3 vorgeschaltete Netze bzw. genau ein Netz/Usermodule, das auf die Eingabedatei zugreift, nicht wirklich eine Einschränkung. Insgesamt lassen sich beliebige Netzwerke von miteinander verschalteten Netzen/UserModulen definieren und auswerten.

Im Folgenden werden wir nun eine Simulation auf der Basis dieser Multinetzanwendung erzeugen. Starten Sie NN-Tool neu, wählen Sie **„Neue Online-Anwendung“**, Registerkarte **„Konfiguration“**, drücken Sie den Button **„MNA“** und wählen Sie die unter UserModule beschriebene Multinetzanwendung **„UM\_PCAX\_X\_PCAY\_Y.mna“**. Zunächst kommt der folgende Hinweis:



der anzeigt, dass bei der Dateizuordnung noch was getan werden muss, dann ergibt sich:



Wechseln Sie nun auf die Registerkarte **„Schieberegister“**.

Parameter zuordnen

PCA\_X1 Shift 0 Lfdnr 12

Zuordnen

Lfd. Nr.	Parameternamen	Min	Max	Shift	Spaltennr. in Datei	Bezeichnung in Datei
1	PCA_X1	-5,26	4,88	0	12	y1
2	PCA_X2	-3,77	3,74	0	13	y2
3	PCA_X3	-2,47	2,27	0	14	y3
1	y1	-1,00	2,04	0	12	y1
2	y2	-2,82	1,56	0	13	y2
3	y3	-2,39	2,09	0	14	y3
4	y4	-2,49	4,24	0	15	y4
5	y5	-3,58	1,83	0	16	y5
6	y6	-1,75	2,21	0	17	y6
7	y7	-3,96	3,09	0	18	y7
8	y8	-2,45	3,47	0	19	y8
9	y9	-4,07	2,59	0	20	y9

Wie Sie sehen, ist die Zuordnung für die Outputs y1,..., y9 (grün) korrekt, den Inputs PCA\_X1 bis PCA\_X3 werden jedoch die falschen Spalten der Datei zugeordnet. Die Zuordnung muss nun für jeden Input einzeln mittels der beiden Listenfelder und der Taste „Zuordnen“ vorgenommen werden:

Parameter zuordnen

PCA\_X1 Shift 0 Lfdnr 21

Zuordnen

Lfd. Nr.	Parameternamen	Min	Max	Shift	Spaltennr. in Datei	Bezeichnung in Datei
1	PCA_X1	-5,26	4,88	0	21	PCA_X1
2	PCA_X2	-3,77	3,74	0	22	PCA_X2
3	PCA_X3	-2,47	2,27	0	23	PCA_X3
1	y1	-1,00	2,04	0	12	y1
2	y2	-2,82	1,56	0	13	y2
3	y3	-2,39	2,09	0	14	y3
4	y4	-2,49	4,24	0	15	y4
5	y5	-3,58	1,83	0	16	y5
6	y6	-1,75	2,21	0	17	y6
7	y7	-3,96	3,09	0	18	y7
8	y8	-2,45	3,47	0	19	y8
9	y9	-4,07	2,59	0	20	y9

Speichern Sie die Konfiguration unter dem Namen „UM\_PCAX\_X\_PCAY\_Y\_Online.mna“ ab, wechseln Sie auf „Prognose und Regelung“ und drücken Sie „Start“ (ggf. erfolgt noch eine Meldung über die Registerlänge, die automatisch angepasst wurde):

NN-Tool 2024 - [C:\Users\info\Documents\NN-Tool\UM\_PCAX\_X\_PCAY\_Y\_Online.mna]

Datei Ansicht Extras Online-Anwendungen Fenster Hilfe

Konfiguration | Schieberegister | Regelgrößen | Stellgrößen | Prognose und Regelung

Lfd. Nr.	Input	Messwerte	Stellwerte
1	PCA_X1	2.670	2.670
2	PCA_X2	-0.418	-0.418
3	PCA_X3	-0.333	-0.333

Lfd. Nr.	Output	Prognose	Prog.korr	geregelt	Sollwert
1	y1	0.876	0.876	0.876	
2	y2	-0.158	-0.158	-0.158	
3	y3	-0.205	-0.205	-0.205	
4	y4	0.349	0.349	0.349	
5	y5	-0.385	-0.385	-0.385	
6	y6	0.848	0.848	0.848	
7	y7	-1.200	-1.200	-1.200	
8	y8	1.133	1.133	1.133	
9	y9	0.632	0.632	0.632	

Filter zur Prognoseverbesserung

Input-Filter   
0 = keine Glättung, 1 = unendliche Glättung

Offset-Kompensation   
0 = keine, 1 = maximale Kompensation

Lebenszeichen

Regler

Restabweichung in %   
Iterationsschritte   
Reduktionsfaktor   
Max. Iterationen

Simulation

Zeittakt in ms

**Hinweis:** Zu dieser NN-Tool-Komponente ist eine .NET-Runtimebibliothek verfügbar, um die Anwendung mit externen Applikationen (insbesondere Prozessleitsystemen) zu koppeln (siehe Produktübersicht).

## Anhang 18: NN-Tool Netzstruktur

Prognose eines Eingangsvektors  $X_{\text{String}}$  der Länge  $L1$  von Strings besteht aus 5 Transformationsschritten:

1. **Umcodieren der Klassifikatoren:** Numerische Parameter werden durchgereicht, Klassifikatoren werden in entsprechende 0-1 Werte umcodiert, d.h. Eingangsparameter für aktive Klasse erhält den numerischen Wert 1, übrige Parameter des Klassifikators den Wert 0. Codierung nach Reihenfolge des Auftretens der Klasseninstanzen in den Daten. Erzeugt numerischen Eingangsvektor  $X_{\text{num}}$  der Länge  $L2$  ( $L2 \geq L1$ ).
2. **Skalierung auf das Intervall  $[0.1, 0.9]$**  erzeugt skalierten Eingangsvektor  $X$  der Länge  $L3$  ( $L3 = L2 + 1$ ).  $\min, \max, \text{mittelw}, \text{stdabw}$  sind die entsprechenden Werte der einzelnen Parameter von  $X_{\text{num}}$ .  $\text{Log}(\ )$  ist der natürliche Logarithmus

a. Linear: 
$$x_s = 0.8 \frac{x_{\text{num}} - \min}{\max - \min} + 0.1 = P_0 + P_1 x_{\text{num}}$$

b. Log.: 
$$x_s = 0.8 \frac{\text{Log}(x_{\text{num}}) - \text{Log min}}{\text{Log max} - \text{Log min}} + 0.1 = P_0 + P_1 \text{Log}(x_{\text{num}})$$

mit  $\text{Log max} = \text{Log}(\max)$

c. Sigmoid: 
$$x_s = \frac{0.8}{1 + e^{-a(x_{\text{num}} - \text{mittelw})}} + 0.1 = \frac{0.8}{1 + e^{(P_0 + P_1 x_{\text{num}})}} + 0.1$$

mit 
$$a = \frac{1.6}{\text{stdabw}}$$

- d. Zusätzlich wird als letzte Vektorkomponente der konstante Wert 1 (=Bias) angehängt

3. **Neuronale Netzabbildung** erzeugt skalierten Ausgangsvektor  $Y$  der Länge  $L4$ :

$$Y_r = \sigma \left( \sum_{k=1}^n \text{net}_k w_{k,r} \right) \text{ für } r = 1, \dots, L4$$

mit  $\text{net}_k = \sigma \left( \sum_{i=1}^{L3} x_i v_{i,k} \right)$  und  $\sigma(x) = \frac{1}{1 + e^{-x}}$



---

4. **Rückskalierung der Outputs** erzeugt Vektor  $Y_{\text{num}}$  der Länge L4:

a. Linear: 
$$y_{\text{num}} = \frac{(y_s - 0.1)(\text{max} - \text{min})}{0.8} + \text{min} = P_0 + P_1 y_s$$

b. Log.: 
$$y_{\text{num}} = \text{Exp}\left(\frac{(y_s - 0.1)(\text{Log max} - \text{Log min})}{0.8} + \text{Log min}\right) = e^{P_0 + P_1 y_s}$$

c. Sigmoid: 
$$y_{\text{num}} = -\frac{1}{a} \text{Log}\left(\frac{0.8}{y_s - 0.1} - 1\right) + \text{mittelw} = P_0 + P_1 \text{Log}\left(\frac{0.8}{y_s - 0.1} - 1\right)$$

mit  $a = \frac{1.6}{\text{stdabw}}$  und Beschränkung von  $y_s$  auf Intervall  $[0.10653, 0.89347]$

5. **Rückcodierung der Ausgangsklassifikatoren:** Numerische Parameter werden durchgereicht, für Klassifikatoren wird der Klassenname der Klasse mit der höchsten Bewertung bei den entsprechenden Ausgangsparametern von  $Y_{\text{num}}$  ausgewählt. Erzeugt Stringvektor  $Y_{\text{String}}$  der Länge L5 ( $L5 \leq L4$ ).

## Anhang 19: NN-Tool Produktübersicht

### A. NN-Tool Vollversion

Vollständiges Entwicklungs- und Anwendungssystem zur Prognose, Simulation und Optimierung mit neuronalen Netzen. Enthält u.a. die folgenden Hauptfunktionalitäten:

- **Analysemodul:** Datenvorverarbeitung und statistische Analyse, incl. entsprechender Module für Klassifikatoren, zur Korrelationsrechnung für die Bestimmung optimaler Eingangsparameter, für Zeitreihen mit Totzeiten sowie zur Behandlung unvollständiger Datensätze.
- **Lernmodul:** vollautomatisches Netztraining incl. Netzstrukturoptimierung und Crossvalidation.
- **Netzauswertung:** Güteabschätzung der Netze mit Scatter- und Verlaufsplots, Einflußanalysen, Fehlersuchmodul zur Bestimmung inkonsistenter Datensätze, Darstellung funktionaler Zusammenhänge zwischen Ein- und Ausgangswerten.
- **Anwendungsmodul Mischpult:** Simulation neuer Anwendungsfälle (z.B. Rezepturen, Betriebspunkte) auf der Basis des erstellten Netzes.
- **Anwendungsmodul Optimierer:** dient der Optimierung von Anwendungsfällen (Rezepturen, Betriebspunkten, etc.) unter Berücksichtigung einer Vielzahl von Nebenbedingungen auf der Basis des erstellten Netzes.
- **Anwendungsmodul Versuchsplanung:** Erstellung von zielgerichteten, d.h. auf den Anwendungsfall optimierten, Versuchsplänen auf der Basis eines erstellten neuronalen Netzes.
- **Dokumentationsmodul:** vollautomatische Protokollierung des erstellten Netzes incl. sämtlicher relevanter Diagramme.
- **Batch-Steuerung:** Steuerung von NN-Tool über Batchdatei. Dient der "Fernsteuerung" des Lernvorgangs bei Integration von NN-Tool in firmenspezifische Anwendungsumgebungen (insbesondere Anwenderdatenbanken).
- **Multinetzmodul:** ermöglicht die Kombination mehrerer Netze zu einer Gesamtanwendung.
- **Anwendungsmodul Dynamische Simulation:** dient der Simulation von Zeitreihen, beispielsweise zur Analyse des Anfahrverhaltens von verfahrenstechnischen Anlagen
- **Konfigurations- und Simulationsumgebung für Online-Anwendungen:** bereitet NN-Tool-Netze für den Online-Einsatz auf.
- **Online-Helpsystem**

### B. NN-Tool Batchversion

Funktionalität wie in der Vollversion mit der Ausnahme, dass das Netztraining nur im Batchmodus (d.h. über ASCII-Steuerdatei) möglich ist. Zur Integration von NN-Tool in firmenspezifische Anwendungen.

### **C. NN-Tool RunTimeversion**

Wie Vollversion ohne die Möglichkeit des Netztrainings (enthält also insbesondere die Anwendungsmodule). Dient der Auswertung erstellter neuronaler Netze zu Prognose- und Optimierungszwecken. Netze selbst lassen sich mit dieser Komponente weder erstellen noch verändern.

### **D. NN-Tool Excel-Add-In**

Ermöglicht die Verwendung bereits erstellter neuronaler Netze innerhalb von Excel-Arbeitsmappen.

### **E. NN-Tool C-Library**

Enthält Funktionen zum Laden und Auswerten erstellter neuronaler Netze als C-Funktionen. Ermöglicht die Einbindung der Netze in beliebige Applikationen, auch unter UNIX, LINUX etc.

### **F. NN-Tool Class-Library für Visual Basic / .NET**

Enthält Funktionen zum Laden und Auswerten erstellter neuronaler Netze als VB-Funktionen. Ermöglicht die Einbindung der Netze in beliebige Visual Basic oder .NET-Applikationen.

### **G. NN-Tool ControlServer**

Komponente zur Realisierung von Online-Anwendungen, z.B. Softsensor-Anwendungen oder modellgestützten Regelungen, auf der Basis von NN-Tool Prozessmodellen (ActiveX-Library/ .NET-Library).

### **H. Wartungsverträge**

Zu allen Komponenten sind Wartungsverträge möglich. Diese umfassen die Behebung auftretender Softwarefehler sowie die Lieferung der jeweils aktuellen Versionen.

## Anhang 20: Aufbau der 'COB'-Datei:

Wie bereits im Anhang 3 beschrieben, lässt sich NN-Tool mittels der COB-Datei fernsteuern. Die entsprechende Datei kann aus NN-Tool automatisch generiert werden. Für user-spezifische Anwendungen (z.B. Datenbanken) kann die Datei jedoch auch aus anderen Anwendungen heraus erzeugt werden. Der Aufbau ist im Folgenden beschrieben:

Hierbei bedeutet Nr die entsprechende Zeile innerhalb der COB-Datei

Nr	Ausdruck	Nr	Ausdruck
1	!path	40	}
2	c:\fb\nn_98_2_9\projecttest	41	}
3	!application	42	#!autotrans_inputs
4	class6	43	#!autotrans_outputs
5	!datenfile_oeffnen	44	!Transformationstype
6	2	45	{
7	!kennungen	46	3
8	{	47	LOG
9	13	48	4
10	14	49	LOG
11	16	50	7
12	}	51	SIG
13	!ausgaenge	52	8
14	{	53	SIG
15	1	54	}
16	2	55	!Analyse ende
17	3	56	!Lernen automatisch
18	15	57	{
19	17	58	mininnere
20	}	59	1
21	!passive	60	maxinnere
22	{	61	8
23	13	62	minlern
24	14	63	10
25	16	64	maxlern
26	}	65	100
27	!IO-Zuordnung	66	steplern
28	{	67	10
29	1	68	testsetanteil
30	{	69	5
31	4	70	schrittzus
32	5	71	0
33	6	72	errornodew
34	}	73	0
35	15	74	}
36	{	75	#!auto_doku
37	5	76	#!netz_laden

<b>38</b>	6	<b>77</b>	#!Programm_ende
<b>39</b>	7	<b>78</b>	

**Vorgaben:**

Jeder Parameter wird durch die absolute Parameternummer identifiziert, d.h. die Spaltennummer in der .pat-Datei.

In jeder Zeile steht nur ein Ausdruck. Befehle werden mit dem !-Zeichen eingeleitet. Zwischen Groß- und Kleinschreibung wird nicht unterschieden. Tabstops und Blanks werden sowohl vor als auch hinter jeden Ausdruck gelöscht. Einzelne Zeilen können mit dem #-Zeichen ausdokumentiert werden. Leerzeilen sind nicht zulässig.

Erklärung der einzelnen Zeilen:

Zeile 1: !Path

Zeile 2: Path der Application

Zeile 3: !application

Zeile 4: Name der Application

Zeile 5: !datenfile\_oeffnen

Die Application wird mit dem angegebenen Path geöffnet

Zeile 6: 2

Parameternamen stehen in der Zeile 2

Zeile 7.: !kennungen

Zeile 8: {

Zeile 9-11: Nummern der Parameter

Zeile 12: }

Falls kein Parameter innerhalb des Programms auf Kennungen gesetzt wurde sind die Zeilen 7 – 12 nicht vorhanden.

Zeile 13: !ausgänge

Zeile 14: {

Zeile 15-19: Ausgangsparameter

Zeile 20: }

Zu Beginn sind alle Parameter als Eingangsparameter gesetzt.

Die aufgeführten Parameter werden als Ausgangsparameter gesetzt.

Zeile 21: !passive

Zeile 22: {

Zeile 23-25: Passive Parameter

Zeile 26: }

Zu Beginn sind alle Parameter als Aktiveparameter gesetzt.

Die aufgeführten Parameter werden als Passiveparameter gesetzt.

Zeile 27: !IO-Zuordnung

Zeile 28: {

Zeile 29: 1. Ausgangsparameter der von folgenden Eingängen nicht abhängt

---

```

Zeile 30:      {
Zeile 31-33:      Eingangspartner
Zeile 34:      }
Zeile 35:      2. Ausgangspartner
Zeile 36:      {
Zeile 37-39:      Eingangspartner
Zeile 40:      }
Zeile 41:      }

```

Zu Beginn sind alle Ausgänge von allen Eingängen abhängig.  
 Falls keine Zuordnung durchgeführt wurde sind Zeilen 27-41 nicht vorhanden.

```

Zeile 42:      #!autotrans_inputs
               z.Z. noch keine Funktion
Zeile 43:      #!autotrans_outputs
               z.Z. noch keine Funktion

```

```

Zeile 44:      !Transformationstypen
Zeile 45:      {
Zeile 46:      1. Parameter
Zeile 47:      Transformationstypen des 1. Parameters (LOG oder SIG)
Zeile 48:      n. Parameter
Zeile 49:      Transformationstypen des n. Parameters
Zeile 54:      }

```

Zu Beginn sind alle Parameter vom Typ LIN. Falls keine Transformation ausgewählt wurde sind die Zeilen 44-54 nicht vorhanden.

```

Zeile 55:      !Analyse ende
               Analyse wird beendet

```

```

Zeile 56:      !Lernen automatisch
Zeile 57:      {
Zeile 58:      mininnere
Zeile 59:      1
Zeile 60:      maxinnere

```

mininnere	→ Minimalzahl innerer Knoten pro Ausgangspartner
maxinnere	→ Maximalzahl innerer Knoten pro Ausgangspartner
minlern	→ Minimalanzahl Lernschritte
maxlern	→ Maximalanzahl Lernschritte
steplern	→ Lernschrittweite
testsetanteil	→ Anteil Testsätze
schriftzus	→ Aufteilung der Datensätze
0	jeder n-te Datensatz im Testset
1	zusammenhängender Testset

```

Zeile 74:      }

```

```

Zeile 75:      #!auto_doku
               Diese Zeile ist ausdokumentiert, wenn die Datei 'class6.dok' nicht vorhanden ist.

```

Zeile 76:     #!netz\_laden  
Ein vorhandenes Netz wird geladen

Zeile 77:     #!Programm\_ende  
Beendet den Batchmode und NNTool ohne Rückfrage

## Anhang 21: GNU LESSER GENERAL PUBLIC LICENSE

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.  
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA  
Everyone is permitted to copy and distribute verbatim copies  
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts  
as the successor of the GNU Library Public License, version 2, hence  
the version number 2.1.]

### Preamble

The licenses for most software are designed to take away your  
freedom to share and change it. By contrast, the GNU General Public  
Licenses are intended to guarantee your freedom to share and change  
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some  
specially designated software packages--typically libraries--of the  
Free Software Foundation and other authors who decide to use it. You  
can use it too, but we suggest you first think carefully about whether  
this license or the ordinary General Public License is the better  
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,  
not price. Our General Public Licenses are designed to make sure that  
you have the freedom to distribute copies of free software (and charge  
for this service if you wish); that you receive source code or can get  
it if you want it; that you can change the software and use pieces of  
it in new free programs; and that you are informed that you can do  
these things.

To protect your rights, we need to make restrictions that forbid  
distributors to deny you these rights or to ask you to surrender these  
rights. These restrictions translate to certain responsibilities for  
you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis  
or for a fee, you must give the recipients all the rights that we gave  
you. You must make sure that they, too, receive or can get the source  
code. If you link other code with the library, you must provide  
complete object files to the recipients, so that they can relink them  
with the library after making changes to the library and recompiling  
it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the  
library, and (2) we offer you this license, which gives you legal  
permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that  
there is no warranty for the free library. Also, if the library is  
modified by someone else and passed on, the recipients should know  
that what they have is not the original version, so that the original  
author's reputation will not be affected by problems that might be



introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

---

GNU LESSER GENERAL PUBLIC LICENSE  
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses

the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

- a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

- b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system,

rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and

all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

#### NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

#### END OF TERMS AND CONDITIONS

#### How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>
```

```
This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.
```

```
This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.
```

```
You should have received a copy of the GNU Lesser General Public
```

---

License along with this library; if not, write to the Free Software  
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-  
1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your  
school, if any, to sign a "copyright disclaimer" for the library, if  
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the  
library `Frob' (a library for tweaking knobs) written by James Random  
Hacker.

<signature of Ty Coon>, 1 April 1990  
Ty Coon, President of Vice

That's all there is to it!

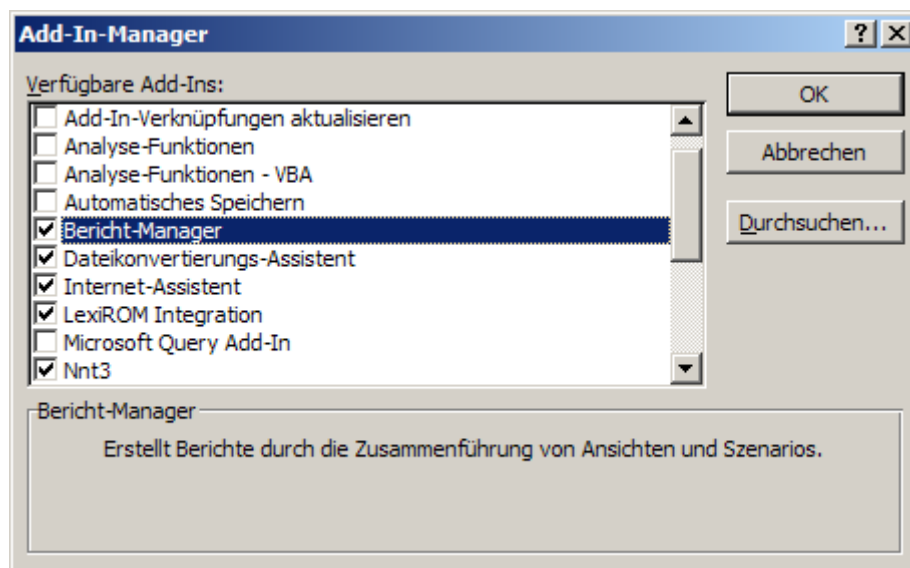


## Anhang 22: Installations- und Laufzeitfehler

### **Menueintrag NN-Tool ist in Excel nicht vorhanden:**

Wenn Excel aus NN-Tool heraus zum ersten Mal gestartet wird, versucht NN-Tool in Excel ein sogenanntes Add-In namens „Nnt3.xla“ zu installieren (nicht zu verwechseln mit dem Add-In unter Anhang 19). Ein Add-In ist eine Zusatzkomponente (ein „Makro“) in Excel mit der Excel um Funktionalitäten erweitert werden kann. Das Add-In nnt3.xla (Name nnt3, Dateiendung .xla für Excel-Add-In) dient dazu, die Daten in einer Excel-Tabelle auf komfortable Weise in eine NN-Tool Datendatei (eine \*.pat-Datei) zu konvertieren. In Abhängigkeit der eingestellten Schutzmechanismen und der verwendeten Version läßt Excel diese automatische Installation nicht immer zu. Gehen Sie dann wie folgt vor:

- Falls Sie Excel 2007 (Office 2007) verwenden: In diesem Fall findet sich das Menu auf dem Registerblatt „Add-Ins“. Sehen Sie zunächst dort nach.
- Versuchen Sie das Add-In manuell zu installieren. Benutzen Sie dazu den Befehl „Add-In-Manager“ im Menu „Extras“. (Office 2007: Bitte sehen Sie in der Hilfe nach wie Sie Add-Ins installieren können). Es öffnet sich der Add-In-Manager.

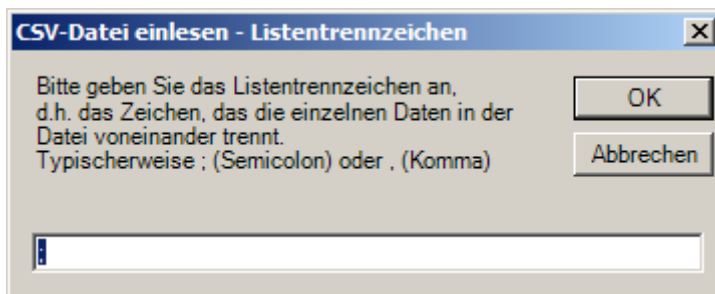


- Mit Durchsuchen können sie das Add-In „Nnt3“ im NN-Tool-Installationsverzeichnis finden und installieren. Falls Sie nicht mehr wissen, in welches Verzeichnis Sie NN-Tool installiert haben, können Sie den Verzeichnisnamen in NN-Tool unter „Optionen“ im Hauptmenu „Extras“ nachsehen.
- Falls auch das nicht funktioniert, besteht auch die Möglichkeit, eine ältere Version des Add-Ins namens nnt3\_old\_Version.xla zu installieren.
- Alternativ besteht die Möglichkeit des **direkten Zugriffs auf eine Excel-Mappe**: statt einer Pat-Datei lädt NN-Tool zunächst eine Excel-Anwendung, der User wählt ggf. ein Tabellenblatt aus (bei mehreren Blättern in der Excel-Mappe) und NN-Tool erzeugt und importiert die Datendatei (siehe Datenaufbereitung).

Falls alle genannten Optionen scheitern (d.h. das Add-In lässt sich nicht installieren oder ausführen, Excel-Zugriff nicht möglich), muss die Konvertierung der Exceldaten mit den Excel-Methoden selbst vorgenommen werden. Dazu muss die Excel-Tabelle mittels „Speichern unter“ mit dem Dateityp „.csv“ („character separated values“) abgespeichert werden. Schließen Sie anschließend Excel. Falls dabei ein Hinweis der Art kommt:

„Datei hat nicht das Microsoft Excel Format. Sollen Ihre Änderungen gespeichert werden...“, **dann drücken Sie „nein“**, (d.h. Datei im vorliegenden Format schließen ohne Änderungen zu speichern).

Dabei wird eine Datei vom .csv-Format erzeugt, in der die Daten mittels eines Listentrennzeichens voneinander getrennt werden. Dieses Listentrennzeichen hängt (leider!) von den Windows-Einstellungen ab, genauer von den sogenannten Regionaleinstellungen. Bei deutschen Regionaleinstellungen handelt es sich um das Semikolon, bei englischen um das Komma (was die Fidschi-Inseln verwenden, ist dem Autor nicht bekannt). Eine .csv-Datendatei kann von NN-Tool im Menüpunkt „Datenfile öffnen“ eingelesen werden. Dazu ist dann beim Öffnen der Dateityp auf CSV umzustellen um die Datei auszuwählen. Anschließend wird nach dem oben genannten Listentrennzeichen gefragt.



Falls Ihre .csv-Datei das Semikolon verwendet (Standardeinstellung bei deutschen Windowsversionen), können sie sofort auf „OK“ drücken. Andernfalls müssen Sie das Komma (oder ggf. ein noch anderes Listentrennzeichen) eingeben. Es darf nur genau ein Zeichen eingegeben werden. Damit importiert NN-Tool die Datendatei, erzeugt zusätzlich eine .pat-Datei und sie können wie im Handbuch beschrieben fortfahren.

### **Ordner „Eigene Dateien\NN-Tool“ ist nicht vorhanden**

Kopieren sie in diesem Fall den Ordner NN-Tool aus dem NN-Tool Installationsverzeichnis von Hand nach „Eigene Dateien“. Den Pfad des Installationsverzeichnisses können Sie in NN-Tool unter Extras/Optionen nachsehen.

### **Lernalgorithmus startet nicht**

Stattdessen kommt eine Fehlermeldung. Ursache ist in der Regel, dass das Installationsverzeichnis oder das Anwendungsverzeichnis oder einzelne Dateien schreibgeschützt sind. Abhilfe: Installieren Sie NN-Tool erneut in das Default-Installationsverzeichnis.

---

### ***Excel-Kommunikation funktioniert nicht richtig***

- **Symptom:** die Übertragung von Ergebnisdaten funktioniert nicht. Es wird keine Excelmappe geöffnet.
- **Ursache:** Es laufen bereits eine Vielzahl von Excel-Prozessen im Hintergrund ohne das eine Excel-Anwendung sichtbar ist.
- **Abhilfe:** Starten Sie den Task-Manager (<Strg><Alt><Entf>), wechseln Sie auf die Registerkarte "Prozesse", sortieren Sie nach "Name" und beenden Sie alle Prozesse namens "excel.exe". Ignorieren Sie die Warnung.
- **Hintergrund:** In einigen Excel-Versionen werden beim Beenden einer Excel-Anwendung nicht immer alle zugehörigen Prozesse beendet.

### ***Daten werden nicht richtig interpretiert bzw. Modell ist sehr schlecht***

#### **Ursachen:**

- In vielen Fällen liegt es daran, dass nicht die richtigen Einstellungen für das Dezimaltrennzeichen eingestellt sind, z.B. weil die Daten von einem anderen Rechner mit anderen Regionaleinstellungen stammen oder die Regionaleinstellungen auf Ihrem Rechner verändert wurden. Öffnen Sie dann die von Ihnen verwendete .pat-Datei (ggf. .csv-Datei) mit einem Editor und kontrollieren Sie, ob das verwendete Dezimaltrennzeichen und das verwendete Listentrennzeichen mit den Regionaleinstellungen auf Ihrem Rechner übereinstimmen. (Die Einführung von länderspezifischen Einstellungen war wohl die größte Dummheit in der langen Microsoft-Liste). Ändern Sie die Zeichen durch „Suchen und Ersetzen“ in der Datei ab. Ggf. müssen Sie auch das gesamte Netz anschließend neu trainieren.
- Ihre Daten enthalten neben den numerischen Werten zusätzliche Zeichen (z.B. „>“) und werden deshalb von NN-Tool als Klassifikatoren behandelt. Bereinigen Sie die Daten. Typ: in Excel stehen numerische Daten rechtsbündig in den Zellen, nichtnumerische linksbündig (auch bei falschem Dezimaltrennzeichen).
- Ihre Daten weisen beträchtliche Messfehler auf. Machen Sie einen oder mehrere Kontrollexperimente bei gleichen Eingangsdaten und analysieren Sie die Variation in den Ausgangsdaten. Ihre Modelle können nicht besser als die Messfehler sein.
- Ein oder mehrere wichtige Eingangsgrößen stehen nicht zur Verfügung (nicht gemessen).
- Es stehen nur sehr wenige Datensätze zur Verfügung. Nutzen Sie die Methoden des Crossvalidations insbesondere die Option „Leave-One-out“.

#### **Keine Lösung gefunden?**

Nehmen Sie Kontakt auf:

Email: [info@baermann.de](mailto:info@baermann.de)

Tel: 0049 (0) 211 7489973