

Investox

Die Börsensoftware für professionelle Trader und engagierte Privatanleger

Zusatztool Neuro Plus!

Handbuch Version 7

Version 7.2

Alle Rechte vorbehalten

Copyright © 2015 Knöpfel Software GmbH, Beichstraße 1, D-80802 München

Die vorliegende Bedienungsanleitung darf ohne die schriftliche Genehmigung der Autoren auch nicht auszugsweise reproduziert, übertragen, umgeschrieben oder auf Datenträgern gespeichert werden.

Folgende verwendete Bezeichnungen sind Warenzeichen bzw. eingetragene Warenzeichen der jeweiligen Inhaber: Investox, MS Windows, MS Excel, DAX, Tai-Pan, Market Maker, bis, Morningstar, sino. Weitere Bezeichnungen sind eventuell Warenzeichen oder eingetragene Warenzeichen.

Inhalt

INHALT	3
EINLEITUNG	4
SCHNELLEINSTIEG.....	5
INDIKATOREN UND FUNKTIONEN	8
Indikator „Neuro-Klassifizierung“	8
Indikator „Neuronale Prognose“	12
Hinweise zum Trading und zur Datenfeed-Simulation	15
Weitere Funktionen und Indikatoren von Neuro Plus!	16
WALK-FORWARD-OPTIMIERUNG	21
REFERENZ DER DIALOGS	23
Dialog „Inputs analysieren“	23
Dialog „Klassifizierungs-Modell speichern“	24
Dialog „Klassifizierungs-Modell wählen“	25
Dialog „Netzarchitektur erzeugen“	25
Dialog „Neuro-Klassifizierung einstellen“	27
Dialog „Neuronale Prognose einstellen“	32
Dialog „Walk-Forward-Optimierung vorbereiten“	32
Dialog „Walk-Forward-Abschnitte ermitteln“	33
Dialog „Zeitraum-Justierung einstellen“	34
BEISPIELE ZUR ANWENDUNG.....	36
Beispiele zur Neuro-Klassifizierung.....	36
Beispiele zur Neuro-Klassifizierung mit Walk-Forward	41
Beispiele zur Neuronalen Prognose	43
INDEX	46

Einleitung

Das Paket „Neuro Plus!“ bietet neuartige Indikatoren und Funktionen für Investox XL zur Entwicklung von mechanischen Handelsstrategien.

Im Zentrum stehen die neuen Berechnungsmethoden der **Neuronalen Klassifizierung** mit einem besonders schnellen und leistungsfähigen Trainingsverfahren, das auf Wunsch auch in einer Walk-Forward-Version arbeitet. Noch nie war es so einfach, verschiedene Inputs in eine Handelsstrategie einfließen zu lassen! Mit einfachsten Inputs erreicht die Neuro-Klassifizierung oft eine schon erstaunliche Prognosefähigkeit.

Hinzu kommt ein neuer **Neuro-Indikator**, dessen Berechnungsweise den herkömmlichen Neuronalen Netzen von Investox ähnelt, der aber komplett im Handelssystem beschrieben wird und mit diesem optimiert werden kann. Ein Nachtraining oder eine Änderung von Einstellungen und Inputs ist hier jederzeit möglich, ohne dass das Training komplett neu beginnen muss.

Das Paket Neuro Plus! bietet aber noch vieles mehr:

- ▶ Den **Zeitraum-Justierer** und den **Variablen-Trimmer**– diese ermöglichen eine „live click“-Einstellung von Zeitraum und Variablen im Chart.
- ▶ Unterstützung von **Walk-Forward-Optimierung** auch ohne Neuronale Netze (→ Seite 21)
- ▶ Hilfreiche Indikatoren wie den **Hurst Exponenten** und den Indikator **Kursprognose** als Output für Neuronale Indikatoren.
- ▶ Verfahren zur Inputanalyse wie die **PCA** (Principal Component Analysis) zur Vermeidung redundanter Inputs.

In den folgenden Abschnitten finden Sie nach einem kurzen Tutorium einführende Erläuterungen zu den Indikatoren und Funktionen von Neuro Plus!.

Eine genauere Beschreibung der Einstellmöglichkeiten finden Sie in der Referenz der Dialoge (→ Referenz der Dialoge, Seite 23).

Einen schnellen Einstieg in die Erstellung von Handelsstrategien mit Neuro Plus! ermöglichen die kommentierten Beispielprojekte (→ Beispiele zur Anwendung, Seite 36).

Diese Dokumentation behandelt nur die speziellen Funktionen des Pakets Neuro Plus! und setzt voraus, dass die Grundlagen der Bedienung von Investox bekannt sind.

Schnelleinstieg

In diesem kleinen Tutorium lernen Sie anhand eines sehr einfachen Systems die Arbeitsweise des Indikators „Neuro-Klassifizierung“ kennen.

Legen Sie ein neues Projekt an mit einem Handelssystem für den FDAX auf Tagesbasis. Wir gehen im folgenden davon aus, dass OHLC-Daten für den FDAX auf Tagesbasis ab Ende 1996 vorliegen.

Am besten öffnen Sie hierzu das Projekt „NeuroPlusTutorium“, in dem ein solches Handelssystem unter dem Namen „Schritt 1“ bereits angelegt ist:

www.download.investox.de/neuroplus/NeuroPlusTutorium.inv

Verwenden Sie für das Handelssystem die folgenden Testbedingungen:

Testbedingungen:	
Long+Short testen	
Enter-/Exit-Basis:	Open, Delay=0
Punkte testen	
Startkapital:	15000 €
Wert pro Punkt:	25 €
Entry-/Exit-Gebühren:	3 €
Slippage:	25 €
Money-Management:	Fester Kontrakt, Anzahl 1

Wir fügen nun eine Neuro-Klassifizierung in das Handelssystem ein. Am schnellsten geht das so:

Gehen Sie in die Handelssystem-Einstellungen, Registerkarte „Regeln“ und dort in „Definitionen“.

Geben Sie dort die folgende Zeile mit der Tastatur ein:

```
Global calc NN: NeuroClassify( );
```

Das ist bis jetzt noch ein so nicht gültiger Aufruf des Indikators ohne Einstellungen. Die Einstellungen nehmen wir am besten mit dem speziellen Einstell-Dialog vor. Um diesen zu öffnen, doppelklicken Sie auf den (gelb gefärbten) Namen des Indikators.

Es öffnet sich der Assistenten-Dialog „Neuro-Klassifizierung einstellen“.

Dort geben wir lediglich einen Input für die Klassifizierung ein, und zwar:

```
Dämpfen( ROC(Close,5,%) );
```

Vergessen Sie bitte den abschließenden Strichpunkt nicht. Dieser Input liefert die prozentuale Wertänderung über fünf Tage, mit einer logarithmischen Dämpfung, um die Bedeutung von Extremwerten zu verringern.

Da wir alle anderen Einstellungen auf dem Standard belassen, können Sie nun **Fertigstellen** klicken.

Zurück in den Handelssystem-Einstellungen sehen Sie den kompletten Indikator, definiert als Variable „NN“. Das sieht dann so aus:

```
Global calc NN: NeuroClassify(#>>Dämpfen(ROC(Close,5,%)));  
<<#, #>><Training MaxCluster=50;MaxAbweichung=1;/><Prognose  
ref(roc(open,1,%),2)/><<#);
```

Der Ausdruck <Prognose ref(roc(open,1,%),2)/> zeigt, dass der Indikator die prozentuale Kursänderung des Open-Kurses von übermorgen gegenüber morgen

prognostizieren soll. Dies ist abgestimmt mit der Enter-/Exit-Basis Open, Delay = 0. Damit dies zu korrekten Ergebnissen führt, muss der Indikator, der ja Close-Kurse im Input verwendet, in den Handelsregeln mit Ref(NN, -1) eingesetzt werden:

Geben Sie unter „Enter Long“ ein (falls dort im vorgefertigten Projekt eine „1“ steht, diese bitte löschen):

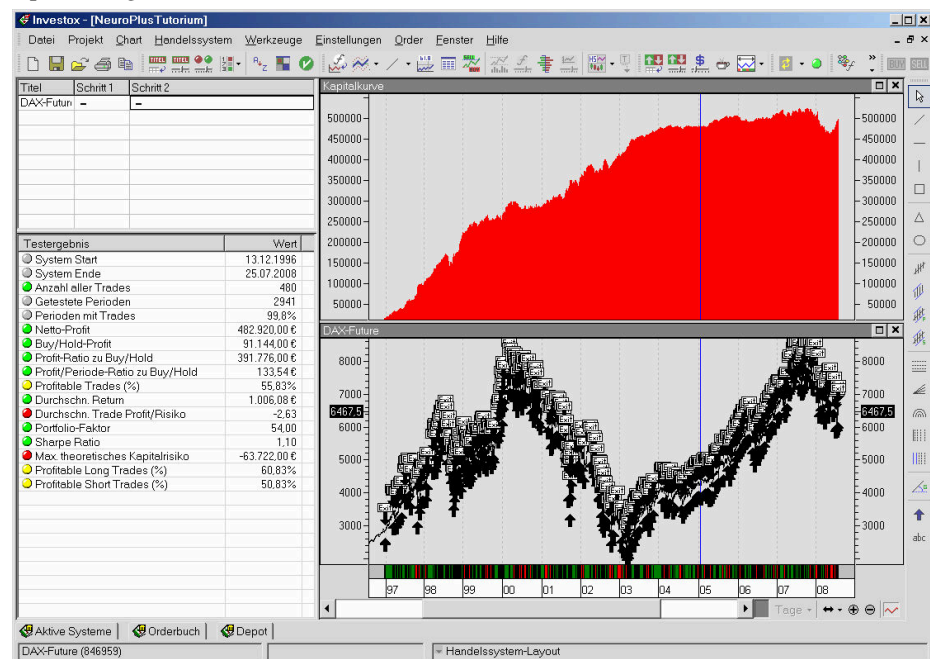
Ref(NN, -1) > 0.2

und entsprechend für Enter Short:

Ref(NN, -1) < -0.2

Ein Kauf bzw. Verkauf soll also stattfinden, wenn die Prognose das Limit +/- 0,2% überschreitet.

Lassen Sie nun noch unter „Zeitraum“ die Zeiträume für Optimierung und Kontrolle anpassen. Klicken Sie dann **OK** bzw. **Fertigstellen**, um das Ergebnis anzeigen zu lassen. Lassen Sie die Kapitalkurve im Chart anzeigen, wenn dies noch nicht der Fall ist. Trotz des sehr simplen Modells liefert das System schon einen ganz ordentlichen Return, vor allem im Optimierungszeitraum.



Das Handelssystem mit diesen Einstellungen finden Sie im Projekt „NeuroPlusTutorium“ unter dem Namen „Schritt 2“.

Was macht der Klassifizierungs-Indikator hier eigentlich? Er untersucht, wie sich die Kurse im Optimierungszeitraum jeweils entwickelt haben, wenn eine bestimmte Wertänderung (Rate-of-Change) gegeben war.

Technisch betrachtet passiert einfach dargestellt das folgende: Er untersucht im Optimierungszeitraum alle Vorkommnisse des Inputs – hier also nur der 5-Perioden-ROC – und bildet dafür (höchstens) 50 Muster (Cluster). Jede Periode des Optimierungszeitraums wird dann dem Muster zugeordnet, dem es am stärksten ähnelt. Anschließend wird analysiert, welches Ergebnis – mit Blick auf den definierten Output – die einzelnen Muster-Cluster im Durchschnitt im Optimierungszeitraum geliefert hätten. Im letzten Schritt wird daraus dann die Ausgabe des Indikators gebildet. Alle Perioden, auch der Kontrollzeitraum, werden klassifiziert und erhalten den entsprechenden Outputwert.

Experimentieren Sie nun mit verschiedenen Einstellungen und prüfen Sie jeweils das Ergebnis, vor allem, wie sich die Kapitalkurve jeweils verändert:

Setzen Sie zunächst im Formeleditor der Handelssystem-Definitionen die Einstellung „MaxAbweichung“ von 1 auf 0.01 und die Einstellung „MaxCluster“ von 50 auf 100. Dadurch wird eine feinere Unterteilung des Inputs durchgeführt. Die Kapitalkurve wird im Optimierungszeitraum etwas glatter (geringere Drawdowns). Zudem werden mehr Trades

generiert. Dies liegt daran, dass sich die Prognose der Cluster, da stärker aufgegliedert, weniger nivellieren, das Limit in den Enter-Regeln aber gleich blieb.

Verändern Sie nun nacheinander die Einstellung für „MaxCluster“ auf 150, 200, 250, 300 etc. und prüfen Sie jeweils, wie sich die Kapitalkurve ändert.

Die Beobachtung wird dabei grundsätzlich sein, dass die Ergebnisse im Optimierungszeitraum immer besser werden, im Kontrollzeitraum dagegen nicht unbedingt. Da die Berechnung (das Training) nach Änderung der Einstellung unmittelbar erfolgt, kann rasch ein sinnvoller Wert ermittelt werden. Dazu kann natürlich auch die Optimierung des Handelssystems oder der Robustheitstest verwendet werden.

Ende dieses Tutoriums

Indikatoren und Funktionen

Indikator „Neuro-Klassifizierung“

Die neuen Indikatoren zur Neuro-Klassifizierung bieten ein Verfahren für ein schnelles Training einer Klassifizierung aufgrund von Musterdaten in einem Lernbereich.

Das besondere Merkmal der Neuro-Klassifizierung ist das so genannte „nicht überwachte“ Training. Das Prognosemodell wird nicht mit Blick auf einen gewünschten Output trainiert, sondern ausschließlich mit Blick auf die im Trainingsbereich auftretenden Inputmuster. Aus diesen werden nach Ähnlichkeit Muster-Gruppen (Cluster) gebildet. Erst im nächsten Schritt wird den Clustern dann ein Output (eine Bedeutung) zugeordnet. Das Ergebnis ist jederzeit reproduzierbar, da das Training keine Initialisierung mit Zufallszahlen voraussetzt.

Der Vorgang bei der Berechnung des Indikators NeuroClassify ist wie folgt:

1. Automatische Skalierung der Daten im Lernbereich. Der Lernbereich entspricht in der Regel dem Optimierungszeitraum des Handelssystems. Auf Wunsch wird hierbei auch eine Reduzierung der Inputs durch eine Principal Component Analysis (PCA) durchgeführt.
2. Bilden von Mustergruppen (Cluster). Dabei werden ähnliche Muster jeweils in einer Gruppe zusammengefasst. Ob Muster dabei noch „ähnlich genug“ sind, wird durch die gewünschte „maximale Abweichung“ eingestellt. Auch die maximale Anzahl der Cluster wird vorgegeben. Die zulässige Abweichung wird falls nötig soweit erhöht, dass die maximale Anzahl der Cluster (mit einer gewissen Toleranz) eingehalten werden kann. Bis zu diesem Schritt erfolgt das Training nicht überwacht, also ohne Zuordnung zu einem Output.
3. Im nächsten Schritt wird den Clustern ein Outputwert zugeordnet. Dieser errechnet sich aus dem Mittelwert der Outputwerte aller einem Cluster zugeordneten Muster.
4. Berechnung des Outputs: Zuordnung der gesamten Zeitreihe auf die trainierten Cluster und entsprechende Konstruktion des Outputs. Hierbei gibt es ebenfalls noch diverse Optionen.
5. In der Walk-Forward-Version des Indikators werden die Schritte 1-4 für jeden Abschnitt einzeln durchgeführt.

Das Training erfolgt also dynamisch „on the fly“ bei der Berechnung des Indikators. Die Auswirkung einer Änderung der Inputs oder einer Einstellung auf das Handelssystem erfolgt ohne nennenswerte bzw. je nach Einstellung mit relativ geringer Verzögerung.

Wird der Titel oder der Optimierungszeitraum geändert, verändert sich auch das Modell und damit der Output. Es besteht aber auch die Möglichkeit (nicht in der WF-Version), das trainierte Modell abzuspeichern. Dies hat zur Folge, dass bei der Berechnung die Schritte 1-3 entfallen und zudem für jeden Titel und Zeitraum dasselbe Modell zugrunde liegt.

Hinweis: Mit dem **Variablen-Trimmer** im Chart können Sie die Einstellung des Indikators mit Live-Kontrolle durchführen (→ Seite 17). Der Umstand, dass das Modell des NeuroClassify-Indikators bei einer Änderung des Optimierungszeitraums (= Trainingszeitraum) sofort neu trainiert wird, wird besonders plastisch mit Hilfe der Funktion **Optimierungszeitraum justieren** (→ Seite 18).

Einsatz in Handelssystemen

Das ausgegebene Ergebnis des Indikators ist letzten Endes eine Prognose des trainierten Outputs (z.B. Wertänderung in drei Perioden). In einem Handelssystem wird der Indikator daher normalerweise vergleichbar einem „normalen“ Neuronalen Netz von Investox eingesetzt: Es wird geprüft, ob der Indikator in Bezug auf ein bestimmtes Limit ein Kauf- oder Verkaufssignal liefert. Also nach folgendem Muster:

```
global calc NN: NeuroClassify (.....);  
Enter Long: NN>Limit  
Enter Short: NN<Limit
```

Zusammenhänge

Für den Einsatz der Klassifizierungs-Indikatoren ist es sinnvoll sich klar zu machen, dass die Komponenten Anzahl Cluster, maximale Abweichung und Ausgabeart mit dem Limit abgestimmt sein müssen (bzw. bei der Optimierung anzupassen sind).

Wenn der Lernbereich zum Beispiel 1000 Perioden umfasst und das Modell auf 50 Cluster begrenzt wird, so fallen im Durchschnitt 20 Muster (Perioden) in ein Cluster (wobei die Muster natürlich nicht gleich verteilt sein müssen). Entsprechend bildet sich auch der Ausgabewert eines Clusters aus durchschnittlich 20 Outputmustern.

Wird die Clusteranzahl nun auf 200 erhöht, so müssen pro Cluster weniger Muster abgebildet werden. Die Outputwerte der einzelnen Cluster sind also im Schnitt weniger nivelliert. Die Limits im Handelssystem können normalerweise entsprechend erhöht werden.

Am deutlichsten ist dieser Effekt, wenn die Ausgabeart „gewichtet“ verwendet wird. Da hier gegebenenfalls mehrere Cluster wirksam werden, kann der Ausgabewert weiter nivelliert, aber eventuell auch signifikanter werden.

Es empfiehlt sich, die Auswirkung der Einstellungen zu studieren, indem Sie den Indikator im Chart anzeigen lassen und dann die Einstellungen entsprechend variieren.

Inputs testen mit Optimierung

Gerade auf Grund dieses Zusammenhangs der gewählten Inputs mit der Einstellung von Clusteranzahl (bei Walk-Forward auch Lernperioden) und dem Limit zur Signalerzeugung bietet sich die Verwendung der automatischen Optimierung an.

Hierzu setzen Sie für die jeweilige Einstellung eine Optimierungsvariable ein und starten nach Veränderung der Inputs die Optimierung. Recht schnell wird dann eine passende Einstellung für den Optimierungszeitraum gefunden.

Auf der nächsten Seite finden Sie die Vorlage für ein Optimierungsgerüst für die Walk-Forward-Klassifizierung. Dabei sind die folgenden Komponenten optimierbar gestaltet:

- Das Limit für das Kauf-/Verkaufssignal
- Die maximale Cluster-Anzahl
- Die maximale Abweichung
- Die Lernperioden pro WF-Abschnitt

Als Beispiel enthält die Vorlage sieben Inputs. Nach einer Änderung der Inputs kann das Modell durch Starten der Optimierung neu justiert werden.

Gerüst für ein optimierbares Walk-Forward-Modell:**Definitionen:**

```

global const Limit: [0.3,0,1,0.01,0.7,0.02,3,];
global calc NN: NeuroClassifyWF(#>>
<inputcache name=NN43424/>
Dämpfen(ROC(close,1,%)); // Input1
Dämpfen(ROC(close,5,%)); // Input2
Dämpfen(ROC(close,10,%)); // Input3
Dämpfen(ROC(close,20,%)); // Input4
HHVBars(close,10); // Input5
LLVBars(close,10); // Input6
ROC(VHF(close,10),3,$); // Input7
<<#, #>>
<Training MaxCluster=[6,2,100,3,30,1,3,I];
MaxAbweichung=[5,1,30,1,10,0.5,3,I];/>
<Prognose Ref(ROC(open,1,%),2) /><Abschnitte Wochen/>
<Lernperioden [100,20,500,20,300,5,3,I]/>
<<#);
Enter Long:
NN>Limit
Enter Short:
NN<Limit

```

Walk-Forward-Version der Neuro-Klassifizierung

Neben der Standardversion der Neuro-Klassifizierung steht auch eine Walk-Forward-Version (WF) zur Verfügung. In der WF-Version erfolgen das Training und die Ausgabe in Abschnitten. Dies können zum Beispiel Tage, Wochen, Monate oder auch andere Abschnitte sein, die durch entsprechende Bedingungen definiert sind.

Für jeden Prognoseabschnitt wird das Modell, also die Zusammenstellung der Muster-Cluster, mit der angegebenen Anzahl davor liegender Perioden neu trainiert. Zusammengefügt ergeben die Prognosen einen durchgängigen Output, der komplett auf „Out-of-Sample“-Daten beruht.

Walk-Forward-Arbeitsweise:

		Prognose mit M1	Prognose mit M2	Prognose mit M3
Trainingsbereich für Modell M1				
	Trainingsbereich für Modell M2			
	Trainingsbereich für Modell M3			

Vor- und Nachteile der Walk-Forward-Berechnung

Ein Vorteil der Walk-Forward-Version besteht darin: Wenn man diesen Indikator im Handelssystem optimieren lässt, wird er wenig Tendenz zur Überoptimierung zeigen, da das Modell ja nur Out-of-sample-Daten ausgibt. Ein vergrößertes Modell (mehr Clustermuster) führt hier *auch im Optimierungszeitraum* nicht zwangsläufig zu einem besseren Ergebnis.

Es kann daher sein, dass bei einer Optimierung der maximalen Clusteranzahl die Genetischen Algorithmen einen relativ kleinen Wert wählen, wenn nur mit reduzierter Modellgröße der jeweilige Out-of-sample-Bereich im Optimierungszeitraum gute Ergebnisse liefert.

Lässt man dagegen in der Version ohne Walk-Forward die maximale Clusteranzahl optimieren, so werden die Genetischen Algorithmen in der Regel einen hohen Wert

einstellen, da dadurch der Indikator bessere Ergebnisse erzielt – die Ausgaben im Optimierungszeitraum entsprechen ja dem Trainingszeitraum des Indikators.

Prinzipiell besteht aber auch bei diesem Verfahren ein Zusammenhang zwischen Modellgröße (Anzahl Inputs, maximale Anzahl Clustermuster, freie Parameter bei der Optimierung) und Überoptimierung.

Ein Vorteil der WF-Version ist auch, dass die Einstellungen des Modells in einem Zeitraum optimiert werden können, und danach in anderen Zeiträumen u.U. auch dann funktionieren, wenn sich die Daten völlig verändern, da die Muster ja neu trainiert werden.

Als möglicher Nachteil ist aber auch zu beachten, dass eine Walk-Forward-Analyse vor allem dann, wenn das Zeitfenster für das Training (Lernperioden) relativ klein ist, zu einer starken Trendfolge neigt. Dies führt naturgemäß dann zu schlechten Ergebnissen, wenn sich der Trend ändert.

Ein weiterer Nachteil ist die längere Berechnungsdauer – je nach Anzahl der Abschnitte und der Lernperioden pro Abschnitt sowie der Einstellung der Clusteranzahl und der maximalen Abweichung kann der Trainings- und Ausgabevorgang eine spürbare Zeit benötigen. Beim realen Einsatz in einem Handelssystem sollte darauf geachtet werden, dass nur die wirklich benötigte Anzahl Abschnitte berechnet werden muss (Reduzierung der Perioden).

Die Neuro-Klassifizierung in der Formelsprache

Die Einstellung der Neuro-Klassifizierung erfolgt am bequemsten im entsprechenden Einstellungsdialog, der beim Einfügen oder Bearbeiten der Einstellungen des Indikators automatisch aufgerufen wird (→ Seite 27). Für manuelle Bearbeitungen direkt im Formeleditor dient die folgende Beschreibung der Syntax:

Schreibweise NeuroClassify

```
NeuroClassify (#>>Inputs<<#, >>#Einstellungen#<<)
```

„Inputs“ ist eine Liste von Inputberechnungen, wobei jeder Input mit einem Strichpunkt endet. Im Bereich „Einstellungen“ werden alle Einstellungen zum Training und zum Output definiert.

Die Walk-Forward-Version unterscheidet sich lediglich in der Benennung des Indikators und in einigen Einstellungsoptionen:

Schreibweise NeuroClassifyWF

```
NeuroClassifyWF (#>>Inputs<<#, >>#Einstellungen#<<)
```

Die Einstellung erfolgt normalerweise im Dialog „Neuro-Klassifizierung einstellen“ (→ Seite 27) und wird mit Hilfe von Kennzeichnungen (sog. „Tags“) im Bereich „Einstellungen“ gespeichert. Die Einstellung kann natürlich auch direkt von Hand vorgenommen werden. Folgende Tags stehen zur Verfügung:

Beschreibung der Kennzeichnungen

Gespeichertes Modell (optional)

<Training Modell = *Modellname* />

Es wird das zuvor gespeicherte Modell mit dem Namen *Modellname* verwendet.

Inputgewichte (optional)

<Gewichte x1; x2; x3; ...xn... /> wobei n die Anzahl der Inputs ist.

Wenn Gewichte verwendet werden, muss für jeden Input ein Gewichtungswert angegeben werden.

Maximale Anzahl Cluster und maximale Abweichung (erforderlich)

<Training MaxCluster = (2-10000); MaxAbweichung = (0,0001-100); />

Je höher die Anzahl der Cluster und je geringer die maximale Abweichung, desto mächtiger und rechenintensiver wird das Modell.

Sammelmethode für die Clusteranalyse (optional)

<Training Sammeln=Gemischt|VornelHinten/>

Standard ohne Angabe = Gemischt

PCA (optional)

<PCA Filter = EinzellGesamtAnzahl; Wert = (0-1000); />

Trainingszeitraum auf Gesamtzeitraum ausweiten (optional)

<Training OptZeitraum = Nein; />

Dies bewirkt, dass nicht nur die Muster des Optimierungszeitraums, sondern alle Muster für das Training verwendet werden.

Definieren eines Lernbereichs (optional)

<Lernbereich *xxx* />

wobei *xxx* für eine Berechnung steht. Nur Perioden, in denen die Berechnung einen Wert $\neq 0$ ergibt, werden für das Training verwendet.

Outputmuster (erforderlich)

<Prognose *Berechnung* />

Die Berechnung gibt an, welcher Prognosewert den Muster-Clustern zugeordnet wird.

Ausgabe (optional)

<Output Ausgabe = Bestes|Treffer|Gewichten; />

Standard ohne Angabe = Gewichten

Output Toleranz (optional)

<Output Toleranz = (0-100000) />

Standard ohne Angabe = 0

Mindestmuster (optional)

<Output MindestMuster = (0-10000) />

Standard ohne Angabe = 1

Outputmodus (optional):

<Output Outputmodus = Output|Outputnummer|AnzahlTreffer|AnzahlCluster;/>

Standard ohne Angabe = Output

Definition der Abschnitte (Walk-Forward, erforderlich)

<Abschnitte Stunde|Tag|Woche|Monat|Quartal|Jahr|*Freie Berechnung* />

Es kann hier wahlweise eine der vorgefertigten Kennzeichnungen für Zeitabschnitte oder aber eine beliebige Berechnung verwendet werden. Im letzteren Falle beginnt ein neuer Abschnitt immer in solchen Perioden, in denen die Berechnung einen Wert $\neq 0$ liefert.

Lernperioden (Walk-Forward, erforderlich)

<LernPerioden *xxx* />

Wobei *xxx* ein konstanter oder berechneter Wert ist.

Prognoseabstand (Walk-Forward, optional)

<Output ProgAbstand = (0-100000) />

Standard ohne Angabe = 0

Indikator „Neuronale Prognose“

Eigenschaften des Indikators „Neuronale Prognose“

Der Neuro-Indikator wird im Unterschied zu den standardmäßigen Neuronalen Netzen von Investox nicht trainiert, sondern wird im Rahmen eines Handelssystems optimiert. Zu unterscheiden sind also:

- Neuronales Netz mit Backpropagation: Das Standardmodell von Neuronalen Netzen in Investox XL seit Version 1. Das Neuronale Netz wird unabhängig von Handelssystemen aufgebaut und trainiert und kann danach als Indikator eingesetzt werden.

- Die Neuronale Prognose: Dieser neue Indikator ist ein Neuronales Netz, dessen Inputs und Architektur in der Formelsprache komplett beschrieben werden. Daher kann das Neuronale Netz komplett im Rahmen des Handelssystems mit Genetischen Algorithmen optimiert werden.

Vorteile der Neuronalen Prognose:

- Die Bewertung der Netzqualität erfolgt mit den Testergebnissen des Handelssystems und ist damit stärker an der Anwendung in einem Handelssystem orientiert.
- Das Neuronale Netz (NN) der Neuronalen Prognose kann bei der Optimierung an bestimmte Optimierungskriterien angepasst werden und auch an die Zusammenarbeit mit anderen Systemkomponenten wie Stops.
- Das NN muss keine bestimmte Zielfunktion lernen, die womöglich mit den gegebenen Inputs gar nicht sinnvoll darstellbar ist.
- Insgesamt ist die Vorgehensweise vereinfacht, da die Konstruktion des NNs, die Verwendung des NNs im Handelssystem und die Optimierung in einem Vorgang vorgenommen werden können.
- Das NN kann auch Aufgaben lernen, für die keine sinnvolle Zielfunktion existiert oder bekannt ist, z.B. einen Selektor, der darüber entscheidet, welcher von verschiedenen gegebenen Strategien das Handelssystem folgen soll. So lassen sich auch ausgefallene Dinge machen, wie das Training eines adaptiven GDs durch ein NN (→ Projekt „NN-Output als variabler GD“, Seite 45).
- Auch nach der Optimierung können Inputs oder auch Verbindungsgewichte verändert werden, ohne dass auf jeden Fall ein Neutraining erforderlich ist.
- Der Fortschritt der Optimierung kann im Handelssystem mitverfolgt werden. Es ist auch ein Nachoptimieren im Handelssystem ohne komplettes Neutraining möglich.
- Die Netzarchitektur kann sehr frei vorgenommen werden und ist nicht den Begrenzungen des Feedbackward-Algorithmus (Schwäche bei mehreren Hiddenschichten) unterworfen. Dabei stehen zum Beispiel auch unterschiedliche Bias-Typen zur Verfügung.

Mögliche Nachteile:

- Die Optimierung kann länger dauern als das Training eines einzelnen Neuronalen Netzes. Allerdings relativiert sich dieser Nachteil, wenn beim Netztraining eine GA-Optimierung eingesetzt wird. Zudem ist kein eigener Zeitaufwand für die Einbindung und Kontrolle des Neuronalen Netzes im Handelssystem erforderlich.
- Es stehen nicht alle Architekturen/Trainingsmöglichkeiten des Backpropagation-Modells zur Auswahl (Rauschen, Schablonenmodell, Rekurrenz).
- Es wird keine vorgegebene Zielfunktion trainiert. Wird also eine möglichst gute Annäherung an eine bestimmte Zielfunktion (Output) benötigt, so kann hierfür weiterhin das Backpropagation-Modell der Neuronalen Netze in Investox verwendet werden.

Der Indikator „Neuronale Prognose“ in der Formelsprache

Schreibweise Neuronale Prognose:

NeuroProg (#>>Inputs<<#, #>>Architektur<<#)

Der Parameter „Inputs“ enthält eine Liste der vom NN zu verwendenden Inputs (vergleichbar einer Inputschablone). Mit „Architektur“ wird die Beschreibung der Units und der Verbindungsgewichte angegeben.

Die Einstellung erfolgt am besten im speziellen Dialog für diesen Indikator, der beim Einfügen oder Bearbeiten der Einstellungen des Indikators automatisch erscheint (→ Seite 27).

Kommentiertes Beispiel für die Neuronale Prognose:

Das folgende Beispiel zeigt die Formulierung eines Neuro-Indikators. Alle Angaben können per Dialog eingegeben werden (→ Seite 27). Die entsprechenden Kennzeichnungen müssen also nicht gelernt werden.

```

global calc NN: NeuroProg(#>>
<Inputcache Name=NNCache4847/>
//Input-Zwischenspeicher, kann verwendet
//werden, da Inputs nicht optimiert werden.
//Definition von 3 Inputs:
roc(close,1,%)>0;
roc(close,5,%)>0;
roc(close,20,%)>0;
<<#, #>>

//Hier beginnt die Beschreibung der Netz-Architektur. Das NN verwendet 3
//Inputs und 5 Hidden-Units (erforderliche Angaben). Die Kennzeichnung
//„HStrukt“ speichert genauere Angaben zur Verteilung der Hidden-Units
//auf Schichten und zu den Biastypen (Angabe optional):
<NNDef Inputs=3;Hidden=5;HStrukt=3,2S; 0;/>

//Es folgt die Beschreibung der einzelnen Hidden-Units und ihrer
//Verbindungsgewichte, die optimiert werden. Die Hidden-Units 1-3 sind
//mit dem Input verlinkt, die nachfolgenden Units 4-5 mit der
//darüber liegenden Hidden-Schicht.
<Unit:H1
Bias=[0.038,-100,100,-0.083,0.083,0.008,3];
In1=[0.472,-100,100,-1,1,0.2,3];
In2=[0.166,-100,100,-1,1,0.2,3];
In3=[-0.490,-100,100,-1,1,0.2,3];
/>
<Unit:H2
Bias=[0.023,-100,100,-0.083,0.083,0.008,3];
In1=[0.457,-100,100,-1,1,0.2,3];
In2=[0.322,-100,100,-1,1,0.2,3];
In3=[-0.041,-100,100,-1,1,0.2,3];
/>
<Unit:H3
Bias=[0.048,-100,100,-0.083,0.083,0.008,3];
In1=[-0.222,-100,100,-1,1,0.2,3];
In2=[0.319,-100,100,-1,1,0.2,3];
In3=[0.346,-100,100,-1,1,0.2,3];
/>
<Unit:H4
//Die Units 4-5 verwenden den Biastyp „Schwellenwert“
BTyp=S;
Bias=[0.501,-100,100,-1.000,1.000,0.100,3];
H1=[-0.235,-100,100,-1,1,0.2,3];
H2=[-0.026,-100,100,-1,1,0.2,3];
H3=[-0.226,-100,100,-1,1,0.2,3];
/>
<Unit:H5
BTyp=S;

```

Fortsetzung auf der nächsten Seite...

```

Bias=[0.066,-100,100,-1.000,1.000,0.100,3];
H1=[0.470,-100,100,-1,1,0.2,3];
H2=[-0.050,-100,100,-1,1,0.2,3];
H3=[-0.178,-100,100,-1,1,0.2,3];
/>
//Die Output-Unit, hier verlinkt mit den Units 4-5, darf nicht fehlen:
<Unit:0
Bias=[0.052,-100,100,-0.125,0.125,0.013,3];
H4=[0.241,-100,100,-1,1,0.2,3];
H5=[0.014,-100,100,-1,1,0.2,3];
/>
<<#);

```

Hinweise zum Trading und zur Datenfeed-Simulation

Beim realen Trading-Einsatz und bei der Datenfeed-Simulation eines Handelssystems ist darauf zu achten, dass die Signalberechnung möglichst schnell erfolgt. Die wichtigste Voraussetzung dafür ist eine Begrenzung der Perioden zur Signalberechnung auf das Nötige (Handelssystem einstellen / Aktualisierung). Bei den Indikatoren von Neuro Plus! ist hierbei das folgende zu beachten:

Trading mit der Neuro-Klassifizierung

Die Neuro-Klassifizierung wird standardmäßig bei jeder Berechnung neu mit den Daten des Optimierungszeitraums trainiert. Das Modell ändert sich daher, wenn der Titel oder der Optimierungszeitraum wechselt oder wenn nicht alle Daten aus dem Optimierungszeitraum zur Verfügung stehen. Daher empfiehlt es sich für eine möglichst kurze und stabile Berechnung im Realeinsatz, das Klassifizierungs-Modell zu speichern und dann dieses gespeicherte Modell zu verwenden (→ Dialog „Klassifizierungs-Modell speichern“, Seite 24).

Schritt-für-Schritt-Anleitung zum Speichern eines Modells:

1. Setzen Sie im Formeleditor den Cursor auf den Indikator „NeuroClassify“.
2. Klicken Sie mit der rechten Maustaste auf den Indikator und wählen Sie im Kontextmenü den Befehl "Klassifizierungs-Modell speichern".
3. Geben Sie einen Namen für das Modell ein und klicken Sie dann **OK**.
4. Es erfolgt die Nachfrage: „Soll das gespeicherte Modell in der Formel zur Verwendung eingetragen werden?“. Beantworten Sie diese Frage durch Klicken auf **JA**.
5. Zurück im Editor finden Sie in der Einstellung des Indikators in der Kennzeichnung <Training/> den Eintrag Modell=xy, wobei xy der angegebene Name ist.
6. Speichern Sie das Handelssystem.

Durch diese Einstellung verwendet der Indikator in diesem Handelssystem stets dasselbe gespeicherte Modell, auch wenn sich der Titel oder der Optimierungszeitraum ändert.

Trading mit der Walk-Forward-Version

In der Walk-Forward-Version erfolgen Berechnung (Ausgabe) und Training in einem Vorgang. Eine Speicherung des trainierten Modells ist hier nicht möglich. Allerdings werden zur Berechnung des aktuellen Signals auch nur die Inputs des aktuellen Abschnitts zuzüglich der Lernperioden benötigt. Wenn also zum Beispiel der größte Walk-Forward-Abschnitt 120 Perioden umfasst, und 200 Lernperioden eingestellt sind, so genügen zur Berechnung 320 Perioden zuzüglich der Anzahl Perioden, die für die Berechnung der Inputs vonnöten ist:

Anzahl Perioden für Signalgenerierung = Perioden größter Abschnitt
 + Lernperioden
 + Für Inputs benötigte Perioden

Trading mit der Neuronalen Prognose

Bei diesem Indikator ist das trainierte Modell vollständig mit der Beschreibung der Verbindungsgewichte in den Handelsregeln definiert. Für jeden Titel und Zeitraum wird der Output gemäß der jeweiligen Basisdaten berechnet. Es findet außerhalb der Optimierung des Handelssystems kein Training mehr statt. Für einen korrekten Real- oder Simulationseinsatz genügt es daher, wenn alle Inputs korrekt berechnet werden können.

Weitere Funktionen und Indikatoren von Neuro Plus!

Indikator Kursprognose

Zeigt den vorherrschenden zukünftigen Trend für einen angegebenen Zeithorizont.

Der Indikator „Kursprognose“ ist in Neuro Plus! enthalten und dient zur Definition einer Kursprognose über einen bestimmten Zeithorizont. Er soll ähnlich wie der Indikator „ZigZag“ möglichst gut den aktuellen Trend in Bezug auf eine angegebene Änderungsrate bestimmen, dabei aber (anders als ZigZag) nur eine definierte Anzahl Perioden in die Zukunft blicken. So kann dieser Indikator auch in Walk-Forward-Indikatoren als Outputziel eingesetzt werden. Dabei ist der Prognosehorizont durch einen entsprechenden „Prognoseabstand“ auszugleichen.

Schreibweise

Kursprognose(Daten, Änderungsrate, Horizont, Punkte/Prozent, Kurs/Signal)

Die Änderungsrate gibt an, welche minimale Kursänderung für einen Trend vorausgesetzt wird.

Der „Horizont“ ist die Anzahl der Perioden in die Zukunft, die zur Berechnung verwendet werden sollen.

Zusätzlich kann eingestellt werden, ob die Berechnung prozentual oder absolut in Punkten erfolgt.

„Kurs/Signal“ gibt an, ob die Stärke der verbleibenden Kursänderung oder nur ein Signal -1 / 1 für fallende / steigende Kurse ausgegeben wird.

Beispiel

KursPrognose(Open, 2, 10, %, K)

Betrachtet die aktuelle Trendrichtung mit einer Mindestbewegung von 2% in den nächsten 10 Perioden und liefert die noch zu erwartende Kursänderung in dieser Trendrichtung.

Indikator Hurst-Exponent

Liefert eine Schätzung dafür, ob eine Zeitreihe gerade eher einen Trend oder Antitrend ausbildet oder aber sich als Zufallsprozess ausprägt.

Der Hurst-Exponent schwankt im Bereich von 0 bis 1. Ein Wert von 0,5 zeigt den höchsten Grad an Zufälligkeit der Zeitreihe an. Je stärker der Wert gegen 1 läuft, desto stärker ist das Trendverhalten der Zeitreihe, die Zeitreihe neigt also zu einem gleich bleibenden Verhalten. Je stärker der Wert umgekehrt gegen 0 läuft, desto stärker ist ein Anti-Trendverhalten ausgeprägt (nach steigenden folgen eher fallende Werte und umgekehrt).

Die Schätzung des Hurst-Exponenten erfolgt derart, dass von Zeitfenstern verschiedener Größe jeweils eine so genannte „rescaled range“ berechnet wird. In die sich ergebenden Werte wird eine lineare Regression gelegt, deren Steigung den Hurst-Exponenten ergibt. Diese Berechnung wird für jede Periode der Zeitreihe durchgeführt, so dass sich ein fortlaufendes Bild der Entwicklung des Hurst Exponenten bietet.

Das Ergebnis hängt maßgeblich davon ab, welche Größe die Zeitfenster verwenden (im Indikator einstellbar). Je größer das größte Zeitfenster, desto längerfristig ist die Aussage.

Der Indikator sollte mit stationären Daten eingesetzt werden, also Daten, deren Wertebereich sich im Untersuchungszeitraum nicht ändert (zum Beispiel mit der ROC von Kursen anstatt mit deren Absolutwerten).

→ Hurst-Exponent Projekt, Seite 40

Schreibweise

```
HurstExpo(Daten, Min Perioden, Max Perioden)
```

Liefert eine Schätzung des Hurst-Exponenten für die angegebenen Daten. Die Berechnung verwendet Zeitfenster beginnend mit „Max Perioden“. Weitere Zeitfenster werden jeweils durch Halbierung der Periodenzahl gebildet, wobei der Wert „Min Perioden“ als kleinstes Zeitfenster nicht unterschritten wird.

Beispiel

```
HurstExpo(ROC(Close,1,%), 10, 320)
```

Liefert eine Schätzung des Hurst-Exponenten für die prozentuale 1-Perioden-Kursänderung, wobei Zeitfenster zwischen 10 und 320 Perioden verarbeitet werden.

Indikator „Mehrere Daten markieren“

Markiert mehrere Datumswerte auf einen Schlag und berechnet auf Wunsch daraus Positionen.

Für manche Zwecke kann es wünschenswert sein, mit einem Indikator nicht nur eine, sondern gleich mehrere Datenperioden zu markieren. Ein Beispiel dafür ist das Trainingsziel eines Neuronalen Netzes. Üblicherweise wird dieses durch eine Berechnung, wie z.B. die Kursänderung über 5 Perioden, erzeugt. Als Alternative zu diesem relativ starren Verfahren ermöglicht es der Indikator „DateMarkMulti“, manuell Trends zu markieren.

Hinweis: Die Einstellung des Indikators wird durch den Befehl PERIODENMARKIERUNG ERZEUGEN im Menü CHART unterstützt. Mit diesem können Sie aus Trends, die Sie visuell im Chart markiert haben, eine Einstellung für den Indikator erzeugen lassen (→ Periodenmarkierung erzeugen, Seite 19).

Schreibweise

```
DateMarkMulti(Datumswerte, Methode, Startposition)
```

Datumswerte: In diesem Parameter übergeben Sie die Datumswerte, die markiert werden sollen (jeweils durch Strichpunkt getrennt).

Methode: Hier wählen Sie zwischen „Markierung“ (jede markierte Periode erhält den Wert 1), „Positionswechsel“ (wechselt zwischen Enter Long = 1 und Enter Short = -1, sonst 0) oder „gehaltene Positionen“ (liefert immer 1 / -1).

Startposition: Gibt an, mit welcher Position begonnen wird (Long oder Short).

Beispiel

```
DateMarkMulti(#04.03.2009, 12.01.2011, 25.11.2011, 23.02.2012, 28.06.2012#, Pos, Long)
```

Markiert Positionen, die an den angegebenen Daten wechseln und beginnt dabei mit Long (= 1).

Variablen-Trimmer im Chart

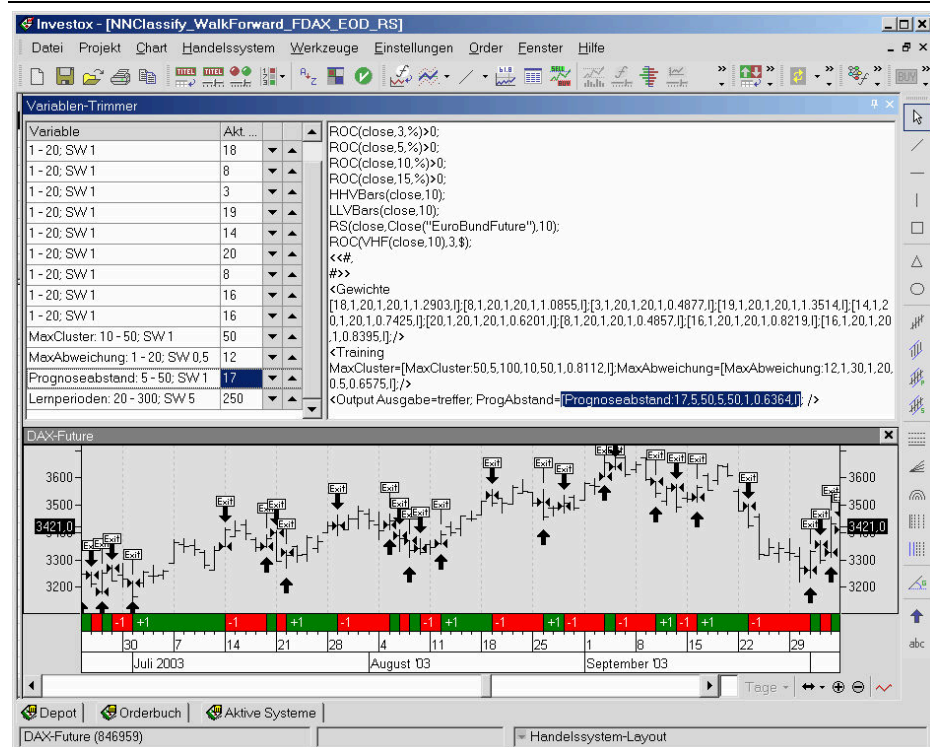
- Wird in den Chart-Einstellungen aktiviert/deaktiviert.

Der Variablen-Trimmer ermöglicht einen direkten Zugriff auf die Optimierungsvariablen aus dem Definitionen-Bereich der Handelsregeln. Damit können Sie die Variablen mit Live-Kontrolle direkt im Chart einstellen.

Sobald Sie den Variablen-Trimmer in den Chart-Einstellungen aktiviert haben, werden alle in den Definitionen der Handelsregeln vorhandenen Optimierungsvariablen aufgelistet. In der Spalte „Akt. Wert“ finden Sie die jeweilige aktuelle Einstellung der Variable. Sobald Sie auf den Wert klicken, können Sie ihn verändern. Nach Betätigen der Eingabetaste (oder wenn Sie

den Wert mit dem Scrollpfeil ändern) erfolgt sofort eine Aktualisierung des Handelssystems mit der neuen Einstellung.

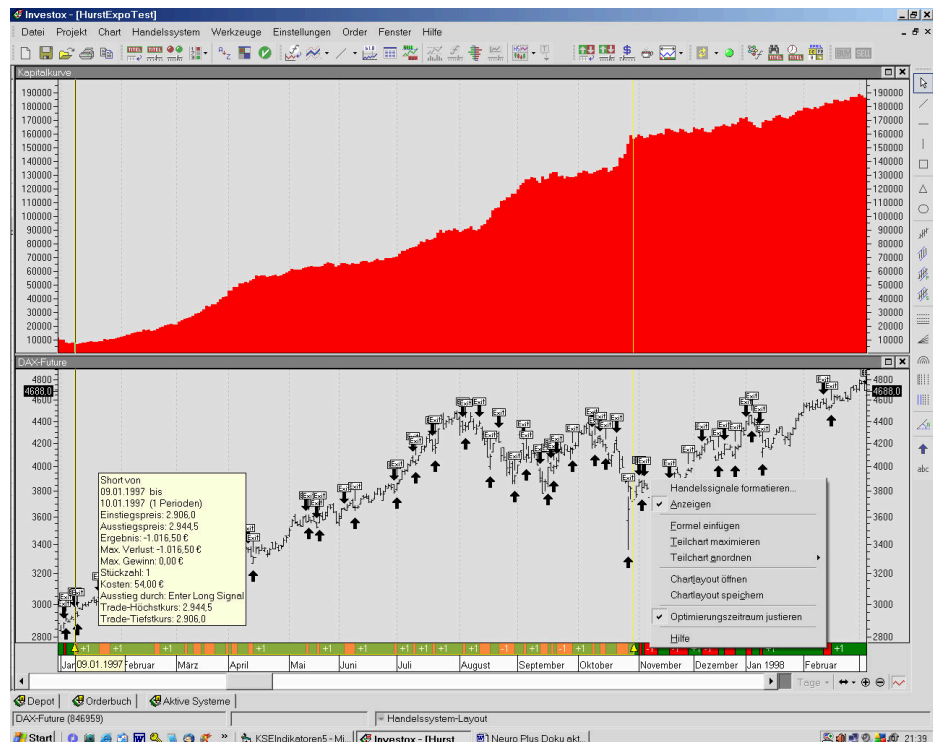
Hinweis: Sie können Optimierungsvariable im Formeleditor auch mit einem Namen versehen. Der Name wird dann ebenfalls in der Liste angezeigt und ermöglicht so eine bessere Übersicht.



Funktion „Optimierungszeitraum justieren“

- Wird im Kontextmenü der Signalleiste des Charts aktiviert/deaktiviert.

Wenn diese Option aktiviert ist, wird der Optimierungszeitraum in der Signalleiste des Charts grafisch durch eine Einfärbung markiert. Der Beginn und das Ende des Optimierungszeitraums, die jeweils mit einem Aufwärtspfeil gekennzeichnet sind, können mit der Maus verschoben werden. Nach Loslassen der Maustaste wird das Handelssystem sogleich mit dem neuen Optimierungszeitraum aktualisiert.



Interessant ist diese Funktion vor allem im Zusammenhang mit dem Indikator „NeuroClassify“, da hier eine Änderung des Optimierungszeitraums unmittelbar in ein mit dem neuen Zeitraum trainiertes Modell umgesetzt wird. So kann interaktiv getestet werden, wie sich eine Änderung des Trainingszeitraums auswirkt.

(→ Dialog „Zeitraum-Justierung einstellen“, Seite 34)

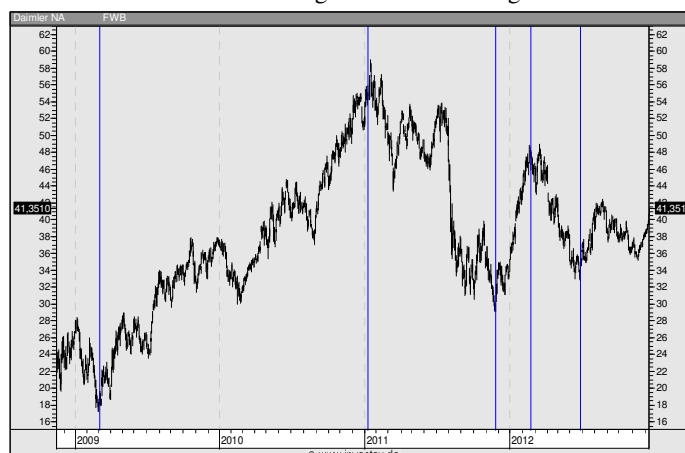
Periodenmarkierung erzeugen

- Befehl im Menü CHART

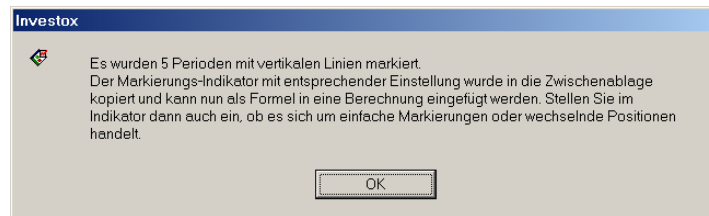
Mit Hilfe dieser Funktion kann der Indikator „DateMarkMulti“ (→ Seite 17) bequem mit Datumsinformationen für die Markierungen versorgt werden.

So gehen Sie vor, um einen Indikator zu erzeugen, der die gewünschten Positionen manuell markiert (kann dann als Output für ein Neuronales Netz eingesetzt werden):

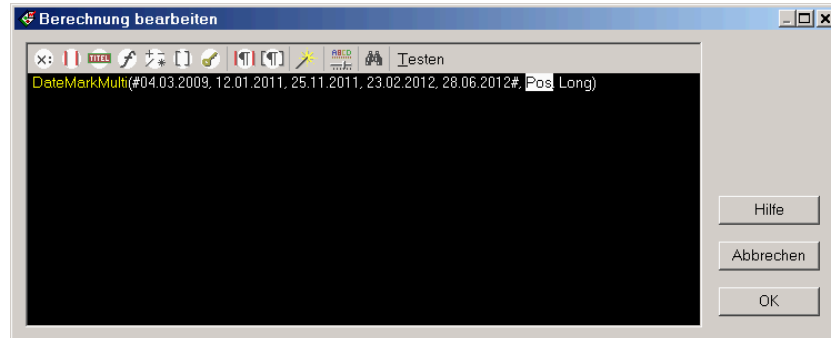
1. Markieren Sie alle gewünschten Perioden im Chart mit jeweils einer vertikalen Linie. Beim Einfügen mehrerer Linien bietet es sich an, die Option **Automatisch auf Auswahlmodus zurückschalten** in den Chart-Werkzeugen zu deaktivieren, so dass das Vertikale-Linien-Werkzeug nach einem Einfügen aktiv bleibt.



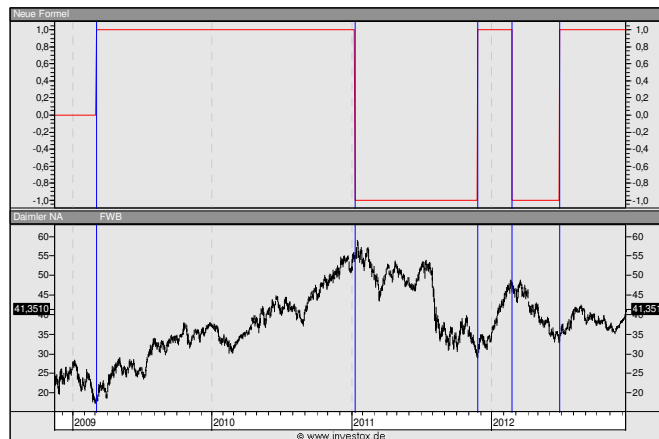
2. Wählen Sie dann den Befehl CHART / PERIODENMARKIERUNG ERZUEGEN. Investox antwortet:



3. Fügen Sie eine Formel in einen neuen Teilchart im Chart ein. In der Formelberechnung fügen Sie per Kontextmenü / EINFÜGEN den erzeugten Indikator ein. Ändern Sie dann noch den Parameter „Methode“ auf Positionen (Pos):



Das Resultat im Chart ist dann wie folgt:



Im oberen Teilchart werden die Positionen Long/Short mit 1/-1 der roten Linie gekennzeichnet. Den Indikator können Sie dann als Output bzw. Trainingsziel in ein Neuronales Netz einfügen. Beachten Sie dabei, dass das Neuronale Netz nur in dem Zeitbereich trainiert wird, für den Sie Markierungen vorgenommen haben. Kontrollieren Sie stets im Chart, dass der erzeugte Indikator Ihren Vorstellungen entspricht.

Optimierungsvariablen dokumentieren

- Befehl im Menü HANDELSYSTEM

Ermöglicht einen Vergleich der Optimierungsvariablen aller Handelssysteme im Projekt.

Diese Funktion listet die Werte aller bezeichneten, also mit Namen versehenen Optimierungsvariablen des markierten Handelssystems auf; ebenso die gleichnamigen Optimierungsvariablen aller anderen Systeme im Projekt. Anhand der auf diese Weise erzeugten Tabelle können die Werte der Optimierungsvariablen verglichen werden. Dies kann insbesondere dann von Interesse sein, wenn die Handelssysteme mit derselben Strategie, aber unterschiedlichen Zeiträumen optimiert wurden (Walk-Forward-Optimierung).

→ Einleitung Walk-Forward-Optimierung, Seite 21

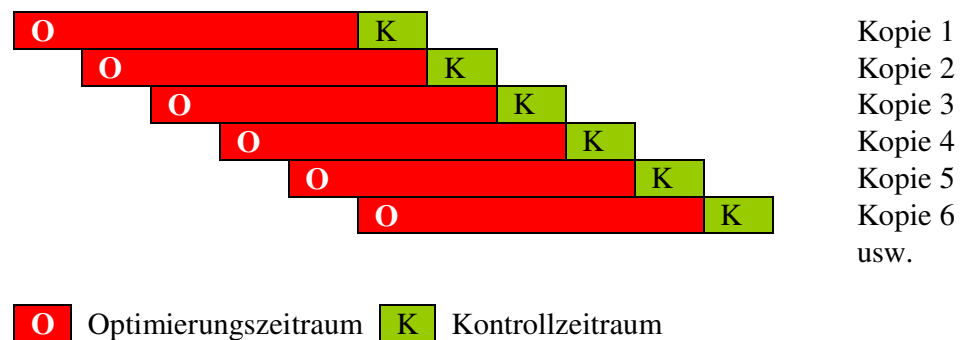
Walk-Forward-Optimierung

Neuro Plus! unterstützt auch die Walk-Forward-Optimierung von Handelssystemen ohne den Einsatz von Neuronalen Netzen. Um ein Handelssystem an unterschiedliche Marktphasen wie trendstarke oder eher oszillierende Märkte anzupassen gibt es ja grundsätzlich zwei Möglichkeiten:

- Durch passende Indikatoren wird ermittelt, welche Marktphase erwartet wird. Es werden dann unterschiedliche Indikatoren oder Einstellungen zur Signalgenerierung eingesetzt.
- Oder aber die Einstellungen des Handelssystems werden immer wieder neu justiert, um an die aktuelle Marktphase angepasst zu werden.

Für die zweite Vorgehensweise eignet sich die Walk-Forward-Optimierung. Dabei wird davon ausgegangen, dass es sinnvoll ist, die Einstellungen (Optimierungsvariablen) des Handelssystems in bestimmten Abständen immer wieder neu einstellen (optimieren) zu lassen.

Damit für den Backtest ein durchgehender Kontrollzeitraum entsteht, ist folgendes Schema der Abschnittsbildung für die Optimierung empfehlenswert. In Investox wird dazu für jeden Abschnitt eine eigene Kopie des Grund-Handelssystems angelegt:



Sowohl der Optimierungs- als auch der Kontrollzeitraum werden also so verschoben, dass der neue Kontrollzeitraum am Ende des vorangehenden Kontrollzeitraums beginnt.

In Investox ist die Vorgehensweise zur Walk-Forward-Optimierung wie folgt:

A. Das Grundsystem anlegen: Das Grundsystem enthält alle Einstellungen und Optimierungsvariablen, die für die Walkforward-Optimierung benötigt werden. Zudem sollten die gewünschten Titel für die Optimierung gewählt und dieselben Titel auch für die Erzeugung aktueller Signale aktiviert werden. Stellen Sie am besten auch gleich den gewünschten Kontroll- und Optimierungszeitraum ein, mit dem die Walk-Forward-Optimierung starten soll.

B. Walk-Forward-Optimierung vorbereiten: Mit dieser Funktion im Menü HANDELSYSTEM werden die Abschnitte der Walk-Forward-Optimierung definiert und für jeden Abschnitt eine Kopie des Grundsystems, also ein weiteres Handelssystem, erzeugt. Dabei stehen mehrere Optionen und Mechanismen zur Verfügung. Die Abschnitte können wahlweise manuell angegeben oder automatisch erzeugt werden, auf Wunsch sogar an Hand eigener Berechnungen (→ Dialog „Walk-Forward-Optimierung vorbereiten“, Seite 32). Die Einstellungen des Dialogs werden im Handelssystem gespeichert und stehen daher sofort für eine erneute Einteilung mit modifizierten Handelsregeln zur Verfügung.

C. Alle Abschnitte optimieren lassen: Die Optimierung aller erzeugten Handelssysteme erfolgt bequem über die Optimierungs-Warteschlange. In diese werden die Systeme bei Verlassen des Dialogs in Schritt B. auf Wunsch automatisch gestellt.

D. Die Ergebnisse überprüfen: Im Projekt-Portfoliotest kann nun das Ergebnis der Optimierung analysiert werden. Dieser Test berechnet das Ergebnis aller Systeme im Projekt jeweils für alle aktiven Titel. Die Ergebnisse für den Kontrollzeitraum zeigen dann entsprechend einen lückenlosen Test mit „Out-of-Sample“-Daten. Sollte das Ergebnis nicht überzeugen, wird das Grundsystem in der Regel modifiziert und dann mit **Schritt B.** erneut gestartet.

E. Die Variablen dokumentieren: Mit dieser Funktion im Menü HANDELSYSTEM können die Werte der Optimierungsvariablen der einzelnen Abschnitte bequem verglichen werden. Damit entsteht ein Bild, die Werte welcher Variablen wie stark in den einzelnen Abschnitten schwanken (→ Optimierungsvariablen dokumentieren, Seite 20).

F. Das System im Handel einsetzen: Wird ein mit Walk-Forward optimiertes System für den Handel eingesetzt, sollte es gemäß der zugrunde liegenden Idee immer dann neu optimiert werden, wenn sich ein neuer Walk-Forward-Abschnitt ergibt (entsprechend der Abschnitte beim Backtest).

Vorteile / Nachteile der Walk-Forward-Optimierung

Vorteile

Großer Kontrollzeitraum (Out-of-Sample), wenn man die einzelnen Kontrollzeiträume im Backtest zusammenfasst.

Einfache Anpassung des Systems an unterschiedliche Marktphasen durch Neuoptimierung

Nachteile

Jeweils kürzere einzelne Optimierungs- und Kontrollzeiträume. Dies kann vor allem bei wenigen Trades die Aussagekraft einschränken.

„Nachhinken“ der Einstellungen durch Optimierung mit einem zurückliegenden Zeitbereich (keine antizipierende Wirkung von Indikatoren zur Phasenunterscheidung)

Referenz der Dialoge

Dialog „Inputs analysieren“

Dient zur Analyse der Inputs einer Neuronalen Prognose oder Klassifizierung.

- Der Dialog wird geöffnet, wenn Sie in den Dialogen „Neuronale Prognose einstellen“ oder „Neuro-Klassifizierung einstellen“ die Schaltfläche **Analyse...** klicken.

Formel	Min.	Max.	Mittel	Std.-Abw.	% <> 0	Hurst
<input checked="" type="checkbox"/> (POszi(close,1,[115,2,200,2,2...	0.0000	1.0000	0.5725	0.4948	57.25%	0.940
<input checked="" type="checkbox"/> (POszi(close,1,[30,2,200,2,20...	0.0000	1.0000	0.5792	0.4938	57.92%	0.798
<input checked="" type="checkbox"/> (POszi(close,1,[26,2,200,2,20...	0.0000	1.0000	0.5792	0.4938	57.92%	0.783
<input checked="" type="checkbox"/> (POszi(close,1,[124,2,200,2,2...	0.0000	1.0000	0.5720	0.4949	57.20%	0.925
<input checked="" type="checkbox"/> (POszi(close,1,[189,2,200,2,2...	0.0000	1.0000	0.5836	0.4931	58.36%	0.975
<input checked="" type="checkbox"/> (POszi(close,1,[136,2,200,2,2...	0.0000	1.0000	0.5710	0.4951	57.10%	0.915
<input type="checkbox"/> Correl(close,Ref(close,-[33,2...	-0.9504	0.9404	0.0507	0.5023	100.00%	0.685
<input type="checkbox"/> Correl(close,Ref(close,-[7,2,...	-0.9752	0.9511	0.0357	0.5150	100.00%	0.611

Der Dialog informiert über einige statistische Kennzahlen der Inputs und bietet für den Indikator „Neuronale Prognose“ die Möglichkeit, eine Skalierung der Daten einzufügen. Die Analyse wird automatisch beim Öffnen des Dialogs durchgeführt.

Bereich: Für die Analyse wird der im Handelssystem eingestellte Optimierungszeitraum verwendet, dessen Start und Ende hier angezeigt wird.

PCA: In diesem Bereich finden Sie das Ergebnis einer so genannten Principal Component Analysis. Die Analyse zeigt, wie viele Inputs jeweils zur Darstellung für 99%, 95%, 90% und 75% der Zeitreiheninformation benötigt werden, wenn man eine entsprechende Umrechnung vornimmt (die alle Inputs berücksichtigt). Diese Information bietet ein Maß für die Redundanz der Inputinformationen aus der Sicht der PCA. Die PCA macht aber keine Aussage darüber, welche Relevanz die einzelnen Inputs besitzen, da alle Inputs bei der Umrechnung verwendet werden.

Die Ergebnisse für die einzelnen Inputs werden tabellarisch (pro Input eine Zeile) aufgelistet:

Min./Max./Mittel/Std.Abw.: Diese grundlegenden Angaben für Minimum, Maximum, Mittelwert und Standardabweichung zeigen an, in welchem Wertebereich sich die Inputs bewegen. Von Min./Max. hängt es auch ab, ob Investox eine Skalierung der Daten vorschlägt (angezeigt durch ein gesetztes Häkchen am Anfang der Zeile).

Die Skalierung erfolgt, sobald Sie den Dialog mit **OK** verlassen. Der Assistent fügt dann automatisch die notwendigen Ergänzungen in die Berechnung ein.

% <>0: Bei binären Inputs (die nur die Wahrheitswerte 0 oder 1 liefern) ist es interessant zu wissen, wie oft der Wahrheitswert 0 bzw. 1 auftritt. Definiert man als Input eine Regel, die nur selten oder sehr oft zutrifft, kann dieser Input auch nur an entsprechend wenigen Perioden Einfluss nehmen.

Hurst: Der Hurst-Exponent schätzt das Trendverhalten einer Zeitreihe. Er nimmt Werte im Bereich von 0 bis 1 an. Ein Wert von 0,5 zeigt den höchsten Grad an Zufälligkeit der

Zeitreihe an. Je stärker der Wert gegen 1 läuft, desto stärker ist das Trendverhalten der Zeitreihe, die Zeitreihe neigt also zu einem gleich bleibenden Verhalten. Je stärker der Wert umgekehrt gegen 0 läuft, desto stärker ist ein Anti-Trendverhalten ausgeprägt (nach steigenden folgen eher fallende Werte und umgekehrt). Die hier vorgenommene Schätzung verwendet zur Berechnung Zeitfenster von 10 bis 500 Perioden (→ Indikator Hurst-Exponent, Seite 16).

Dialog „Klassifizierungs-Modell speichern“

Dient zum Speichern eines Klassifizierungs-Modells.

- Erscheint, wenn Sie in den Handelsregeln im Kontextmenü des Formeleditors den Befehl „Klassifizierungs-Modell speichern“ wählen.

Klassifizierungs-Modell speichern

Name: BundEoDTechnisch18_6_2009

Anmerkungen:

Informationen:

Erstellt am: 14.07.2009 15:47:48
 Titel: GERMAN BUND-%
 Komprimierung: Täglich
 Trainings-Start: 27.07.1990
 Trainings-Ende: 16.04.2003
 Anzahl Kluster: 49

Inputs

calc gd: GD(close, [23,1,300,3,30,1,1.6848,1], S);
 ROC(close,1,%);
 ROC(close,5,%);
 ROC(gd,5,%)>0;
 ROC(gd,10,%)>0;
 ROC(gd,15,%)>0;
 ROC(gd,20,%)>0;

Einstellungen

<Training MaxKluster=[47,5,60,5,60,5,1.7573,1];MaxAbweichung=1; /><Output

Hilfe Abbrechen OK


Der Dialog dient zum Speichern eines trainierten Klassifizierungs-Modells. Ein Klassifizierungs-Modell kann wie folgt gespeichert werden:

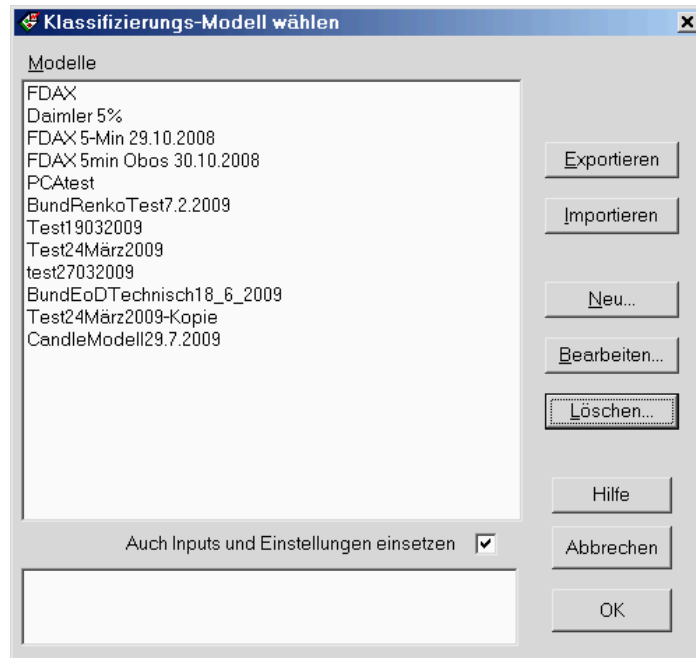
1. Platzieren Sie im Formeleditor des Handelssystems den Eingabecursor auf den Indikator NeuroClassify().
2. Klicken Sie mit der rechten Maustaste auf den Indikator, es erscheint das Kontextmenü.
3. Klicken Sie im Kontextmenü den Befehl **Klassifizierungs-Modell speichern...**
4. Geben Sie im Dialog einen Namen für das Modell ein oder wählen Sie einen vorhandenen Namen nach Klicken auf [...], wenn Sie das entsprechende Modell überschreiben möchten.

Ein einmal gespeichertes Modell kann danach durch Angabe seines Namens in den Einstellungen des Indikators in jedem Kontext verwendet werden (→ Dialog „Neuro-Klassifizierung einstellen“, Seite 27).

Dialog „Klassifizierungs-Modell wählen“

Dient zur Auswahl eines gespeicherten Klassifizierungs-Modells.

- **So öffnen Sie den Dialog:** Klicken Sie im Dialog „Neuro-Klassifizierung einstellen“, Registerkarte „Training“ neben **Gespeichertes Modell verwenden** auf die Schaltfläche  (nicht verfügbar für WF-Version).



Ein gespeichertes Modell enthält die trainierten Muster-Cluster, die Skalierungsangaben sowie die übrigen Einstellungen und Inputs, die beim Training verwendet wurden.

Wählen Sie in der Liste das gewünschte Modell und klicken Sie dann **OK**, um es in das Einstellfeld des aufrufenden Dialogs zu übernehmen.

Auch Inputs und Einstellungen einsetzen: Standardmäßig werden beim Auswählen eines Modells auch die Inputs und die Einstellungen aus dem gespeicherten Modell in die Einstellbox übernommen. Deaktivieren Sie diese Option, wenn Sie nur die gespeicherten Muster-Cluster und Skalierungsangaben verwenden möchten. In diesem Fall wird dann nur der Name des gespeicherten Modells in den aufrufenden Dialog übernommen.

Exportieren/Importieren: Die gespeicherten Modelle können zum Einsatz auf anderen Installationen in eine Datei exportiert bzw. Modelle aus einer Datei (Endung .IKL) importiert werden.

→ Dialog „Neuro-Klassifizierung einstellen“, Seite 27

→ Dialog „Klassifizierungs-Modell speichern“, Seite 24

Dialog „Netzarchitektur erzeugen“

Dient zum Erzeugen der Beschreibung einer Netz-Architektur.

- Dieser Dialog wird geöffnet, wenn Sie im Dialog „Neuronale Prognose einstellen“ in der Registerkarte „Architektur“ die Schaltfläche **Erzeugen...** klicken.

Der Dialog nimmt die Arbeit ab, die Architektur eines Neuronalen Indikators von Hand zu erzeugen (obwohl auch dies möglich ist).

Die Netz-Architektur bietet natürlich ein weites Feld für Experimente. Man sollte aber auch nicht erwarten, dass man durch eine ausgefeilte oder besonders mächtige Architektur mit jedem Input ein sinnvolles Modell erhält. Die Auswahl von sinnvollen Inputs ist also die Hauptaufgabe bei der Konstruktion eines neuen Modells. Wichtig ist auch die Skalierung der Inputdaten mit Hilfe der Input-Analyse (→ Dialog „Inputs analysieren“, Seite 23).

Die Anzahl der Inputs ergibt sich aus der Input-Definition. Der Output beinhaltet immer eine Unit. Das Neuronale Netz kann auch ohne Hidden-Units arbeiten. In diesem Fall wählen Sie als Anzahl der Hiddenschichten den Wert 0.

Tip: Wenn Sie sich nicht weiter mit der Architektur beschäftigen möchten, arbeiten Sie standardmäßig am besten mit einem Netz ohne Hiddenschicht oder mit nur einer Hiddenschicht mit 3-5 Units.

Netzarchitektur erzeugen

Definition der Hidden-Units

Anzahl Hiddenschichten: 3

Units	Biastyp
7	Multiplikation
3	Schwellenwert
4	Optimieren!

Optimierung der Verbindungen (9 Inputs definiert)

☒ Input-Verbindungen optimierbar

☒ Anzahl Input-Verbindungen festlegen: 5

☐ Hidden-Verbindungen optimierbar

☐ Anzahl Hidden-Verbindungen begrenzen: 3

☐ Schichtmodell ignorieren

Hilfe, Abbrechen, OK

Definition der Hidden-Units

Zur Definition der Hidden-Units wählen Sie zunächst die **Anzahl Hiddenschichten** (0-32). Klicken Sie hierzu die Scrollpfeile oder verwenden Sie die Pfeiltasten der Tastatur.

Für jede Hiddenschicht können Sie dann in der Liste die Anzahl der Units angeben (2-100) sowie den gewünschten Biastyp wählen. Klicken Sie hierzu in die Spalte „Biastyp“ der gewünschten Zeile und wählen Sie in der Dropdownliste den gewünschten Typ aus.

Folgende Biastypen stehen zur Auswahl:

- **Addieren (Standard):** Das Biasgewicht wird zu den Inputs der Unit aufaddiert.
- **Multiplikation:** Das Biasgewicht wird mit der Summe der Inputs der Unit multipliziert (Möglichkeit zum Umkehren des Vorzeichens).
- **Schwellenwert:** Das Biasgewicht fungiert als Schwellenwert. Die Unit liefert nur einen Output, wenn die Summe der Inputs der Unit den Schwellenwert erreicht.
- **Optimieren:** Der Biastyp wird aus den oben genannten Typen per Optimierung ermittelt.

Optimierung der Verbindungen

Standardmäßig werden die Units des Neuronalen Netzes von Schicht zu Schicht vollständig verbunden, und die Optimierung wirkt sich lediglich auf die Gewichte der Verbindungen aus. Zusätzlich ist es aber auch möglich optimieren zu lassen, zu welchen anderen Units eine Unit überhaupt Verbindungen knüpft. Damit kann insbesondere auch die Anzahl der Verbindungen einer Unit reduziert werden.

Auf der anderen Seite ist aber auch zu beachten, dass die Optimierung der Verbindungen die Optimierung der Gewichte erschwert: Denn durch eine Änderung einer Verbindung verliert die bisherige Optimierung des Verbindungsgewichts seine Geltung. Die Optimierung der Verbindungsgewichte stellt ohnehin schon eine hohe Anforderung an die Genetischen Algorithmen – durch die Optimierung von Verbindungen und Biastypen wird diese noch deutlich höher.

Hinweis: Während die Optimierung der Gewichte über Werte-Optimierungsvariablen erfolgt, wird die Optimierung der Verbindungen durch Listenvariablen gestaltet.

Input-Verbindungen optimierbar: Wenn aktiviert, werden die Verbindungen von der ersten Hiddenschicht zu den Inputs optimierbar gestaltet. In diesem Fall kann während der Optimierung trainiert werden, zu welchen Inputs die Hidden-Units der ersten Schicht im einzelnen eine Verbindung knüpfen.

Anzahl Input-Verbindungen festlegen: Hier kann angegeben werden, wie viele Verbindungen eine Hidden-Unit zum Input verwendet. Ist die Option nicht aktiviert, entspricht die Anzahl der Anzahl der Inputs. In der Regel wird es sinnvoll sein, die Anzahl zu reduzieren. Hat ein Neuronales Netz zum Beispiel 8 Inputs und die erste Hiddenschicht 5 Units, könnten Sie hier den Wert 4 einstellen. Insgesamt hätten die 5 Hidden-Units dann $5 \cdot 4 = 20$ Verbindungen zu den 8 Inputs, wobei die Optimierung festlegt, zu welchen Inputs die Verbindungen jeweils verlaufen.

Hidden-Verbindungen optimierbar: Auch die Verbindungen von nachfolgenden (der ersten Hiddenschicht folgenden) Hiddenschichten zu darüber liegenden Hidden-Units können optimierbar gestaltet werden (siehe vorstehende Erklärungen zu den Input-Verbindungen). Auch hier kann die Maximalanzahl der Verbindungen pro Unit auf Wunsch festgelegt werden.

Wenn **Schichtmodell ignorieren** nicht aktiv ist, können die Hidden-Units jeweils nur zu jeder Unit der darüber liegenden Schicht Verbindungen knüpfen. Ist **Schichtmodell ignorieren** aktiv, können Verbindungen zu allen vorhandenen davor gelagerten Hidden-Units geknüpft werden (also auch zur eigenen und zu weiter entfernten Schichten).

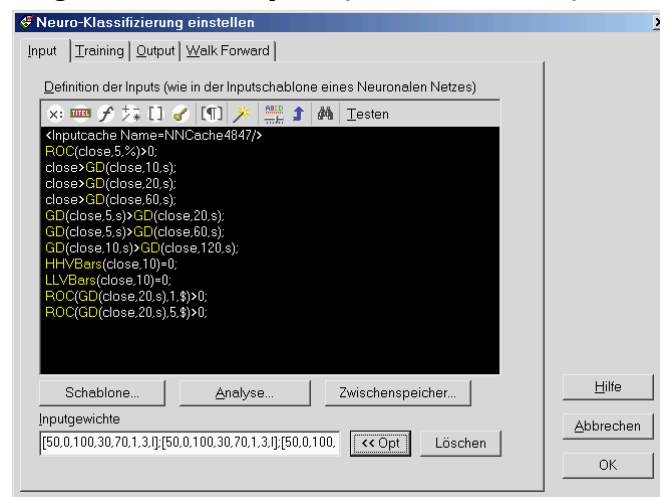
Dialog „Neuro-Klassifizierung einstellen“

Dient zum bequemen Einstellen einer Neuronalen Klassifizierung.

- Der Dialog erscheint, wenn Sie den Indikator NeuroClassify bzw. NeuroClassifyWF neu einfügen (Assistent) oder bearbeiten.

Die Einstellung erfolgt in drei, in der Walk-Forward-Version des Indikators in vier Registerkarten bzw. Assistenten-Schritten.

Registerkarte „Input“ (Schritt 1 von 4)



Geben Sie die gewünschten Inputs für die Klassifizierung in den Formeleditor ein (vergleichbar mit einer Inputschablone eines Neuronalen Netzes). Jeder einzelne Input besteht aus einer Berechnung und endet mit einem Strichpunkt. Zusammen bilden die Inputs das Muster, das der Indikator klassifiziert und zur Prognose heranzieht. Die Skalierung der Daten erfolgt automatisch beim Training des Indikators (im Unterschied zur „Neuronalen Prognose“, bei der die Skalierung innerhalb der Inputberechnung erfolgen muss).

Schablone: Hier können Sie eine vorhandene Inputschablone auswählen und als Inputdefinition einfügen.

Analyse: Klicken Sie hier, um eine Analyse der Inputs durchführen zu lassen (→ Dialog „Inputs analysieren“, Seite 23).

Zwischenspeicher: Die Inputdaten können bei der Optimierung und im Robustheitstest zwischengespeichert werden. Das bedeutet, dass die Inputdaten nur einmal pro Generation bzw. Test berechnet werden müssen. Dies setzt aber voraus, dass die Inputberechnungen selbst **nicht optimiert** werden. Verwenden Sie die Schaltfläche **Zwischenspeicher...**, um auf einfache Weise eine korrekte Einstellung vorzunehmen.

Falls Sie die Einstellung für den Zwischenspeicher von Hand eingeben möchten: Die Aktivierung des Input-Zwischenspeichers erfolgt durch Zusatz des Tags `<inputcache name=xy/>` innerhalb der Inputdefinitionen, wobei xy der Name des Zwischenspeichers ist. Werden mehrere Neuronale Indikatoren innerhalb eines Handelssystems verwendet, muss jeder Zwischenspeicher anders benannt werden.

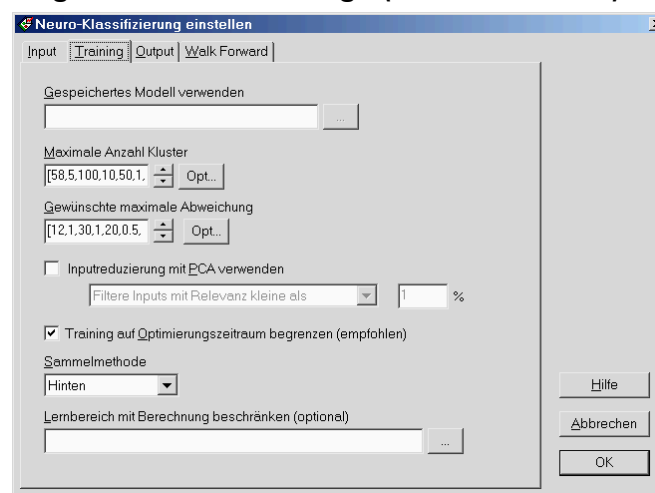
Inputgewichte: Optional können hier Werte zur Gewichtung der einzelnen Inputs angegeben werden. Je nach relativem Gewicht des Inputs fließt der Input mehr oder weniger in das Abstandsmaß ein. Falls Gewichte angegeben werden, muss die Anzahl der Gewichte genau mit der Anzahl der Inputs übereinstimmen.

Wenn Sie die Gewichtung der Inputs optimieren lassen möchten, klicken Sie auf **<< Opt.**

Investox fügt dann die zu den Inputs passende Anzahl Optimierungsvariablen ein.

Mit **Löschen** können die Gewichtungsangaben schnell entfernt werden. Falls keine Gewichte angegeben werden, werden alle Inputs gleich gewichtet.

Registerkarte „Training“ (Schritt 2 von 4)



Gespeichertes Modell verwenden: Hier kann ein bereits gespeichertes trainiertes Modell angegeben werden (nicht verfügbar für die WF-Version des Indikators, → Dialog „Klassifizierungs-Modell wählen, Seite 25). Der Indikator verwendet dann die Muster des gespeicherten Modells, anstatt diese neu zu lernen. Dies ist zum Beispiel dann sinnvoll

- wenn das zuvor trainierte Modell mit anderen Titeln oder anderen Datenbereichen verwendet werden soll,
- wenn ein trainiertes Modell eingesetzt, dabei aber nicht jedes Mal trainiert werden soll.

Tip: So speichern Sie ein Modell ab: Setzen Sie in den Handelssystem-Regeln im Formeleditor den Cursor auf den Indikator NNClassify und klicken Sie dann mit der rechten Maustaste auf den Indikator, um das Kontextmenü zu öffnen. Wählen Sie im Kontextmenü den Befehl **Klassifizierungs-Modell speichern**.

Maximale Anzahl Cluster / Gewünschte maximale Abweichung: Diese beiden Parameter legen die Mächtigkeit des Modells fest. Mit „Maximale Anzahl Cluster“ bestimmen Sie, wie viele Cluster das Modell höchstens bilden darf. Ob diese maximale Anzahl tatsächlich verwendet wird, hängt von der Einstellung der maximalen Abweichung ab. Ist diese groß

genug, genügen eventuell auch weniger Cluster, um alle Muster im Trainingsbereich mit dem angegebenen maximalen Abstand abzubilden.

Der Abstand wird proportional zur Spanne der Werte der einzelnen Inputs gemessen. Es erfolgt also eine Skalierung der Daten (im Optimierungszeitraum).

Der Ablauf des Trainings ist wie folgt:

1. Bilde so wenige Cluster wie möglich, um alle Muster im Trainingszeitraum mit dem eingestellten Maximalabstand abbilden zu können.
2. Falls dies auch mit der angegebenen Maximalanzahl von Clustern nicht möglich ist, wird der zulässige maximale Abstand soweit wie nötig erhöht. Die Begrenzung der Clusteranzahl hat also Priorität.

Der Trainingsvorgang verläuft grundsätzlich schneller, wenn die maximale Abweichung nicht zu eng gelegt wird – vor allem dann, wenn die maximale Anzahl der Cluster eng begrenzt wird. Die maximale Abweichung wirkt sich aber auch auf die Bildung des Outputs aus, wenn als Ausgabe „Gewichtet“ oder „Treffer“ gewählt wird (siehe unten, Registerkarte „Output“).

Inputreduzierung mit PCA verwenden: Mit dieser Option kann optional eine Reduzierung der Inputdimension mit Hilfe der Principal Component Analysis vorgenommen werden (siehe Erläuterungen zur PCA im Dialog „Inputs analysieren“ (→ Seite 23).

Training auf Optimierungszeitraum begrenzen: Wenn aktiv, werden für das Training automatisch nur Muster aus dem Optimierungszeitraum des Handelssystems verwendet. Diese Einstellung sollte standardmäßig eingeschaltet sein, da sonst die Fähigkeit zur Generalisierung im Kontrollzeitraum nicht getestet werden kann (Option nicht relevant für die Walk-Forward-Version des Indikators).

Es kann aber auch Fälle geben, in denen es sinnvoll ist, die Begrenzung auf den Optimierungszeitraum abzuschalten. Zum Beispiel dann, wenn ausschließlich über eine Lernbeschränkung festgelegt werden soll, welche Daten für das Training verwendet werden sollen (siehe unten).

Hinweis: Ist kein Optimierungszeitraum vorhanden (Chart ohne Handelssystem), werden alle Daten verwendet.

Sammelmethode: Hier kann die Reihenfolge beziehungsweise der Startpunkt der Mustersuche beim Training eingestellt werden. Dies wirkt sich am ehesten bei einer relativ starken Begrenzung der maximalen Clusteranzahl aus. In diesem Fall wird die Bildung der Cluster durch die Reihenfolge mehr oder weniger stark beeinflusst.

- **Gemischt:** Die Muster werden nach und nach aus gemischten Samples entnommen, so dass die verschiedenen Zeitbereiche ähnlich behandelt werden.
- **Vorne:** Die Untersuchung der Muster beginnt mit den älteren Daten.
- **Hinten:** Die Untersuchung der Muster beginnt mit den jüngeren Daten.

Lernbereich mit Berechnung beschränken: Hier können durch Angabe einer entsprechenden Berechnung die für das Training verwendeten Muster beschränkt werden (entspricht der Einstellung bei Standard-NNs). Wenn eine Berechnung angegeben wird, werden zur Bildung von Muster-Clustern nur Muster der Perioden verwendet, an denen die Berechnung einen Wert ≤ 0 liefert.

Registerkarte „Output“ (Schritt 3 von 4):

Outputmuster: Das Outputmuster stellt die gewünschte Prognose dar. Sie können aus der Liste eine vorgegebene Definition wählen oder eine eigene Berechnung direkt eingeben. In den meisten Fällen empfiehlt es sich, aus der Liste den ersten Eintrag „Kursprognose (einstellen)...“ zu wählen. Sie können dann die gewünschte Prognose (Trendstärke und Horizont) direkt einstellen.

Das Outputmuster definiert die Bewertung der Cluster. Der Output eines Clusters wird als Mittelwert aller Outputmuster-Werte ermittelt, die dem jeweiligen Cluster-Muster zugeordnet wurden.

Hinweis: Bei der Walk-Forward-Version ist ein geeigneter Prognoseabstand einzustellen, wenn der Output mehr als eine Periode in die Zukunft blickt. Siehe dazu die Erläuterungen zur Einstellung **Prognoseabstand** in der Registerkarte „Walk-Forward“.

Ausgabe: Legen Sie hier fest, wie die Ausgabe ermittelt wird:

- **Bestes:** Das Cluster-Muster mit dem geringsten Abstand wird verwendet, auch dann, wenn es außerhalb der maximalen Abweichung im Trainingszeitraum liegt.
- **Treffer:** Das Cluster-Muster mit dem geringsten Abstand wird verwendet, aber nur, wenn es innerhalb der maximalen Abweichung im Trainingszeitraum liegt. Ansonsten wird 0 ausgegeben. Es lässt sich zusätzlich eine Prognosetoleranz definieren (s.u.).
- **Gewichten (Standard):** Beim Gewichten werden alle passenden Cluster-Muster innerhalb der maximalen Abweichung (unter Berücksichtigung der Prognosetoleranz) verwendet und im Verhältnis zu ihrer Abweichung zum Inputmuster gewichtet. Besser passende Cluster-Muster werden also stärker gewichtet als stärker abweichende Cluster.

Prognosetoleranz: Gibt an, welche zusätzliche Abweichung (in %) eines Musters gegenüber der maximalen Abweichung zu den Muster-Clustern für eine Ausgabe zugelassen wird (Standard = 0, nicht relevant für Ausgabe = Bestes). Der Sinn der Einstellung besteht darin, eine Ausgabe des Indikators zu ermöglichen, obwohl keines der Cluster-Muster innerhalb der ermittelten maximalen Abweichung liegt (bei Ausgabe = Treffer oder Gewichten). Zu beachten ist, dass beim Training die maximale Abweichung aufgrund der Begrenzung der Cluster-Anzahl unter Umständen erhöht wird. Die Toleranz bezieht sich auf die so ermittelte maximale Abweichung.

Muster-Mindesttreffer: Hiermit kann festgelegt werden, dass nur solche Muster für die Ausgabe verwendet werden, die mehrmals im Trainingsbereich angetroffen wurden (Standard = 1). Höhere Werte sprechen für eine höhere Signifikanz des verwendeten Musters, filtern aber gegebenenfalls auch interessante Einzelmuster heraus. Die Einstellung ist auch in Relation zur maximalen Clusteranzahl zu sehen – je geringer diese ist, desto höher wird das durchschnittliche Mindestmuster-Vorkommen sein.

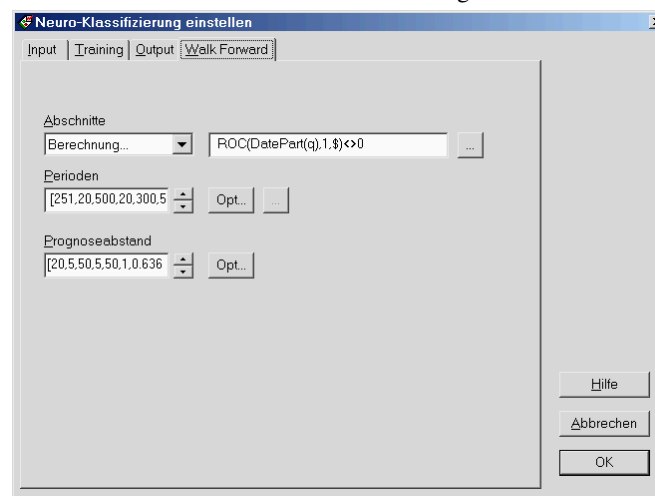
Outputmodus: Legen Sie hier fest, welche Art von Output der Indikator liefern soll.

- **Output (Prognose):** Dies ist die Standardeinstellung. Der Indikator liefert den berechneten Outputwert gemäß dem Outputmuster.

- **Outputnummer:** Geliefert wird die Nummer des Treffer-Clusters. Außer zu Diagnosezwecken kann diese Angabe für ein unüberwachtes, vom Outputmuster unabhängiges Modell verwendet werden. Dabei werden Handelssignale abhängig von bestimmten Cluster-Nummern erzeugt.
→ Projekt „NNClassify_Clusterauswahl_EOD_FDAXBUND“, Seite 39
- **Anzahl Treffer:** Liefert die Anzahl der Treffer für die jeweilige Periode – wie viele Cluster also auf das Muster der Periode passen würden (mit Bezug auf die angegebene maximale Abweichung). Bei der Ausgabeart „Gewichten“ fließen alle Treffer in die Ausgabe ein.
- **Anzahl Cluster:** Zeigt an, wie viele Cluster das trainierte Modell besitzt. Interessant ist diese Angabe vor allem bei der Walk-Forward-Version des Indikators. Sie kann hier zeigen, welche unterschiedliche Anzahl Cluster in den Abschnitten jeweils für das Training erforderlich sind.

Registerkarte „Walk Forward“ (Schritt 4 von 4)

Für die WF-Version des Indikators stehen folgende weitere Einstellungen zur Verfügung:



Abschnitte: Diese Einstellung legt fest, an welchen Punkten jeweils eine Prognose mit einem neuen Modell erfolgt, das mit vorangehenden Daten trainiert wurde. Es können wahlweise verschiedene Zeitabschnitte (Stunden, Tage, Wochen, Monate, Quartale, Jahre) oder aber eine eigene Berechnung hierfür angegeben werden. Für den ersten gültigen Abschnitt muss mindestens die im folgenden angegebene Anzahl Perioden für das Training zur Verfügung stehen.

Lernperioden: Anzahl der Perioden, die für das Training des jeweiligen Abschnitts verwendet werden. Wenn Sie hier zum Beispiel 200 eingeben und für jeden Monat eine neue Prognose erstellt wird, dann wird das jeweilige Modell jeweils mit den 200 Perioden vor dem Monatsende trainiert.

Prognoseabstand: Hier kann ein Abstand der Prognose zum Trainingsbereich angegeben werden. Dies ist dann sinnvoll oder erforderlich, wenn die Berechnung des Outputmusters Daten verwendet, die mehr als eine Periode in die Zukunft blicken (zum Beispiel Ref() mit einer Einstellung >1). Wenn Sie also zum Beispiel als Outputmuster REF(ROC(OPEN, 5), 5) angeben, sollte der Prognoseabstand mindestens 4 betragen. Dann beginnt die Prognose jeweils erst vier Perioden nach dem Ende des letzten Trainingsabschnitts.

Hinweise zur Berechnungsdauer

Die Berechnungsdauer für die Walk-Forward-Klassifizierung kann abhängig von den gewählten Einstellungen sehr unterschiedlich sein. Relevant hierfür sind die Anzahl Cluster, die maximale Abweichung sowie die Anzahl der Abschnitte und die Anzahl Lernperioden pro Abschnitt. Dauert die Berechnung des Indikators länger als drei Sekunden, erfolgt eine Fortschrittsanzeige in der Statusleiste des Programmfensters. Die Berechnung kann dann durch Klicken auf die Fortschrittsanzeige abgebrochen werden.

Dialog „Neuronale Prognose einstellen“

Dient zum Einstellen des NeuroProg-Indikators.

- Der Dialog erscheint, wenn Sie den Indikator NeuroProg neu einfügen (Assistent) oder bearbeiten.

Der Einstelldialog bietet eine bequeme Möglichkeit, ein Neuronales Netz mit dem NeuroProg()-Indikator zu erstellen oder zu bearbeiten.

Die Einstellung erfolgt in zwei Registerkarten bzw. Assistenten-Schritten:

Input (Schritt 1 von 1)

Hier können die Inputs für die Neuronale Prognose eingegeben und bearbeitet werden. Die Angabe erfolgt wie in einer Inputschablone eines Neuronalen Netzes:

- Alle Definitionen (Calc/Const) stehen als Block vor dem ersten Input.
- Jeder Input endet mit einem Strichpunkt.

Schablone: Hier können Sie eine vorhandene Inputschablone auswählen und als Inputdefinition einfügen.

Hinweis: Bei der Neuronalen Prognose erfolgt **keine** interne Skalierung der Inputdaten (anders als bei NeuroClassify). Eine Skalierung muss, falls nötig, in den Berechnungen selbst vorgenommen werden. Hierbei hilft das Analyse-Werkzeug, das Sie mit der Schaltfläche **Analyse...** starten können.

Analyse: Klicken Sie hier, um eine Analyse der Inputs durchführen zu lassen und gegebenenfalls eine Skalierung der Daten einzufügen
(→ Dialog „Inputs analysieren“, Seite 23).

Zwischenspeicher: Die Inputdaten können bei der Optimierung und im Robustheitstest zwischengespeichert werden. Das bedeutet, dass die Inputdaten nur einmal pro Generation bzw. Test berechnet werden müssen. Dies setzt aber voraus, dass die Inputberechnungen selbst **nicht optimiert** werden. Verwenden Sie die Schaltfläche **Zwischenspeicher**, um auf einfache Weise eine korrekte Einstellung vorzunehmen.

Falls Sie die Einstellung für den Zwischenspeicher von Hand eingeben möchten: Die Aktivierung des Input-Zwischenspeichers erfolgt durch Zusatz des Tags `<inputcache name=xy/>` innerhalb der Inputdefinitionen, wobei xy der Name des Zwischenspeichers ist. Werden mehrere Neuronale Prognosen innerhalb eines Handelssystems verwendet, sollte jeder verwendete Zwischenspeicher anders benannt werden.

Architektur (Schritt 2 von 2)

Erzeugen...: Öffnet das Werkzeug zum Erzeugen einer vollständigen Beschreibung der Architektur des Neuronalen Netzes. Die Architektur umfasst die Definition der Hidden-Units und deren Verbindungsgewichte.

→ Dialog „Netzarchitektur erzeugen“, Seite 25

Dialog „Walk-Forward-Optimierung vorbereiten“

Dient zur Vorbereitung der Walk-Forward-Optimierung eines Handelssystems.

- Wird mit Befehl **Walk-Forward-Optimierung vorbereiten** im Menü „Handelssystem“ aufgerufen (→ Einleitung zur Walk-Forward-Optimierung, Seite 21).

Mit Hilfe des Dialogs können die für eine Walk-Forward-Optimierung benötigten Handelssysteme erzeugt werden. Dazu geben Sie die gewünschten Zeitabschnitte manuell an oder Sie lassen diese automatisch ermitteln. Für jeden Zeitabschnitt wird nach Klicken auf **OK** ein Handelssystem erzeugt, das dann mit diesem Zeitabschnitt optimiert werden kann. Die aktuellen Einstellungen des Dialogs werden im Handelssystem gespeichert.

Walk-Forward-Optimierung vorbereiten

Einstellungen
System 'Parabolik', Titel 'SAP' FWB', Gesamtzeitraum: 12.05.1988 - 17.07.2015

Basisname für die zu erzeugenden Handelssysteme

Neues Projekt für die Handelssysteme anlegen ☒

Walk-Forward-Abschnitte

	Optimierung-Start	Optimierung-Ende	Kontroll-Start	Kontroll-Ende
1	01.01.1990	31.12.1995	01.01.1996	31.12.1996
2	01.01.1991	30.12.1996	02.01.1997	02.01.1998
3	03.01.1992	30.12.1997	02.01.1998	04.01.1999
4	01.01.1993	30.12.1998	04.01.1999	03.01.2000
5	01.01.1994	30.12.1999	03.01.2000	02.01.2001
6	01.01.1995	29.12.2000	02.01.2001	02.01.2002
7	29.12.1995	28.12.2001	02.01.2002	30.12.2002
8	28.12.1996	27.12.2002	30.12.2002	30.12.2003
9	28.12.1997	29.12.2003	30.12.2003	29.12.2004
10	28.12.1998	28.12.2004	29.12.2004	29.12.2005
11	28.12.1999	28.12.2005	29.12.2005	29.12.2006
12	31.12.2000	28.12.2006	29.12.2006	02.01.2008
13	28.12.2001	28.12.2007	02.01.2008	29.12.2008
14	27.12.2002	23.12.2008	29.12.2008	28.12.2009

Info: Für jeden Abschnitt wird ein Handelssystem erzeugt. Nach der Optimierung aller Handelssysteme kann das Ergebnis im Projekt-Portfoliotest geprüft werden.

Abschnitte ermitteln
Zufügen
Löschen
Hilfe
Abbrechen
OK

Basisname: Die erzeugten Handelssysteme werden durchnummeriert und verwenden den angegebenen Basisnamen als Präfix.

Neues Projekt für Handelssysteme anlegen: Ist diese Option aktiviert, werden die erzeugten Handelssysteme in ein neues Projekt eingefügt, ansonsten in das aktuelle Projekt.

Abschnitte ermitteln: Klicken Sie hier, um die Zeitabschnitte automatisch von Investox ermitteln zu lassen (→ Dialog „Walk-Forward-Abschnitte ermitteln“, Seite 33).

Zufügen: Fügt einen Zeitabschnitt in die Liste hinzu. Der erste Abschnitt verwendet die Einstellungen des Handelssystems für Optimierung und Kontrolle.

Löschen: Löscht den markierten Abschnitt aus der Liste.

Dialog „Walk-Forward-Abschnitte ermitteln“

- Wird im Dialog „Walk-Forward-Optimierung vorbereiten“ mit der Schaltfläche **Abschnitte ermitteln** aufgerufen (→ Seite 32).

Ermittelt für ein Handelssystem Zeitabschnitte zur Walk-Forward-Optimierung.

Sie können mit diesem Dialog Abschnitte zur Walk-Forward-Optimierung wahlweise auf der Basis von Zeitabschnitten, einer Anzahl von Perioden oder sogar aufgrund selbstdefinierter Bedingungen erzeugen lassen.

Nach Klicken auf **OK** werden die Abschnittsinformationen erzeugt und im aufrufenden Dialog „Walk-Forward-Optimierung vorbereiten“ eingetragen.

Im Feld „Informationen“ sehen Sie unter „Ermittelte WF-Abschnitte“, wie viele Abschnitte mit den aktuellen Einstellungen erzeugt werden.

→ Einleitung zur Walk-Forward-Optimierung, Seite 21

Start-Optimierungszeitraum/-Kontrollzeitraum: Geben Sie hier zunächst die gewünschten Zeiträume für Optimierung und Kontrolle für den ersten Walk-Forward-Abschnitt an. Zu Beginn werden hier die Zeitraum-Einstellungen des Handelssystems vorgegeben.

Im Bereich „Einstellungen“ geben Sie an, wie bei der Bildung von Abschnitten verfahren werden soll:

Start des Optimierungszeitraums fixieren: Ist diese Option aktiviert, bleibt der Start des Optimierungszeitraums für alle Abschnitte gleich. Der Optimierungszeitraum wird dadurch für jeden neuen Abschnitt entsprechend größer.

Verschiebe Zeiträume gemäß: Zur Erzeugung der Abschnitte stehen folgende Möglichkeiten zur Auswahl:

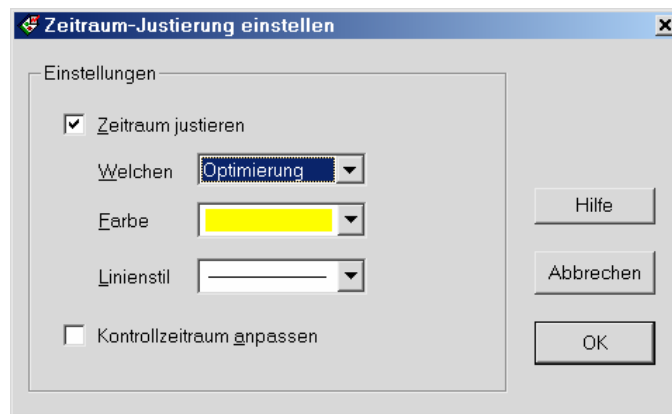
- **Start-Kontrollzeitraum:** Die Abschnitte werden gemäß den Datumsangaben des Start-Kontrollzeitraums verschoben. Diese Einstellung wird standardmäßig für zeitbasierte Komprimierungen verwendet.
- **Perioden des Kontrollzeitraums:** Insbesondere bei nicht zeitbasierten Komprimierungen wie Renko oder Multitick bietet es sich an, die Zeiträume jeweils um eine bestimmte Anzahl Perioden zu verschieben.
- **Berechnung:** Bei dieser Methode wird die im darunterliegenden Feld **Berechnung** angegebene Formel ausgewertet. Ein neuer Abschnitt wird dann für jede Periode erzeugt, in der die Formel einen Wert $\neq 0$ liefert. Damit können benutzerdefinierte Abschnitte zum Beispiel auf Basis von Indikatoren erzeugt werden.

Dialog „Zeitraum-Justierung einstellen“

Ermöglicht die Einstellung der Zeitraum-Justierung im Chart.

- Wird im Kontextmenü der Signalleiste des Charts mit dem Befehl **Justierung einstellen...** geöffnet.

Der Dialog bietet Einstellmöglichkeiten für die Justierung des Zeitraums im Chart.



Welchen: Wählen Sie hier, welcher Zeitraum (Optimierung, Kontrolle oder Gesamt) justiert werden soll.

Farbe: Bestimmt die Farbe der Markierung des Zeitraums in der Signalleiste des Charts sowie der Markierungslinien.

Linienstil: Legt den Stil der Markierungslinien im Chart fest.

Kontrollzeitraum anpassen: Wenn aktiviert (nur wählbar, wenn der Optimierungszeitraum justiert wird), wird nach einer Justierung der Kontrollzeitraum automatisch im Anschluss an den Optimierungszeitraum gelegt (sofern das aktuelle Ende des Kontrollzeitraums überhaupt hinter dem Optimierungszeitraum liegt). Der Kontrollzeitraum beginnt dann in der letzten Periode des Optimierungszeitraums.

Beispiele zur Anwendung

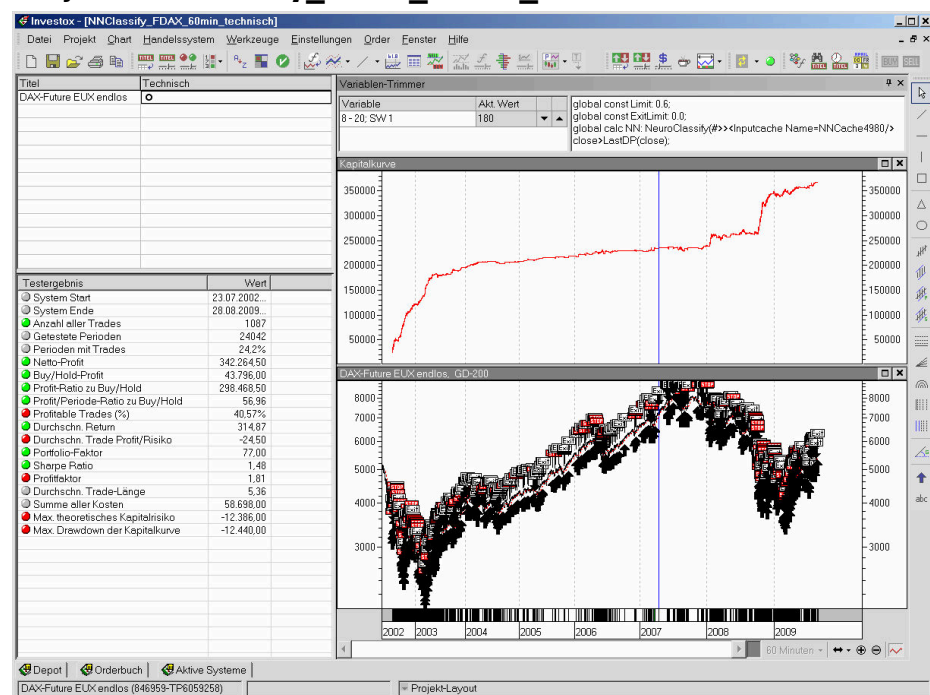
Die im folgenden beschriebenen Projekte zeigen beispielhaft die Anwendung der Neuro-Plus!-Indikatoren.

Hinweis: Die Beispiele geben keine Empfehlungen zum realen Handeln und dienen ausschließlich zu Zwecken der Information über die Arbeitsweise von Neuro Plus!.

Bei den Beispielen zum Indikator „NeuroClassify“ ist zu beachten, dass die gezeigten Ergebnisse auf einem Training mit den abgebildeten Zeiträumen beruhen (Intraday in der Regel mit Daten ab ca. Mitte 2002, End-of-Day ab 1995).

Beispiele zur Neuro-Klassifizierung

Projekt NNClassify_FDAX_60min_technisch



Titel: Dax Future

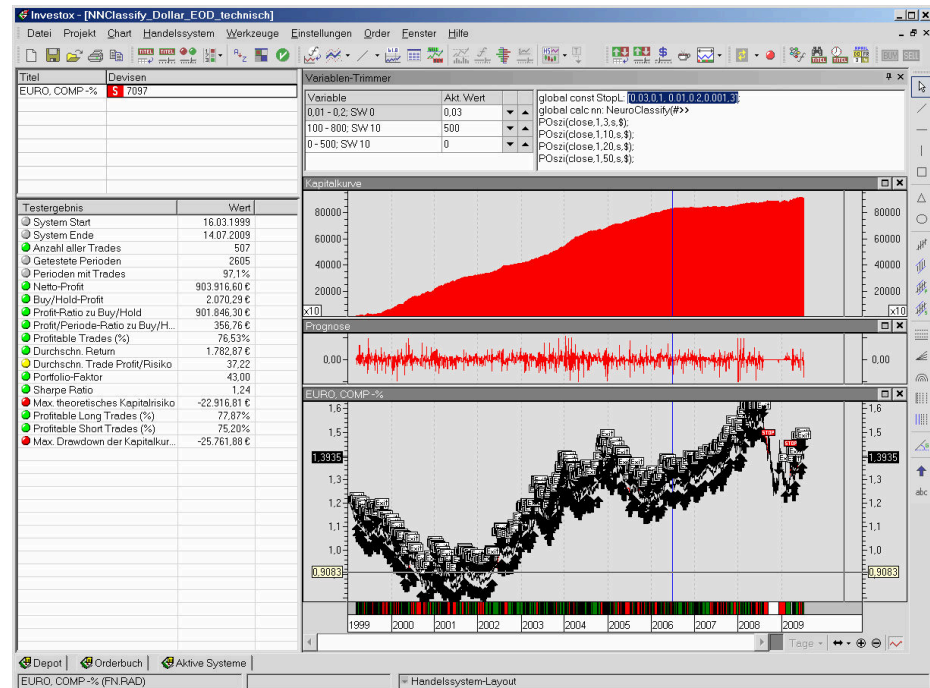
Komprimierung: 60min

Indikator: NeuroClassify

Beschreibung: Die Neuro-Klassifizierung verwendet neben einigen Preissoszillatoren als Input vor allem eine Beschreibung der Kursentwicklung des aktuellen Tages im Vergleich zum vorigen Tag (Vergleich der High-/Low-Kurse). Der Indikator wird zum Einstieg (Enter-Limit +/- 0,6), aber auch zum Ausstieg (Exit-Limit 0) verwendet. Dadurch ist das System nur ca. 25% der Perioden investiert.

Architektur: Die Architektur der Neuro-Klassifizierung zeigt keine Besonderheiten, es werden maximal 180 Cluster verwendet. Der Output ist eine Kursprognose über die nächsten 25 Perioden mit einer Empfindlichkeit von 0,5%.

Projekt „NNClassify_Dollar_EOD_technisch“



Titel: Euro (Euro, Comp-% Daten von Pinnacle)

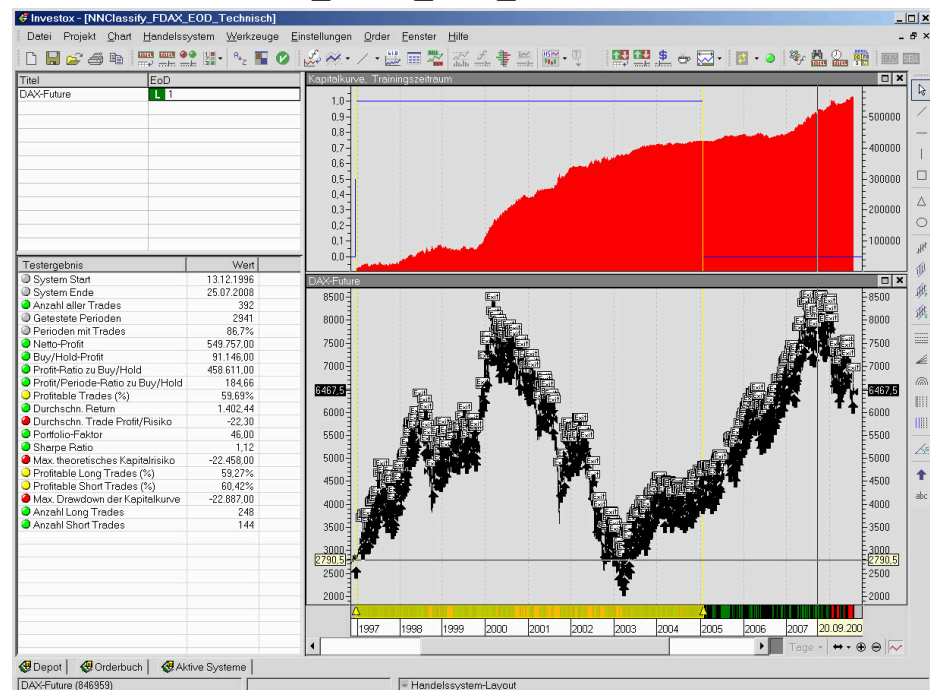
Komprimierung: Täglich

Indikator: NeuroClassify

Beschreibung: Die Neuro-Klassifizierung verwendet als Input vier Preisszillatoren und eine Autokorrelation von Kursen und der 5-Perioden-ROC.

Architektur: Die Architektur ist ohne Besonderheiten, verwendet aber relativ viele Cluster (500). Da die Inputs nicht mit binären Werten (wahr/falsch), sondern mit den Werten der Indikatoren arbeiten, kann eine so hohe Anzahl von Clustern sinnvoll sein.

Projekt „NNClassify_FDAX_EOD_Technisch“



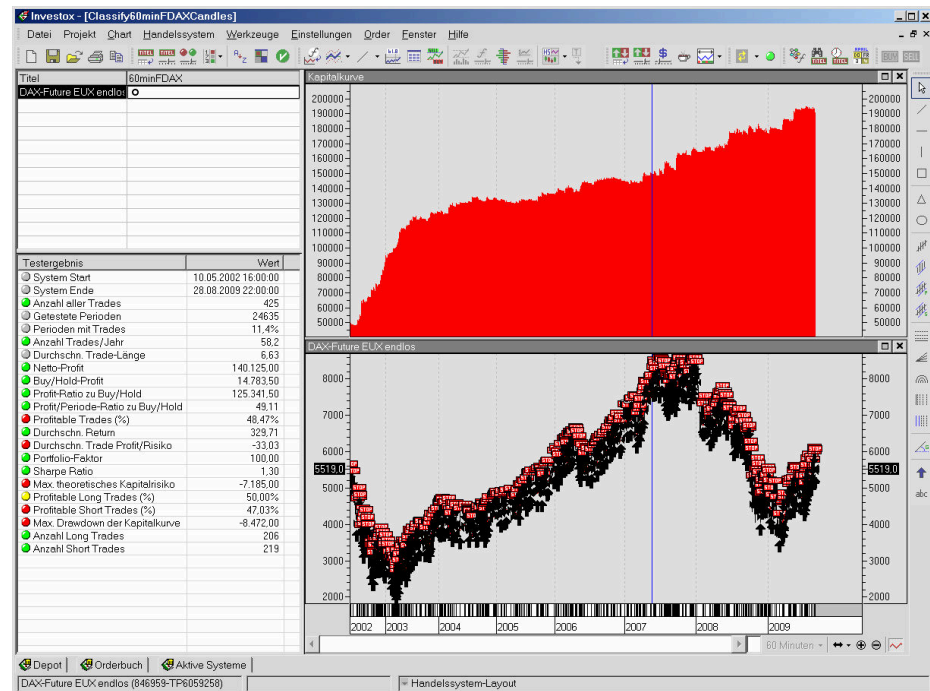
Titel: Dax Future

Komprimierung: Täglich

Indikator: NeuroClassify

Beschreibung: Das System zeigt, wie man mit relativ einfachen Mitteln der Neuro-Klassifizierung ein EoD-Modell erstellen kann. Als Input werden 7 Preisszillatoren geprüft, ob sie über oder unter der 0-Linie liegen. Alle im Optimierungszeitraum vorkommenden Kombinationen werden gespeichert (dafür genügen ca. 80 Cluster). Den Output bildet die Kursprognose über die nächsten 5 Perioden (Mindestschwankung 1%).

Projekt „Classify60minFDAXCandles“



Titel: DAX Future

Komprimierung: 60 Minuten

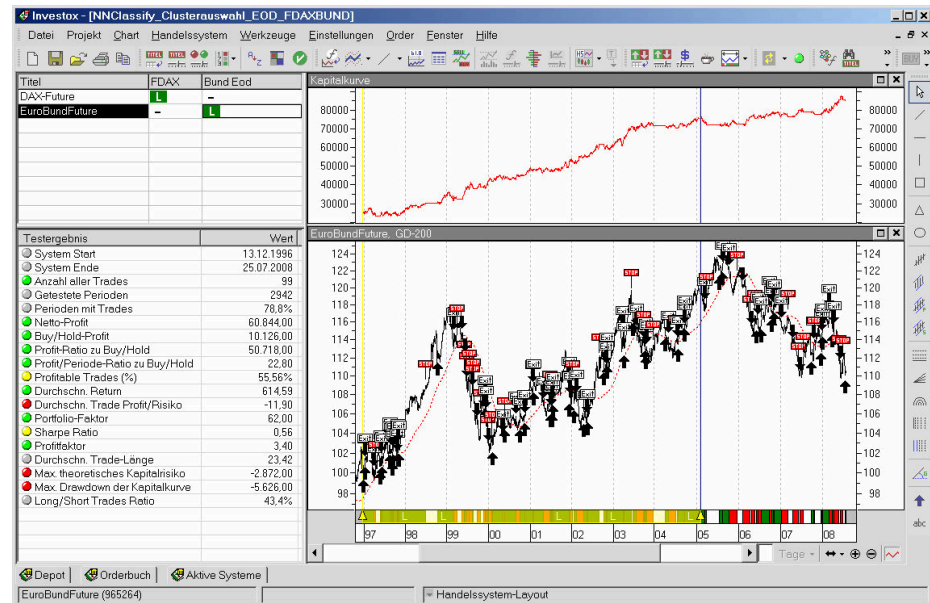
Indikator: NeuroClassify

Beschreibung: Der Indikator verwendet als Input eine Analyse der Beziehung der Kerzen der letzten 3 Perioden. Untersucht wird, ob neue Highs bzw. Lows gebildet werden und ob die High-Low-Spanne größer oder kleiner wird. Das System soll günstige Einstiege liefern und ist im Backtest nur in 11,4% der Perioden investiert. Der Kurs-Trailingstop (berechnet auf Open) ist mit 0,3% relativ eng gesetzt.

Architektur: Die Inputs werden per PCA auf 80% der Informationen gefiltert. Dies reduziert hier die Inputdimension um gut 50%. Die Clusteranzahl ist mit maximal 500 recht hoch. Profitable Kerzenmuster sollen möglichst nicht durch nur entfernt ähnliche, aber nicht profitable Muster überlagert werden.

Hinweis: Wenn ein solches Modell real eingesetzt werden soll, empfiehlt es sich, das Modell zu speichern (→ Dialog „Klassifizierungs-Modell speichern“, Seite 24). Es muss dann nicht bei jeder Aktualisierung trainiert werden. Außerdem kann es so auch mit reduzierten Perioden eingesetzt werden und auf diese Weise Signale effizient erzeugen.

Projekt „NNClassify_Clusterauswahl_EOD_FDAXBUND“



Titel: Dax Future und Bund Future

Komprimierung: Täglich

Indikator: NeuroClassify

Beschreibung: Der Indikator arbeitet in diesem System mit der Ausgabeart „Outputnummer“. Er gibt also nicht das Outputmuster, sondern die Nummer des mit dem Input jeweils ähnlichsten Clusters aus. Das Modell ist damit also komplett „nicht überwacht“, das heißt unabhängig von einer Output-Definition.

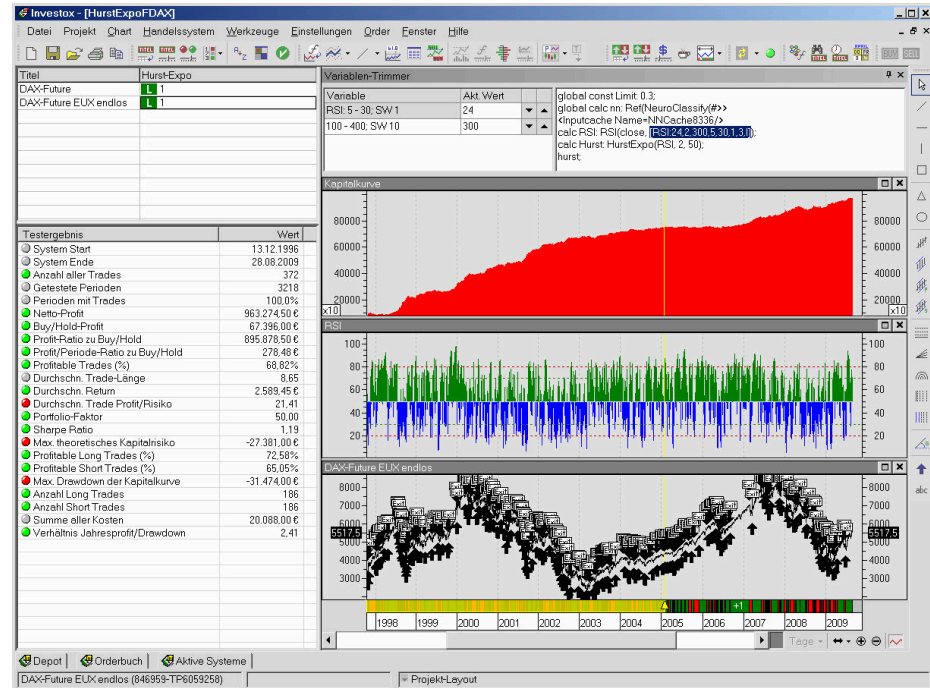
Die Handelsregeln sind entsprechend so aufgebaut, dass untersucht wird, welche Outputnummern einen profitablen Einstieg ermöglichen (kombiniert mit Stops).

Enter Long:

NN1=40 or NN1=27 or NN1=1 or NN1=14 or NN1=37 or NN1=21 or NN1=32

Die passenden Nummern können nur durch „Raten“ bzw. sinnvoller Weise durch eine Optimierung ermittelt werden (siehe Beispielprojekt). Nicht anwendbar ist ein solches Verfahren mit der Walk-Forward-Version des Indikators, da sich hier die Bedeutung der Cluster in jedem Abschnitt ändert.

Hurst-Exponent Projekt



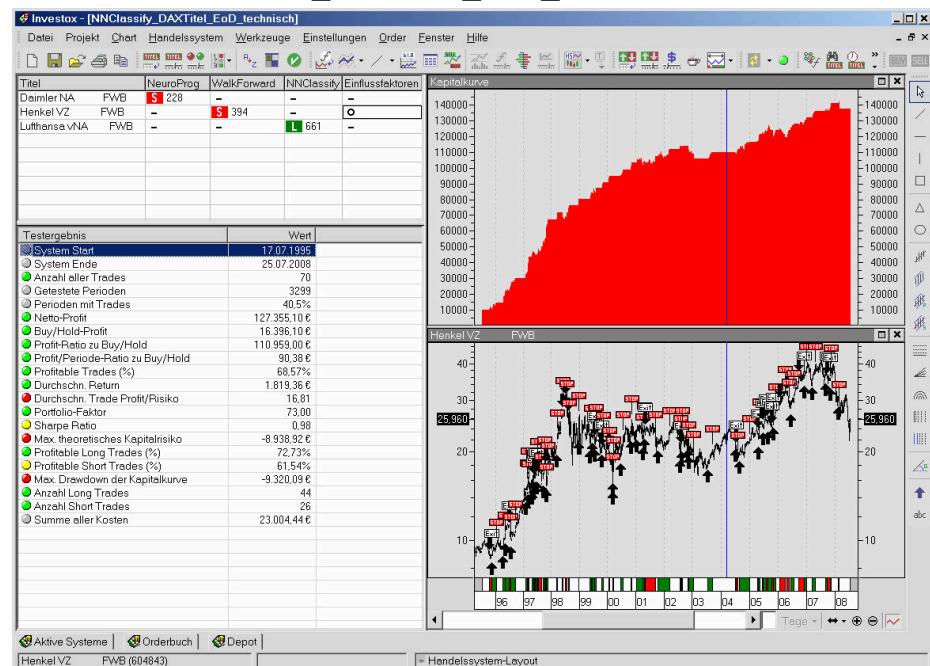
Titel: FDAX

Komprimierung: Täglich

Indikator: NeuroClassify

Beschreibung: Das Handelssystem zeigt, dass der Indikator Hurst-Exponent ein sinnvoller Input in einer Neuronalen Analyse sein kann. In diesem Fall wird er auf einen RSI berechnet. Die Schätzung des Hurst-Exponenten erfolgt dabei in relativ kleinen Zeitfenstern von 2 bis 50 Perioden. Als Input werden der RSI selbst und der Hurst-Exponent verwendet. Neben dem Absolutwert fließt dabei jeweils auch die Steigung der Indikatoren über die letzten 3 beziehungsweise 5 Perioden mit ein (→ Indikator Hurst-Exponent, Seite 16).

Projekt „NNClassify_DAXTitel_EoD_technisch“



Titel: Titel aus dem DAX

Komprimierung: Täglich

Indikator: NeuroClassify und NeuroProg

Beschreibung: Das Projekt zeigt die Anwendung der verschiedenen Neuro-Plus-Indikatoren auf Einzeltitel des DAX.

HS 1: „NeuroProg“: Optimiert für Daimler mit dem Indikator „NeuroPrognose“. Der Indikator verwendet als Input neben einigen Preissoszillatoren eine vergleichende Relative Stärke zum DAX-Index. Das System rechnet mit Long+Short und einem Hebel von 5 (müsste also mit einem Hebel-Zertifikat umgesetzt werden). Die Tradedauer ist auf maximal 20 Tage begrenzt (im Durchschnitt 5,8 Tage). Keine Reinvestition von Gewinnen.

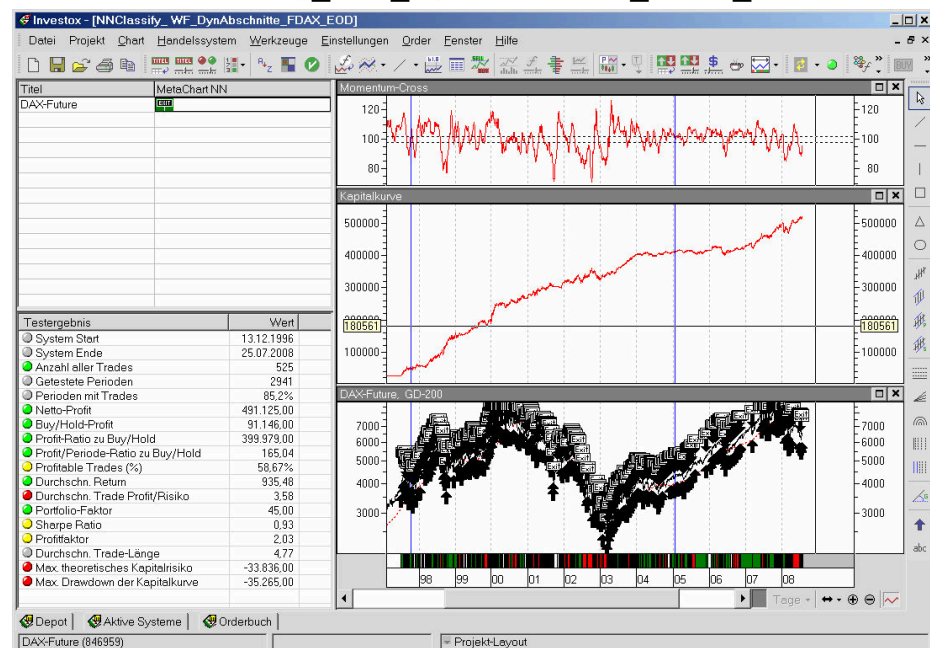
HS 2: WalkForward: Die Neuro-Klassifizierung in Walk-Forward-Version verwendet als Input lediglich drei Momentum-Indikatoren. Die Einstellungen der Klassifizierung wurden optimiert (für Henkel). Die Testbedingungen entsprechen dem HS 1.

HS 3: NNClassify: Diese Neuro-Klassifizierung (ohne WF, optimiert für Lufthansa) verwendet als Input eine breitere Momentum-Analyse, die aber per PCA auf nur zwei Inputs reduziert wird. Der Hebel ist auf 1 gesetzt.

HS 4: Einflussfaktoren: Dieses Handelssystem zeigt, wie man Einflussfaktoren in den Inputs der Neuro-Indikatoren verwenden kann. Direkt in den Inputs können Einflussfaktoren nicht eingesetzt werden. Daher werden die Einflussfaktoren außerhalb des Indikators in den „Definitionen“ als globale Berechnungen angegeben, die man dann im Input verwenden kann (siehe Formelcode in den Handelssystem-Einstellungen).

Beispiele zur Neuro-Klassifizierung mit Walk-Forward

Projekt „NNClassify_WF_DynAbschnitte_FDAX_EOD“



Titel: Dax Future

Komprimierung: Täglich

Indikator: NeuroClassifyWF (Walk Forward)

Beschreibung: Die Neuronale Klassifizierung verwendet als Input mehrere Preissoszillatoren und Kursmuster (Zusatztool Analyse Plus! erforderlich).

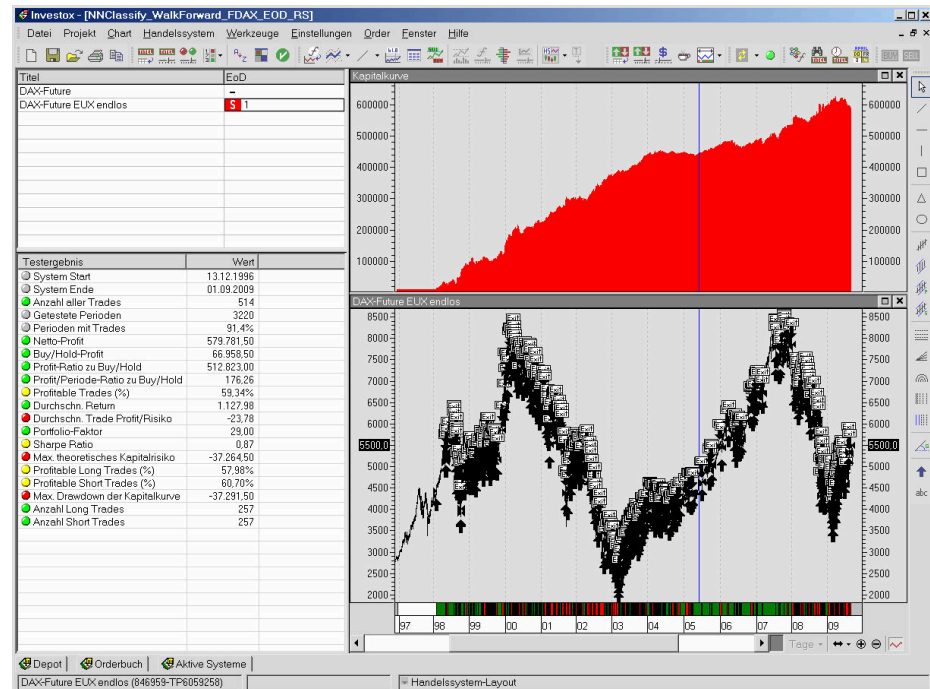
Architektur: Das Besondere an diesem System sind die dynamischen Walk-Forward-Abschnitte. Ein neuer Abschnitt beginnt, wenn das Momentum (30 Tage über 5 Tage geglättet) die 102er-Linie nach oben oder die 98er-Linie nach unten kreuzt.

```
global calc MomCross: (Cross (GD (MOM(close,30),5,s),102,1)=1 or
Cross (GD (MOM(close,30),5,s),98,1)=-1);
<Abschnitte Datenreihe(#MomCross#) />
```

Die Lernperioden umfassen jeweils den davor liegenden Abschnitt (mindestens aber 22 Perioden).

```
<Lernperioden MAX(22,Ref(BarsSince(Datenreihe(#MomCross#),1),-1)) />
```

Projekt „NNClassify_WalkForward_FDAX_EOD_RS“



Titel: Dax Future

Komprimierung: Täglich

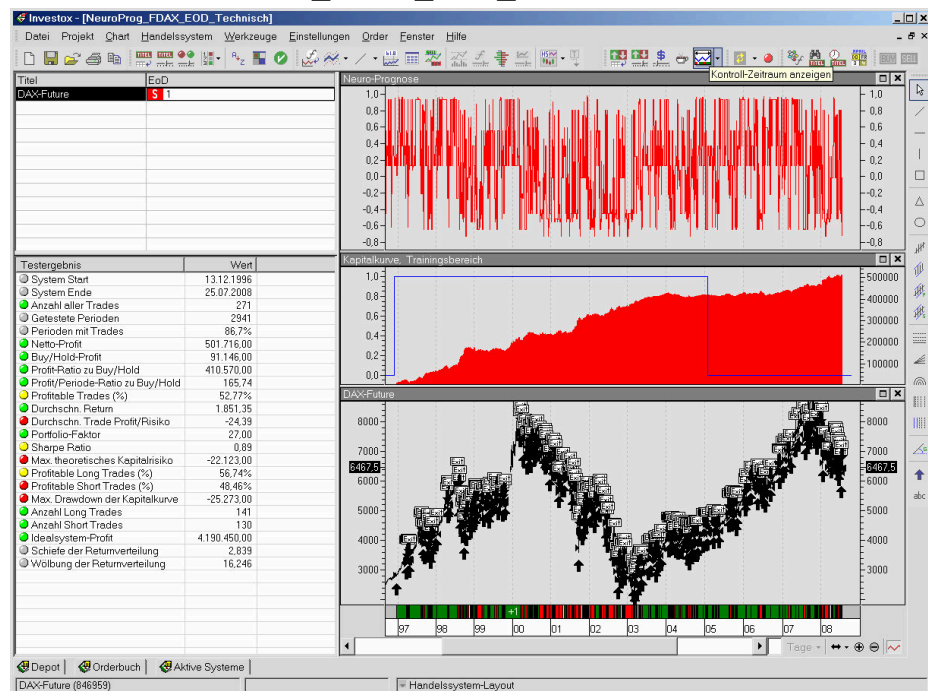
Indikator: NeuroClassifyWF (Walk-Forward)

Beschreibung: Neben einigen technischen Indikatoren wie der Rate of Change verwendet der Indikator als Input auch eine vergleichende Relative Stärke zum Bund Future über 10 Perioden. Prinzipiell können WF-Modelle zur Trendfolge neigen. So zeigt auch dieses einfache Modell am ehesten in stark ausgeprägten Seitwärtsmärkten Schwächen.

Architektur: Die Architektur verwendet maximal 50 Cluster mit einer zulässigen Abweichung der Inputmuster von maximal 12%. Mit jedem Quartal beginnt ein neuer Walk-Forward-Abschnitt, wobei das Modell mit jeweils 250 Perioden trainiert wird (also ca. ein Börsenjahr). Über gut 12 Jahre setzt sich der Output also aus etwa 50 jeweils neu trainierten Modellen zusammen. Dennoch erfolgt die Berechnung ohne größere Verzögerung.

Beispiele zur Neuronalen Prognose

Projekt „NeuroProg_FDAX_EOD_Technisch“



Titel: Dax Future

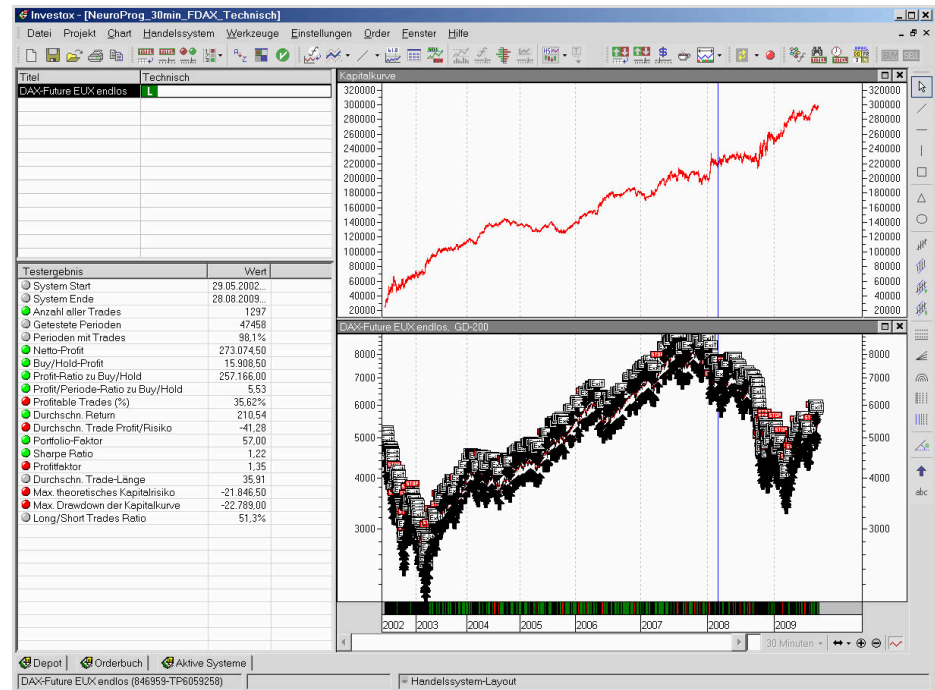
Komprimierung: Täglich

Indikator: NeuroProg

Beschreibung: Die Neuronale Prognose verwendet als Input einige Preissoszillatoren sowie eine Autokorrelation (Korrelation zu früheren Kursen derselben Datenreihe). Die Einstellung der Indikatoren in den Inputs wurde zusammen mit dem Neuronalen Netz optimiert.

Architektur: Das Netz verwendet 8 Hidden-Units mit dem Biastyp „Schwellenwert“.

Projekt „NeuroProg_30min_FDAX_Technisch“



Titel: Dax Future

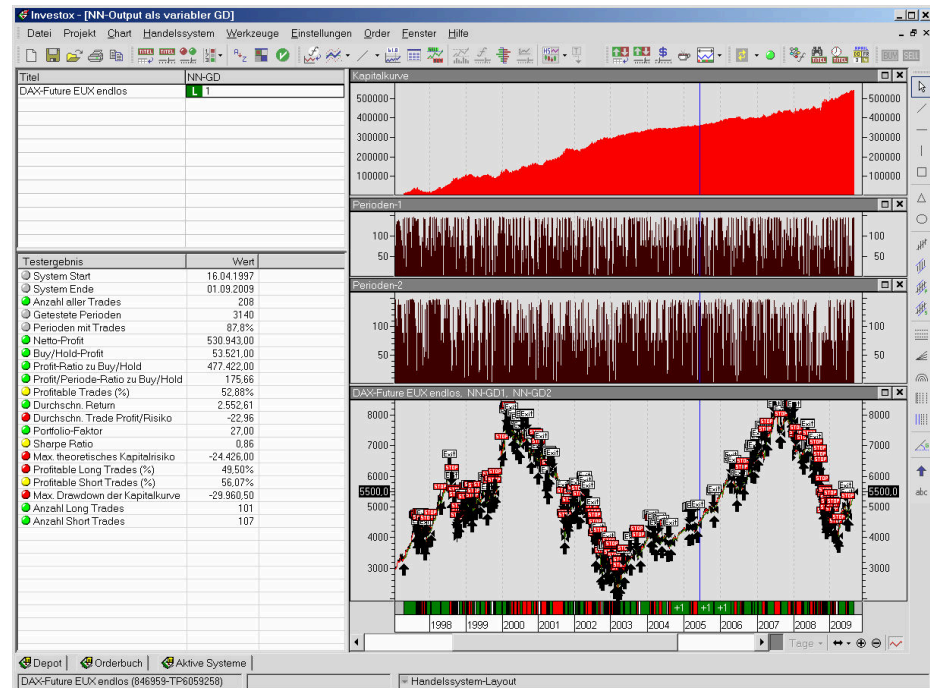
Komprimierung: 30 Minuten

Indikator: NeuroProg

Beschreibung: Die Neuronale Prognose verwendet als Input die prozentuale Rate of Change über fünf Perioden in verschiedenen Zeitfenstern. Mit optimiert wurden jeweils ein Intraday-Stop für die Gewinn- und für die Verlustseite.

Architektur: Das NN besitzt zwei Hidden-Schichten mit 8 und 6 Hidden-Units. Sowohl die Verbindungen zur Hiddenschicht als auch die Verbindungen zwischen den Hiddenschichten wurden optimiert (jeweils drei Verbindungen pro Unit). Es wird also während der Optimierung entschieden, zu welchen drei Inputs bzw. Hidden-Units eine Hidden-Unit eine Verbindung herstellt. Gleichzeitig werden auch noch die dazu gehörenden Verbindungsgewichte optimiert.

Projekt „NN-Output als variabler GD“



Titel: FDAX

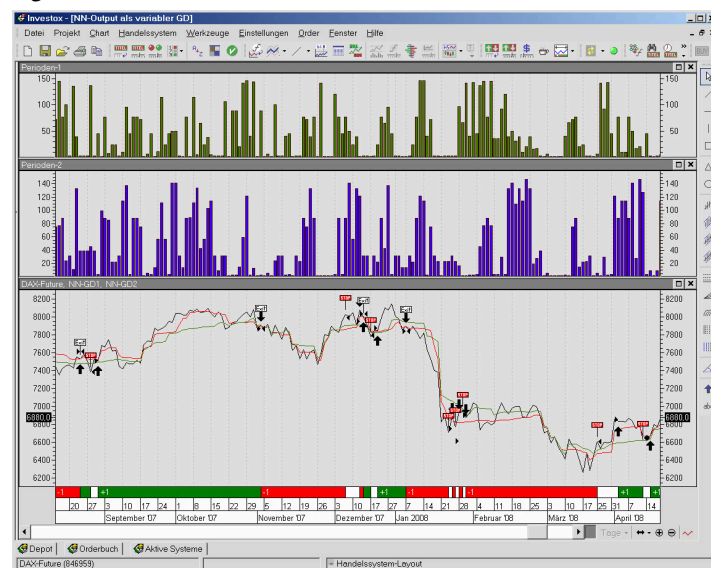
Komprimierung: Täglich

Indikator: NeuroProg

Beschreibung: Der NeuroProg-Indikator ermöglicht auch eher ungewöhnliche Vorgehensweisen beim Training eines Neuronalen Netzes. In diesem Projekt wird der Indikator so in der Optimierung des Handelssystems eingesetzt, dass er als Output einen variablen Gleitenden Durchschnitt (GD) liefert, dessen Gestalt zuvor nicht bekannt ist. Beim Training eines Standard-NN müsste die Outputfunktion dagegen vorgegeben werden und daher bekannt sein.

In diesem Handelssystem werden genauer gesagt zwei variable GDs trainiert und dann wie herkömmliche GDs in einem Signalsystem eingesetzt (weitere Infos siehe Formelcode des Handelssystems).

In der größeren Zoomstufe sieht man in den mittleren Teilcharts die durch die Outputs der NNs gebildeten variablen Periodeneinstellungen und im unteren Teilchart die beiden sich ergebenden GDs mit der Basis:



Index

Abschnitte.....	12	Mindestmuster	12
Abweichung.....	6	Mindesttreffer	30
Aktualisierung	15	Modellgröße	11
Analyse.....	32	Modellname.....	11
Ausgabe.....	12, 30	Netzarchitektur	32
Autokorrelation	37	Netzarchitektur erzeugen	25
Backpropagation.....	12, 13	Neuro-Klassifizierung	5, 8, 11, 27
Berechnungsdauer	11, 31	Neuronale Prognose.....	12, 32
Bias.....	26	Optimierungsgerüst	9
Bias-Typen	13	Optimierungsvariablen dokumentieren..	20
Candlestick	38	Optimierungszeitraum	29
Cluster	8, 10	Optimierungszeitraum justieren... 8, 18, 34	
Clusteranzahl	6, 9	Out-of-Sample	10
Datenfeed-Simulation.....	15	Output definieren.....	30
Genetische Algorithmen.....	10	Outputmodus	30
Gewichte.....	11	Outputmuster	12
Gewichtete Ausgabe	9	Outputnummer.....	39
Gleitender Durchschnitt.....	45	Output-Toleranz	12
Handelssystem.....	9	PCA	12, 23, 29
Hidden-Units	26	Periodenbegrenzung	15
Hurst-Exponent	16, 23, 40	Prognose	12
Inputgewichte	11	Prognoseabschnitt.....	10
Inputs analysieren.....	23	Prognoseabstand	12, 31
Inputs reduzieren	29	Prognosetoleranz	30
Inputschablone.....	28, 32	Sammelmethode	12, 29
Kerzenmuster.....	38	Schwellenwert	26, 43
Klassifizierungs-Modell		Signalberechnung	15
verwenden	28	Skalierung.....	8, 32
Klassifizierungs-Modell		Toleranz.....	12
speichern	15, 24	Trading	15
wählen	25	Training	8
Kontrollzeitraum anpassen	35	Trainingszeitraum.....	12
Kursprognose.....	16	Variablen-Trimmer	8, 17
Lernabschnitte	31	Verbindungsgewicht.....	13, 26
Lernbereich.....	12, 29	Walk-Forward	8, 9, 10, 12, 15, 31, 41
Lernperioden	12, 31	Walk-Forward-Optimierung	21, 32
Maximale Abweichung.....	6, 9, 11, 28	Zeitraum justieren.....	8, 18, 34
Maximale Anzahl Cluster	9, 11, 28	Zwischenspeicher	28, 32