Gesichtserkennung mit MemBrain

A. Daten erheben

1. Eure Gruppe erhält einen Ordner mit den **Trainingsdaten** (9 Bilder von insgesamt 3 Personen) und einen Ordner mit den **Testdaten** (3 Bilder von insgesamt 3 Personen)

2. Für die Gesichtserkennung müsst ihr zunächst ein **Gesichtsmodell** entwickeln. Dazu sollt ihr beim Betrachten der Fotos ein **eigenes** Gesichts-Modell (mit nur **vier** Merkmalen) entwickeln. Die Fotos sind ungefähr auf den Augenabstand normiert und alle ungefähr aus der gleichen Perspektive aufgenommen. Das erleichtert zwar die Vergleichbarkeit, es ist deswegen jedoch nicht sinnvoll, z.B. den Augenabstand als ein Merkmal zu nehmen. Winkel oder Streckenverhältnisse sind da besser. Die 4 Merkmale könnt ihr z.B. in GIMP erheben. Wie das geht wird im Folgenden gezeigt:

   - **GIMP** starten (ggf. eignen sich auch andere Programme dazu) und ein einzelnes Foto der Trainingsdaten in GIMP öffnen
   - Soweit ins Bild **zoomen**, dass es möglichst groß auf dem Bildschirm zu erkennen ist.
   - Mit dem Werkzeug **Maßband** nun ganz einfach **Distanzen** (in Pixel) und **Winkel** (in °) messen. Dazu das Maßband-Werkzeug auswählen und auf die beiden Endpunkte klicken. GIMP zeigt in einem kleinen Infofeld die gewünschten Informationen an, die ihr dann einfach in eine Textdatei schreiben könnt. Wie das geht wird im Folgenden beschrieben:

3. Anlegen einer **CSV-Datei mit dem Trainingsdaten**: Öffnet ein Textprogramm (wie z.B. Notepad oder Notepad++), mit dem man **reine Textdateien** erstellen kann (nicht Word oder so etwas). Erstellt ggf. eine neue Datei und speichert sie schon einmal unter dem Namen **train.csv** ab (wichtig: nicht .txt). In diese schreibt ihr nun für jedes bearbeitete Bild genau eine Zeile hinein, so dass sich eine Datei ergibt, die wie das folgende Fiktiv-Beispiel aufgebaut ist (Erläuterungen im Folgenden):

```
M1;M2;M3;M4;Y
0.22;0.33;0.44;0.55;0
0.11;0.22;0.33;0.44;0
0.99;0.88;0.77;0.66;0
0.34;0.45;0.56;0.67;0.5
0.23;0.34;0.45;0.56;0.5
0.12;0.23;0.34;0.45;0.5
0.80;0.70;0.60;0.50;1
0.70;0.60;0.50;0.40;1
0.60;0.50;0.40;0.30;1
```

   Dargestellt sind hier die vier Merkmale (mit zwei Nachkommastellen) jeweils gefolgt durch ein Semikolon. Ganz rechts wird dann noch die Nummer der Person eingetragen, zu der das Foto gehört. An dieser Stelle **müssen** jedoch **drei wichtige Vereinbarungen** eingehalten werden:

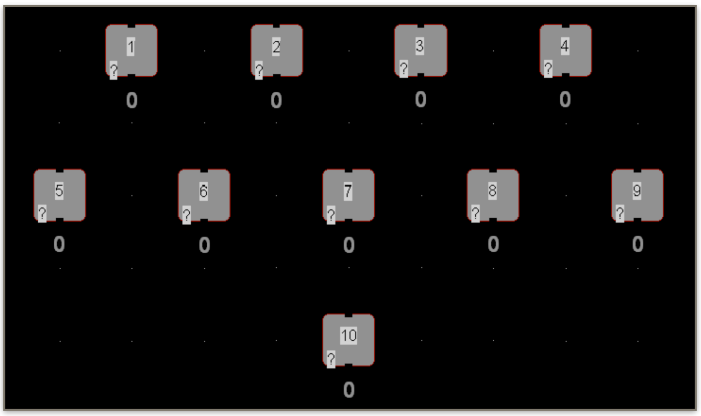
   - **Die erste Zeile der CSV-Datei muss die Namen der später in MemBrain angelegten Neuronen** als Spaltenüberschrift enthalten. Es wird geraten, diese einfach M1, M2, M3 und M4 zu nennen. Als ‚letzte Spaltenüberschrift' darf der Name des Ausgabeneurons nicht fehlen. Am besten nimmt man hier den Namen Y.

- **Der Eingabebereich muss im Bereich [0..1] liegen**. Rohwerte müssen vor der Verarbeitung mit neuronalen Netzen auf den Wertebereich der Aktivierungsfunktionen [0..1] normiert werden. Es reicht aus, dass ihr einfach eine 0 gefolgt von einem Punkt vor euren Wert schreibt. Habt ihr also eine 81 als Wert in GIMP gemessen, wird 0.81 in die CSV-Datei eingetragen. Dies gilt immer für die ersten vier Werte in jeder Zeile.

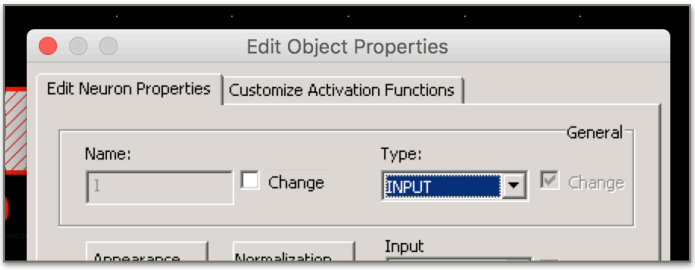
- Der letzte Wert in jeder Zeile enthält das so genannte **Klassenlabel**. Hier muss also kodiert werden, um **welche Person** es sich bei den vier Merkmalen handelt. Da auch hier der Wertebereich [0..1] gewählt werden muss, bietet es sich an, die erste Person mit 0 zu kodieren, die zweite Person mit 0,5 und die dritte Person mit 1.

B. Erstellen des neuronalen Netzes in Membrain

1. Zunächst erfolgt die Modellierung: Mit dem Insert new neurons-Symbol  können nun per Klick Neuronen auf dem Bildschirm verteilt werden. Wir benötigen vier Eingabeneuronen und ein Ausgabeneuron, wenn wir vier Merkmale auf ein Klassenlabel Y abbilden wollen. Dazwischen können sich beliebig viele Schichten von so genannten versteckten Neuronen befinden. Für unsere Zwecke reicht ein kleines Netz mit drei Schichten (Eingabeschicht, verdeckte Schicht und Ausgabeschicht) und ein paar zusätzlichen Neuronen in der verdeckten Schicht mehr als aus. Die Topologie (also quasi das Aussehen des Netzes) ist dabei Geschmacks- und Erfahrungssache und sollte ruhig in den verschiedenen Gruppen unterschiedlich sein.

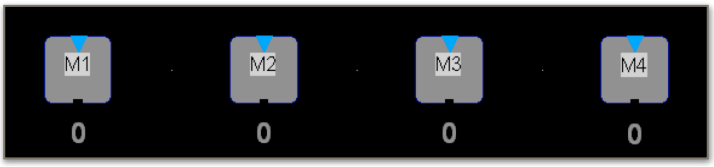


2. Da die Neuronen unterschiedliche **Aufgaben** haben (Eingabe, verdeckt, Ausgabe), müssen diese noch festgelegt werden. Dazu kann man in MemBrain bei gedrückter Maustaste gleich mehrere Neuronen auswählen und dann auf eines mit einem Doppelklick klicken, um die Eigenschaften für alle ausgewählten Neuronen gleichzeitig festzulegen. In der oberen Reihe (bei den **Eingabeneuronen**) muss dazu im aufspringenden Menü unter Type der Eintrag INPUT gewählt werden.

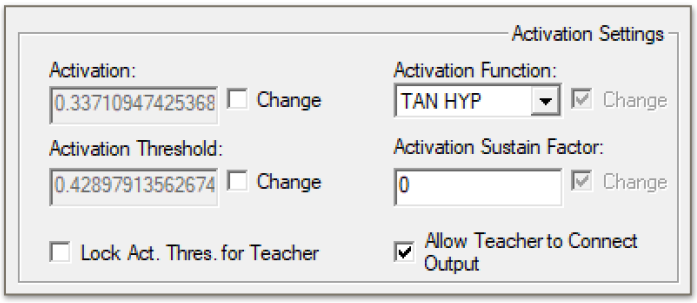


Im Anschluss müssen die Neuronen noch dieselben **Namen** wie in der CSV-Datei angegeben haben. Dazu klickt man nacheinander auf jedes Eingabeneuron und ändert den Namen des Neurons unter Name entsprechend ab. Also M1, M2, M3 und M4.

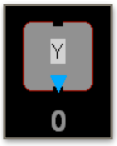
Die Eingangsschicht sollte dann wie folgt aussehen:



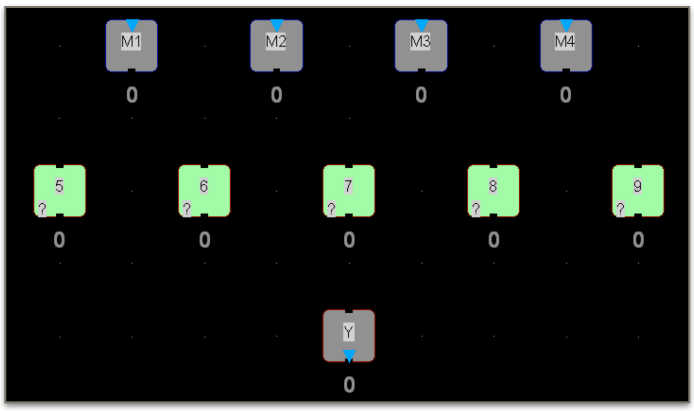
Die **Neuronen in der verdeckten Schicht** müssen auch bearbeitet werden. Man kann einfach alle (in der mittleren Reihe) gleichzeitig auswählen und auf eines doppelklicken. Es öffnet sich dann ein Eigenschaftenfenster, in dem alle verdeckten Neuronen gleichzeitig bearbeitet werden können. Hier muss unter Activation Settings in der Mitte beim Punkt Activation Function die Aktivierungsfunktion **TAN HYP** (Tangens hyperbolicus) ausgewählt werden. Diese Funktion skaliert die Gewichte zwischen den Neuronen auf den Wertebereich von [-1 .. 1]. So können verdeckte Neuronen quasi sowohl hemmend als auch aktivierend wirken.



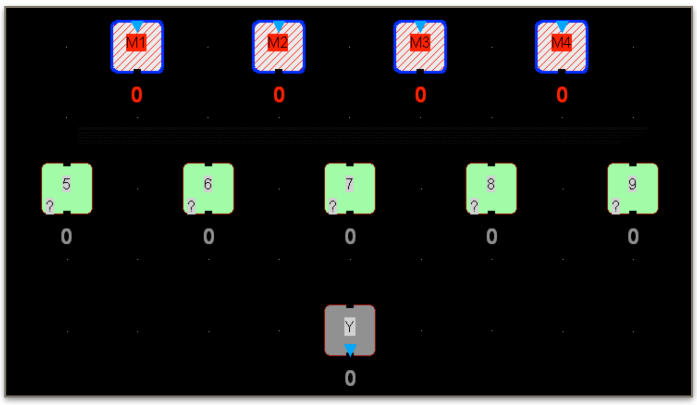
Das einzelne **Ausgabeneuron** wird per Doppelklick ausgewählt, danach wird im Feld Name der Name Y eingetragen und es wird unter Type als OUTPUT deklariert. Das Ausgabeneuron sieht nun also wie folgt aus (die kleinen blauen Pfeile zeigen übrigens den Verwendungszweck an):



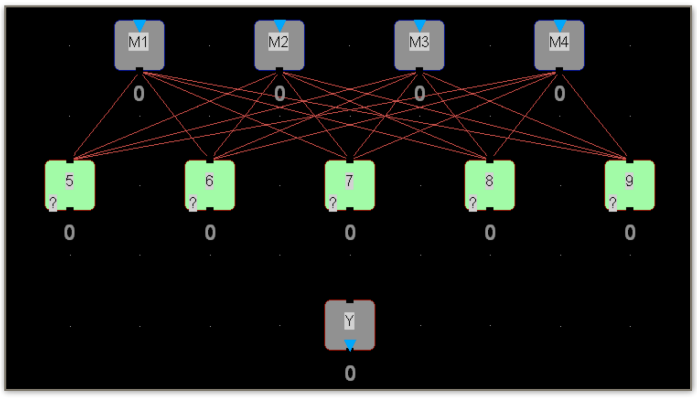
3. Damit auch wirklich ein Netz entsteht, müssen die Neuronen miteinander verbunden werden.  Dazu bietet MemBrain das Konzept der **Extraauswahl**. Mit dem Symbol Add to Extra Selection kann man gleich mehrere Neuronen in die Extraauswahl aufnehmen. Sinnvollerweise markiert man dazu die Neuronen der verdeckten Schicht und klickt auf das nebenstehend abgebildete Symbol. Die Neuronen erscheinen daraufhin grün.



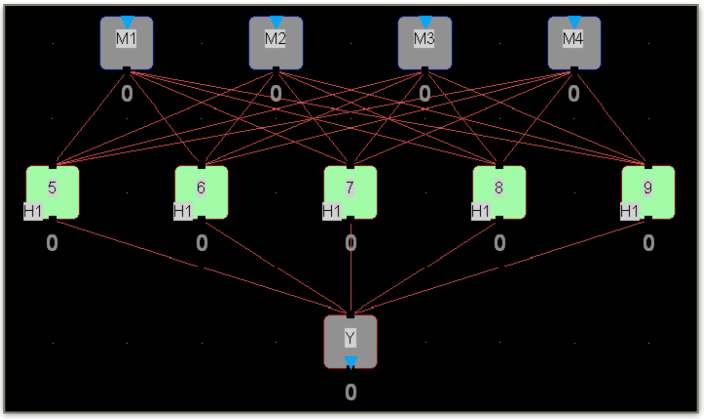
Nun ist es möglich, diese alle auf einmal mit der kompletten oberen Eingangsschicht zu **verbinden**. Dazu wird jetzt die Eingangsschicht ausgewählt (mit gedrückter Maustaste alle Neuronen der Eingangsschicht auswählen) …



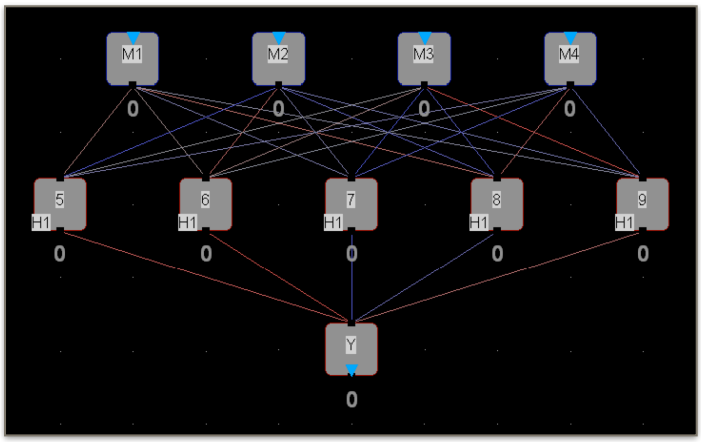
… und danach dann auf das Symbol Connect TO Extra Selection klicken. Nun werden die beiden Schichten automatisch verbunden und man spart sich Bearbeitungszeit. Das Ergebnis sieht dann wie folgt aus:



Danach reicht ein Klick auf das **Ausgabeneuron**, um danach mit einem Klick auf das Symbol Connect FROM Extra Selection die verdeckte Schicht vollständig mit dem Ausgabeneuron zu verbinden.



Es bleibt, zuletzt noch die Extraauswahl wieder **rückgängig** zu machen. Dies geht mit einem Klick auf das Symbol Clear Extra Selection.

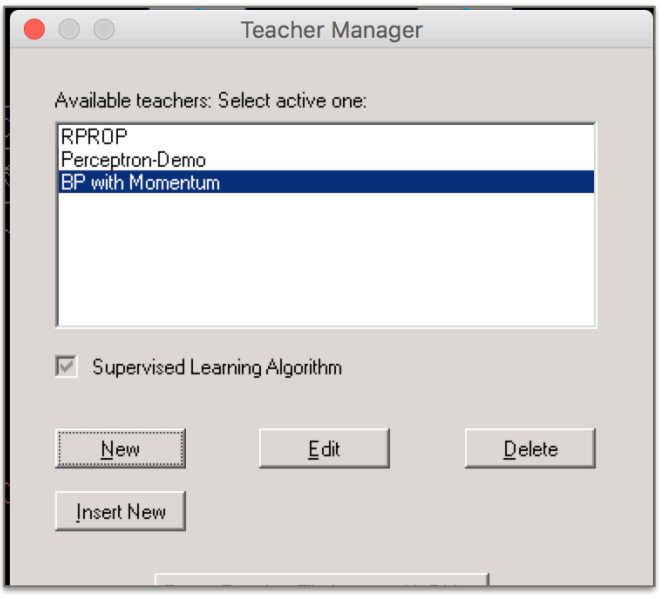


Das Netz ist nun **bereit** für die Gesichtserkennung

4. Der Teacher Manager, der im Menü Teach zu finden ist, ist Steuerzentrale für unterschiedliche Lernalgorithmen. Um die **richtige Lernregel** auszuwählen, geht ihr bitte wie folgt vor:

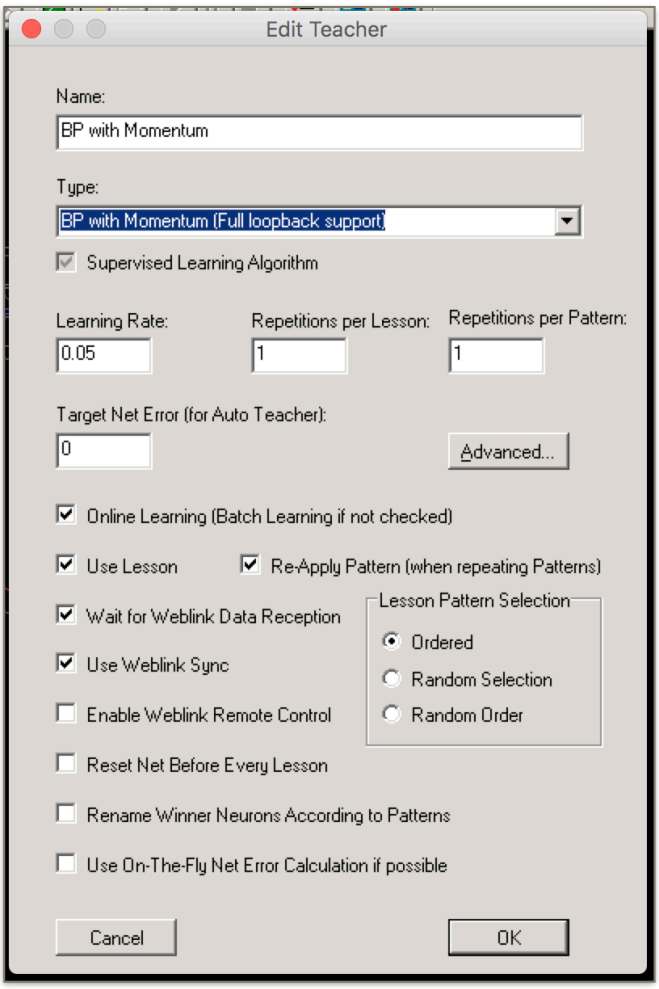
   - Im Menü Teach auf den Eintrag Teacher Manager klicken. Es öffnet sich ein Dialog wie in der linken Abbildung (mit eventuell weniger Einträgen bei einer Neuinstallation des Programms):

   - Klickt nun auf New, um eine neue Lernregel zu erstellen und wählt dann BP with Momentum (Full loopback support) oben aus der Liste des neuen Dialogfensters.



- Stellt danach alle anderen Parameter, wenn nötig, so ein wie auf dem rechten Bild gezeigt:

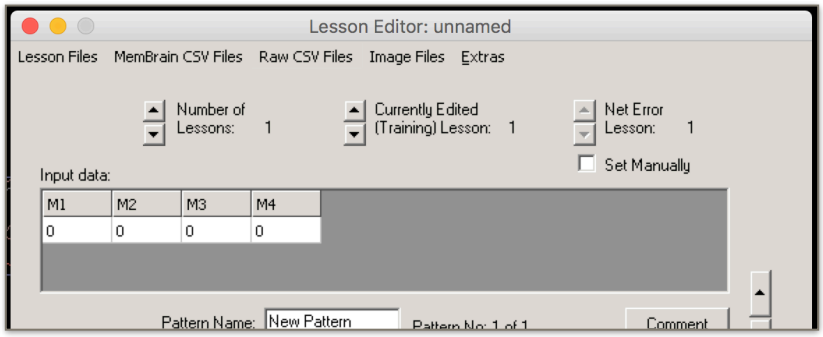
- OK klicken, das neue Lernverfahren auswählen und nochmals OK drücken. Die Einstellung bleibt gespeichert



C. Trainieren und Testen des neuronalen Netzes

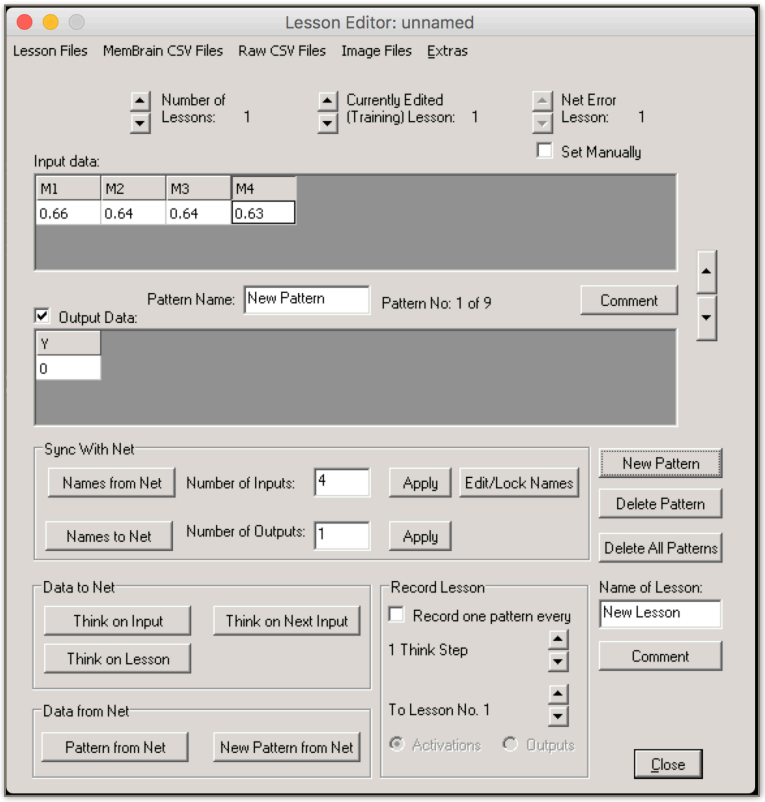
1. Vor jedem Training muss das Netz mit einem Klick auf das Symbol Randomize Net **randomisiert** werden. Die Farbgebung der Verbindungen zwischen den Netzen repräsentiert (wie bei den Neuronen) einen numerischen Wert.

2. Mit einem Klick auf das Symbol Show Lession Editor wird ein Dialogfenster geöffnet, mit dem man u.a. CSV-Dateien bequem **importieren** kann.



Im Menü Raw CSV Files ist unter dem Menüpunkt Import Current Lesson (Raw CSV) … die Funktionalität versteckt, die Trainingsdaten-Datei zu laden.

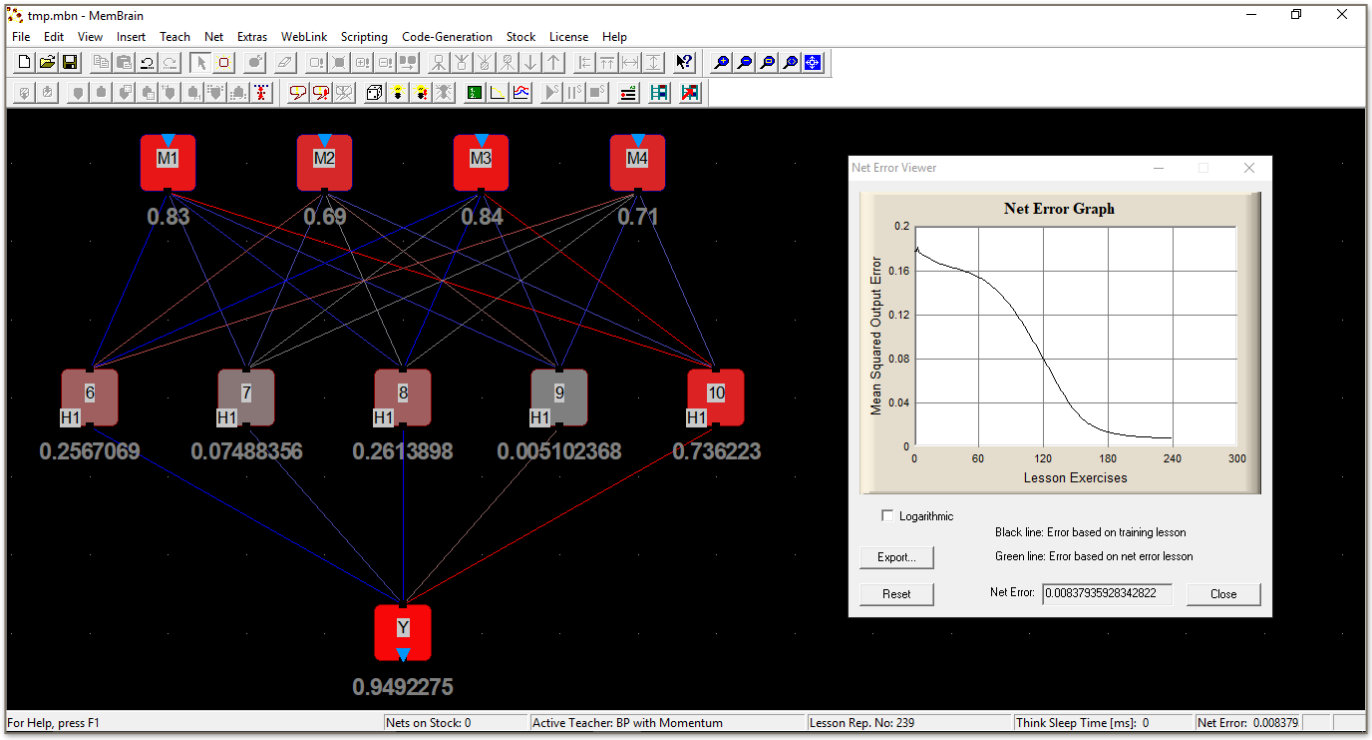
Nachdem die richtige Datei ausgewählt wurde, werden die Muster automatisch geladen und es sollte auch angezeigt werden, dass nun neun Muster verfügbar sind, durch die man mit den beiden Pfeil-Buttons navigieren kann, evtl. auch, um eine finale Kontrolle vorzunehmen. Ist alles in Ordnung, kann der Dialog mit einem Klick auf Close geschlossen werden.



3. Bevor das Training startet, sollte unbedingt noch mit einem Klick auf das Symbol Show Net Error Viewer das **Fenster für den Fehler des Netzes** geöffnet werden. Dieses kann währenddessen geöffnet bleiben und zeigt sehr schön, wie der Fehler des Netzes während des Trainings minimiert wird bis das Netz alle Gesichter ausreichend gut gelernt hat.

Mit einem Klick auf das Symbol Start Teacher kann das Training nun endlich **gestartet** werden.

Jedoch wird bei der geringen Menge der Daten der Fehler des Netzes im Error Viewer sehr schnell sinken, weswegen ein **baldiger** Klick auf das Symbol Stop Teacher erfolgen sollte, sobald die Fehlerkurve erkennbar ihren tiefsten Punkt erreicht.



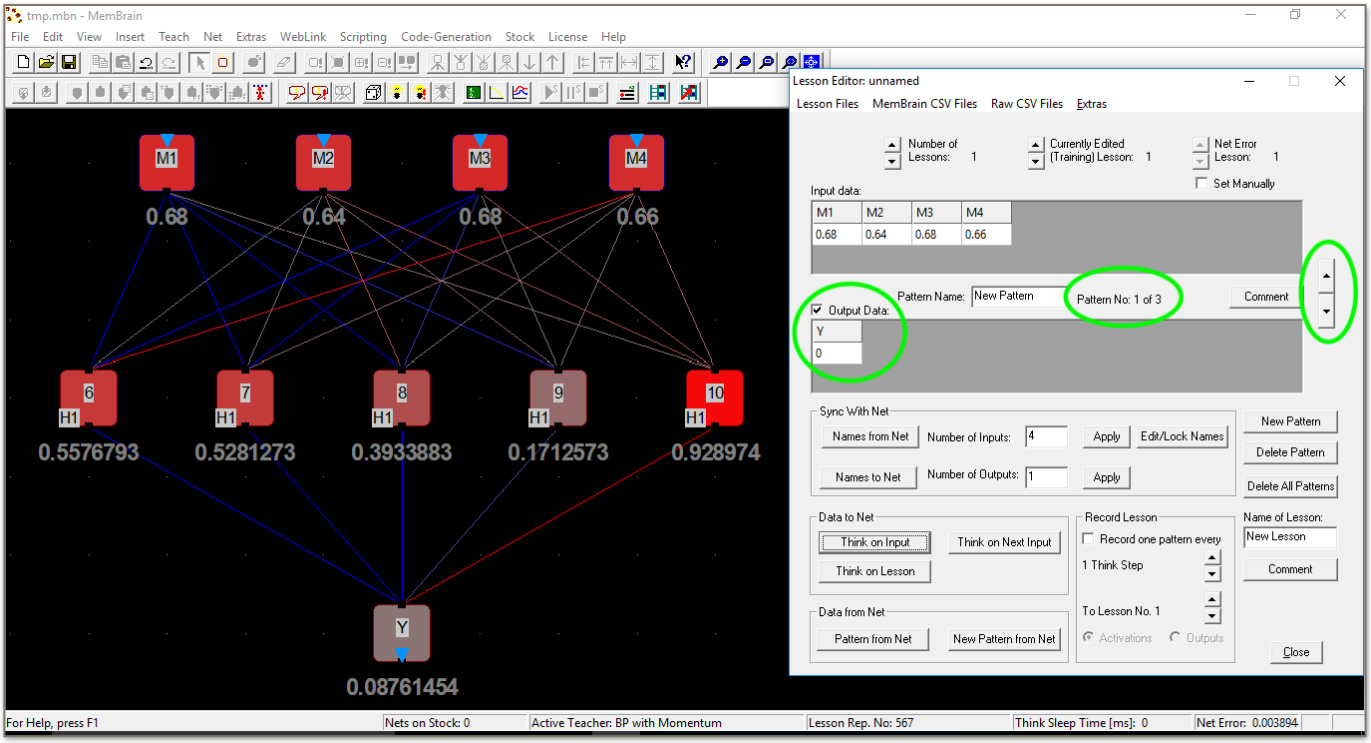
4. Zum **Testen** des Netzes müsst ihr zunächst eine **neue** CSV-Datei erstellen, die ihr z.B. unter dem Namen test.csv abspeichert. Diese Datei muss exakt gleich aufgebaut sein wie die Trainingsdaten-Datei. Wieder ermittelt ihr (dieses Mal nur für drei Fotos) die vier exakt gleichen Merkmale und tragt sie dann entsprechend in die CSV-Datei ein.

   Die fertige CSV-Datei könnte z.B. wie folgt aussehen (hier nur fiktives Beispiel):

   ```
   M1;M2;M3;M4;Y
   0.12;0.34;0.56;0.78;0
   0.99;0.99;0.99;0.99;0.5
   0.77;0.66;0.55;0.44;1
   ```

5. Ladet In MemBrain nun die **Testdaten** in den Lesson Editor auf die gleiche Art und Weise wie die Trainingsdatei importiert wurde. Sie ersetzt nun damit die bisher neun vorhandenen Muster durch die drei Muster, die sich in der Datei befinden. Auch hier bietet es sich an, nochmals zu kontrollieren, ob alle Werte richtig übernommen wurden. Ist das der Fall, kann der Test **direkt** im Lession Editor gestartet werden.

6. Das Dialogfenster sollte am besten so auf dem Bildschirm platziert werden, dass neben ihm auch das Netz sichtbar ist. Für das **Testen** sind dann alle in der folgenden Abbildung grün markierten Teile des Dialogfensters von Bedeutung:



Zunächst wird mit den Pfeil-Buttons rechts das erste Muster ausgewählt, so dass in der Mitte Pattern No. 1 of 3 steht und unter Output Data ebenso eine 0 als Klassenlabel zu sehen ist. Dann bewirkt ein Klick auf den Button **Think on Input** im Bereich Data to Net, dass das aktuelle Testmuster durch das Netz geschickt wird und das Netz im Haupfenster von MemBrain direkt unter dem Ausgabeneuron anzeigt, **welcher Person** es das Muster zuordnet.

Wird nun also die richtige Person erkannt, dann notiert dies auf einem Zettel und wechselt mit den Pfeil-Buttons zum nächsten Muster.

Ein Wert nahe 0 bedeutet also, dass hier auch die Person mit dem Klassenlabel 0 erkannt wurde. Ein Wert von 0.39 würde darauf hindeuten, dass das Netz hinter den aktuellen vier Merkmalen eher die Person mit dem Label 0.5 vermutet als die Person mit dem Label 0. Um welche Person es sich (zum Vergleich) handelt, lässt sich in dem ebenfalls grün markierten Bereich unter Output Data erkennen (die 0 in diesem Fall wird für das KNN schlicht ignoriert und dient nur der eigenen Kontrolle).

7. Im Anschluss kann dann recht einfach die **Erkennungsrate** ermittelt werden, die je nach Netztopologie und gewählten Merkmalen tendenziell eher bei 100% liegen sollte.