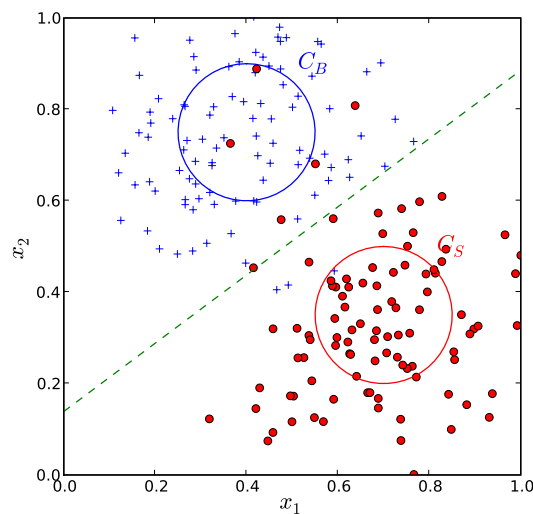


**Übung zur Vorlesung “Statistische Methoden der Datenanalyse”
H. Kolanoski, A. Schälicke – SS 2008**

Übung 11

11.1 Klassifikation mit Fisher-Diskriminante



Eine Signalverteilung C_S und eine Untergrundverteilung C_B soll durch eine Fisher-Diskriminante in den Variablen (x_1, x_2) optimal geteilt werden.

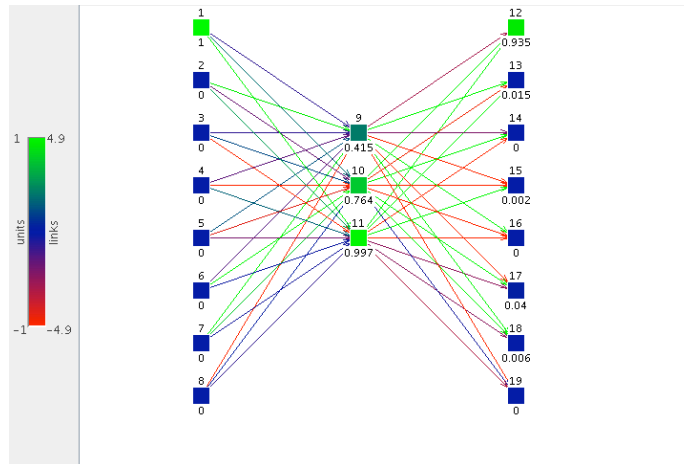
- Erzeugen Sie 10000 Ereignisse einer Signalverteilung als zwei-dimensionale Gauss-Verteilung mit Mittelwerten $(0.7, 0.35)$ und $\sigma = 0.15$, und als Untergrundverteilung eine zwei-dimensionale Gauss-Verteilung mit Mittelwerten $(0.4, 0.75)$.
- Bestimmen Sie die Fisher-Diskriminante $t(\vec{x}|\vec{a})$, d.h. finden Sie die Parameter a_1, a_2 , die die beiden Ereignismengen C_S und C_B optimal trennt.
- Bestimmen Sie Effizienz ϵ und Reinheit π der Signal-Selektion aus der Stichprobe in Abhängigkeit vom kritischen Wert t_c .

$$t = a_1 x_1 + a_2 x_2 < t_c$$

$$\epsilon = \frac{N_S(t < t_c)}{N_S} \qquad \pi = \frac{N_S(t < t_c)}{N_S(t < t_c) + N_B(t < t_c)}$$

Hinweis: Ob die Bedingung $t < t_c$ oder $t > t_c$ lauten muß, hängt von der Vorzeichenwahl der $a_{1,2}$ ab.

11.2 8-3-8 Dekoder mit Neuronalem Netz



Die Funktionsweise eines künstlichen neuronalen Netzes mit jeweils 8 Eingangs- und Ausgangsneuronen und 3 Neuronen in der versteckten Zwischenschicht soll untersucht werden. Das Netz soll darauf trainiert werden, dass wenn ein Eingangsneuron aktiviert wird, das zugehörige Ausgangsneuron ebenfalls aktiviert wird. Der erste Teil des Netzwerkes wirkt dann als 8-3-Dekoder, und der zweite Teil als 3-8-Enkoder.

- Wie ist eine solche Datenkompression möglich?
- Bestimmen Sie alle Vorzeichen der Gewichte in der Dekoder-Schicht. Wie ist das Muster zu erklären?

Hinweise

- Hilfe zum SNNS, bzw. JavaNNS findet man unter www.ra.cs.uni-tuebingen.de/SNNS/.
- Python-Beispiele zum SNNS, insbesondere ein Encoder-Beispiel, gibt es im Pool unter:
`/users/eel/dreas/python/snns_example`
- Die JavaNNS startet man am einfachsten mit `/users/eel/dreas/python/bin/JavaNNS`. Beispiel `net-` und `pat-` Dateien gibt es unter `/users/eel/dreas/python/javanns_example`
- Vor der Verwendung von SNNS, MINUIT, PYCUBA oder ROOT in Python müssen folgende Umgebungsvariablen gesetzt werden:

```
export PYTHONPATH=/users/eel/dreas/python/packages:/usr/global/root/lib
export LD_LIBRARY_PATH=/usr/global/root/lib:/users/eel/dreas/python/lib
```

Das geschieht am einfachsten in der Initialisierungsdatei der Shell, z.B. für `bash` in der Datei `.bashrc`.

- In ROOT stellt die Klasse `TMultiLayerPerceptron` die Funktionalität künstlicher neuronaler Netze bereit.
- Ein modernes Python Framework zur Datenverarbeitung stellt das *Modular toolkit for Data Processing (MDP)* (<http://mdp-toolkit.sourceforge.net>) dar, aber bisher leider nicht im Physik-Pool.