

# Verfahren zur Datenanalyse gemessener Signale

Dr. rer. nat. Axel Hutt

Vorlesung 11

### 3. Aufgabe: Trennung von Neuronen

- Principal Component Analysis
- Clusteranalyse

# Principal Component Analysis:

$$\mathbf{q}(t) = \sum_{i=1}^N x_i(t) \mathbf{v}_i \quad , \quad \mathbf{q} \in \mathbb{R}^S \quad , \quad \int \mathbf{q}(t) dt = 0$$

Annahme:  $\mathbf{v}_i \mathbf{v}_j = \delta_{ij}$  ,  $\forall i, j = 1, \dots, N$

$$\mathbf{v}_k \mathbf{q}(t) = \sum_{i=1}^N x_i(t) \mathbf{v}_k \mathbf{v}_i$$

$$x_i(t) = \mathbf{q}(t) \mathbf{v}_i \quad , \quad \forall i, j = 1, \dots, N.$$

Nun : Näherung durch  $M < N$  Moden

$$\mathbf{q}(t) \approx \tilde{\mathbf{q}}(t) = \sum_{i=1}^M (\mathbf{q}(t) \mathbf{v}_i) \mathbf{v}_i$$

# .....Principal Component Analysis (1)

$$\mathbf{q}(t) \approx \tilde{\mathbf{q}}(t) = \sum_{i=1}^M (\mathbf{q}(t) \mathbf{v}_i) \mathbf{v}_i$$

Fehler durch Näherung:

$$E = \frac{\langle ||\mathbf{q}(t) - \tilde{\mathbf{q}}(t)||^2 \rangle}{\langle \mathbf{q}^2(t) \rangle} \quad 0 \leq E \leq 1 \quad , \quad \langle \cdot \rangle = \int dt$$

$$= \frac{\langle ||\sum_{i=1}^N x_i(t) \mathbf{v}_i - \sum_{i=1}^M x_i(t) \mathbf{v}_i||^2 \rangle}{\langle \mathbf{q}^2(t) \rangle}$$

$$= \frac{\langle ||\sum_{i=M+1}^N x_i(t) \mathbf{v}_i||^2 \rangle}{\langle \mathbf{q}^2(t) \rangle}$$

## .....Principal Component Analysis (2)

$$= \sum_{i,j=M+1}^N \frac{\langle x_i(t)x_j(t)\mathbf{v}_i\mathbf{v}_j \rangle}{\langle \mathbf{q}^2(t) \rangle}$$

$$= \sum_{i,j=M+1}^N \frac{\langle (\mathbf{q}(t)\mathbf{v}_i)(\mathbf{q}(t)\mathbf{v}_j) \rangle \mathbf{v}_i\mathbf{v}_j}{\langle \mathbf{q}^2(t) \rangle}$$

$$= \sum_{i=M+1}^N \frac{\langle (\mathbf{q}(t)\mathbf{v}_i)(\mathbf{q}(t)\mathbf{v}_i) \rangle}{\langle \mathbf{q}^2(t) \rangle}$$

## ....Principal Component Analysis (3)

$$= \sum_{i=M+1}^N \frac{\langle (\mathbf{q}(t)\mathbf{v}_i)(\mathbf{q}(t)\mathbf{v}_i) \rangle}{\langle \mathbf{q}^2(t) \rangle}$$

$$(\mathbf{q}^t \mathbf{v})^t (\mathbf{q}^t \mathbf{v}) = \sum_i q_i v_i \sum_j q_j v_j$$

$$= \sum_{i,j} v_i (q_i q_j) v_j$$

$$= \sum_{i,j} v_i Q_{ij} v_j$$

$$= \mathbf{v}^t \mathbf{Q} \mathbf{v}$$

$$= \sum_{i=M+1}^N \mathbf{v}_i \mathbf{C} \mathbf{v}_i$$

## ....Principal Component Analysis (4)

$$E = \sum_{i=M+1}^N \mathbf{v}_i \mathbf{C} \mathbf{v}_i \quad C_{ij} = \frac{\langle q_i(t) q_j(t) \rangle}{\langle \mathbf{q}^2(t) \rangle}$$

Kovarianzmatrix

E ist der Fehler, den man bei der Näherung durch  $M < N$  Moden macht.

Wie muss man  $\mathbf{v}_i$  wählen, damit der Fehler minimal ?

Dabei ist die Basis  $\{\mathbf{v}_i\}$  normiert.

$$V(\{\mathbf{v}_k\}) = \sum_{i=M+1}^N \mathbf{v}_i \mathbf{C}_i \mathbf{v}_i - \sum_{i=M+1}^N \lambda_i (\mathbf{v}_i^2 - 1).$$

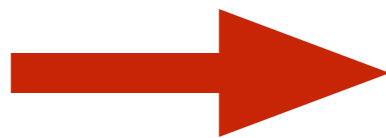
## ....Principal Component Analysis (5)

$$V(\{\mathbf{v}_k\}) = \sum_{i=M+1}^N \mathbf{v}_i \mathbf{C}_i \mathbf{v}_i - \sum_{i=M+1}^N \lambda_i (\mathbf{v}_i^2 - 1)$$

$$\frac{\partial V}{\partial \mathbf{v}_k} = 0$$

$$= 2\mathbf{C}\mathbf{v}_k - 2\lambda_k \mathbf{v}_k$$

↑  
Lagrange-Parameter



$$\mathbf{C}\mathbf{v}_k = \lambda_k \mathbf{v}_k$$

↑  
Eigenwert

$$\begin{aligned} \mathbf{v}_k \mathbf{C} \mathbf{v}_l &= \langle x_k(t) x_l(t) \rangle \\ &= \lambda_k \delta_{kl}. \end{aligned}$$

$$\langle x_k^2(t) \rangle = \lambda_k$$



## ....Principal Component Analysis (6)

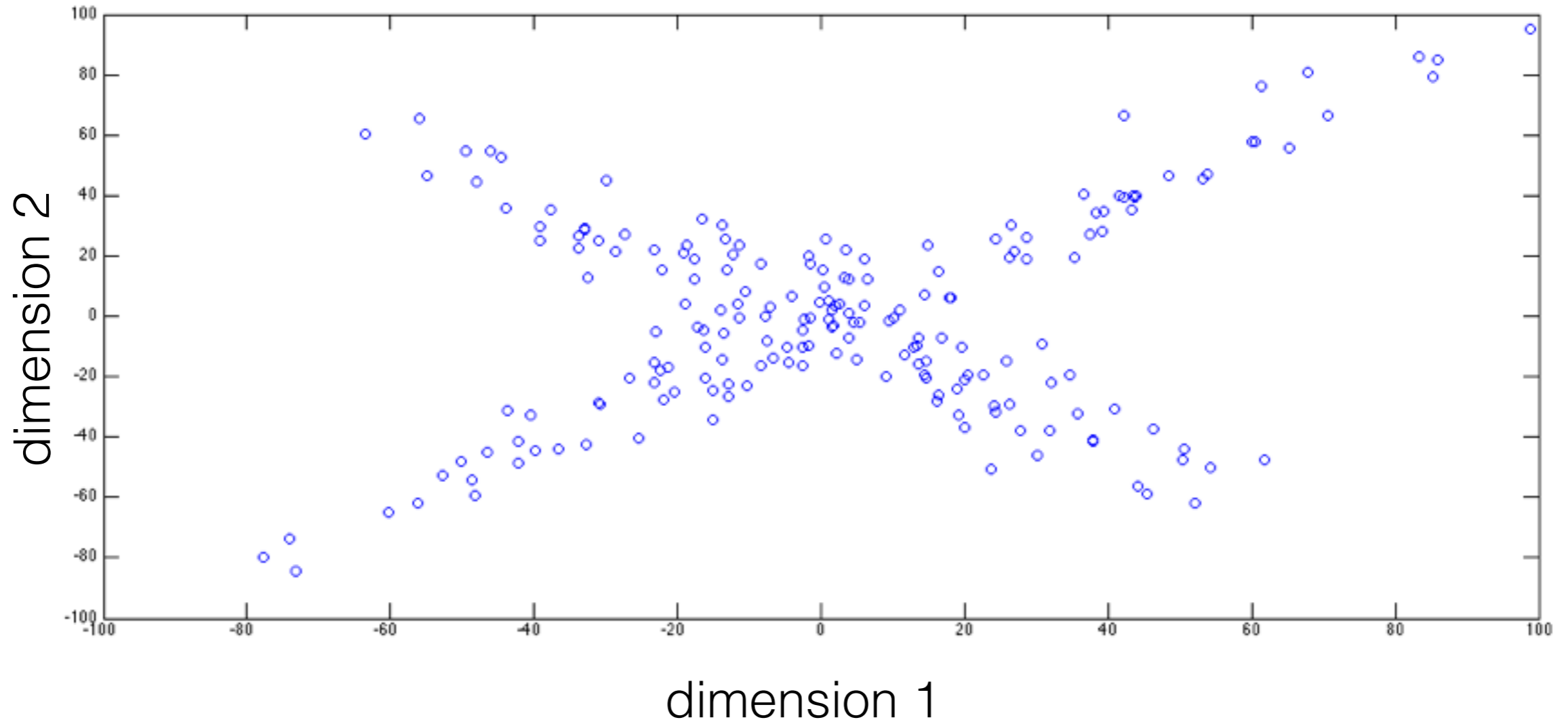
eingesetzt : Fehler:  $E = 1 - \sum_{i=1}^M \lambda_i$

Ranking der Eigenwerte:  $\lambda_1 > \lambda_2 > \lambda_3 > \dots$

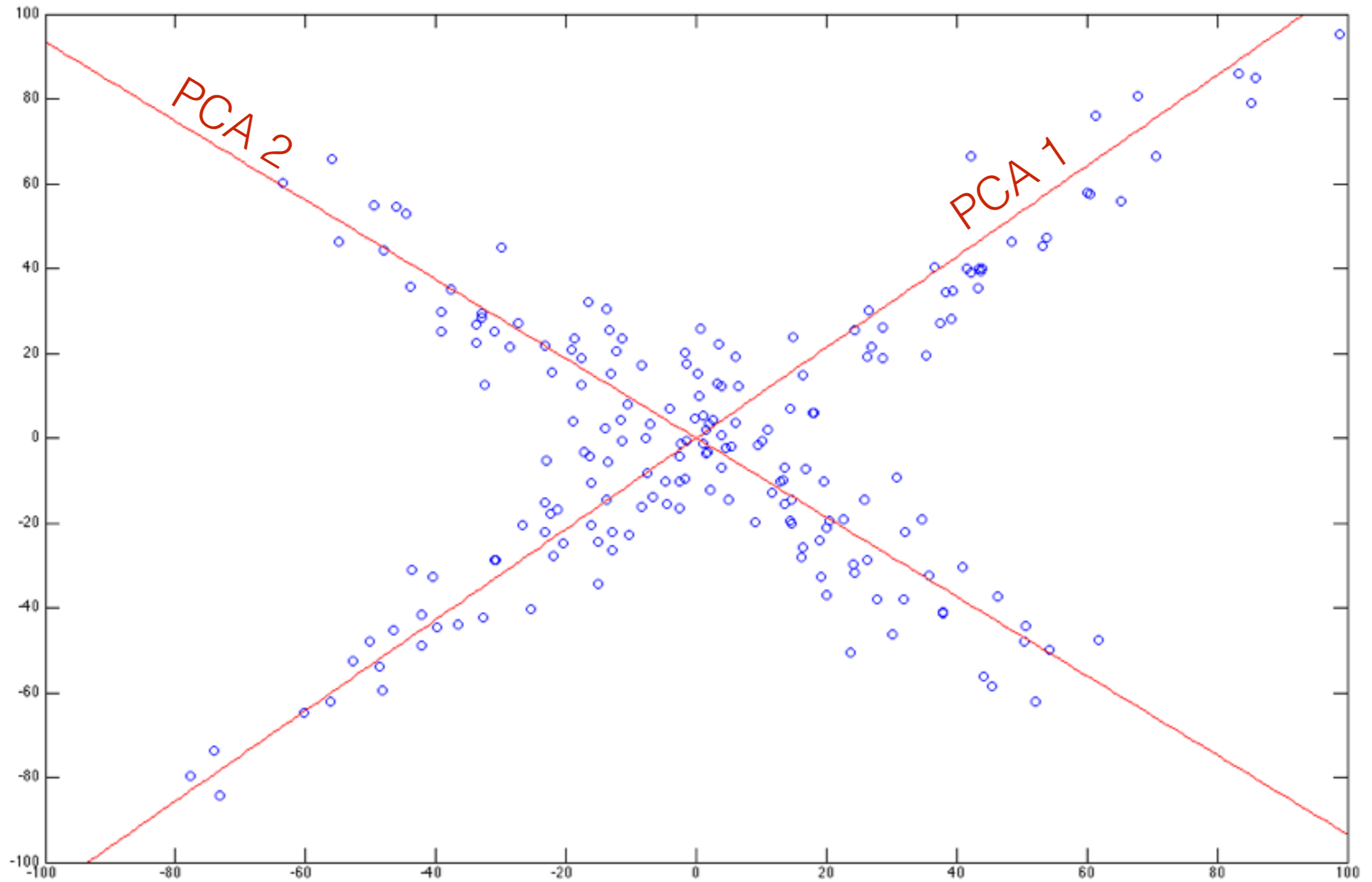
Erste Mode hat größten Beitrag zum Signal,  
die zweite Mode den zweitgrößten etc.

Hierarchie der Eigenvektoren

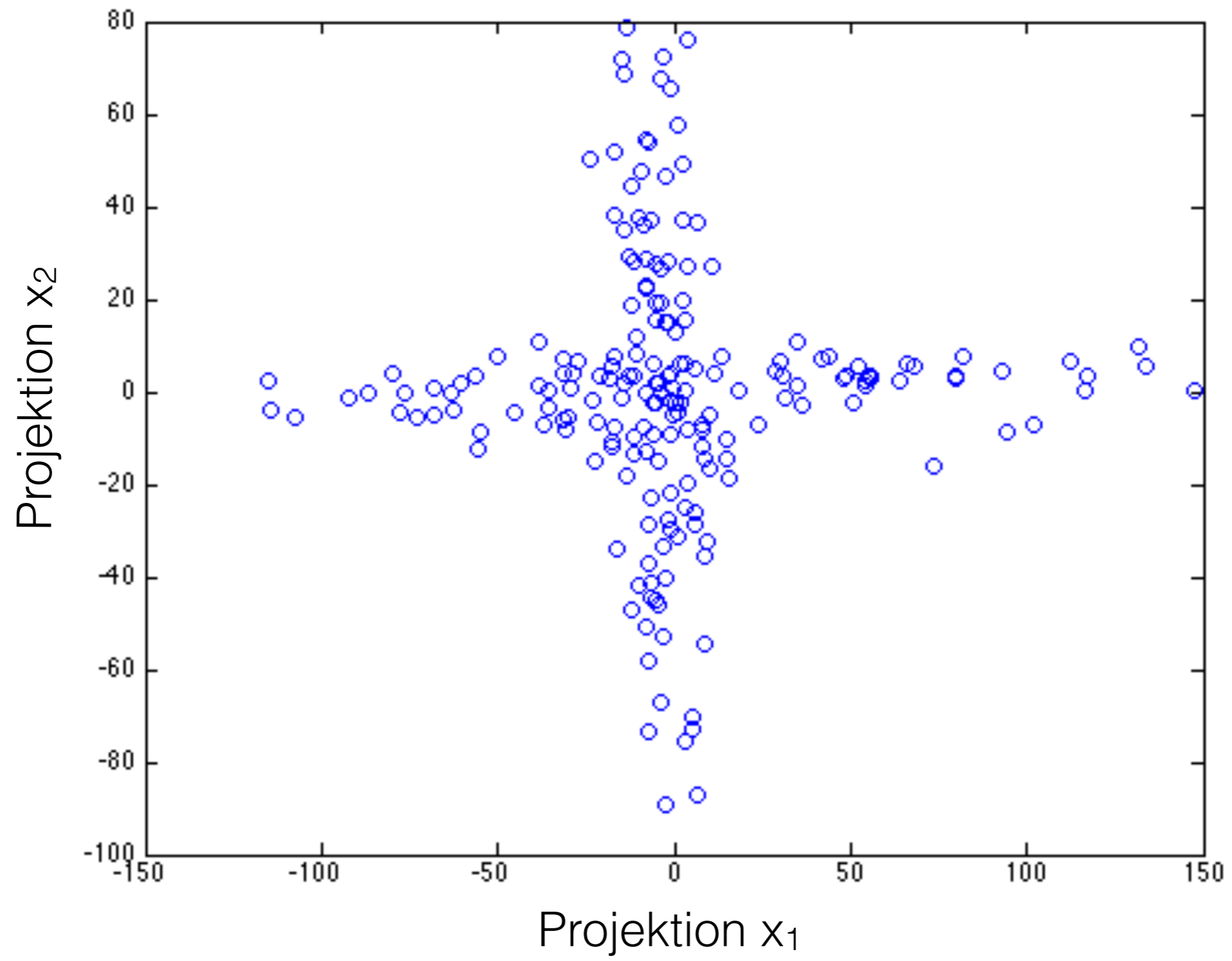
# Anwendung auf 2-dimensionale Daten:



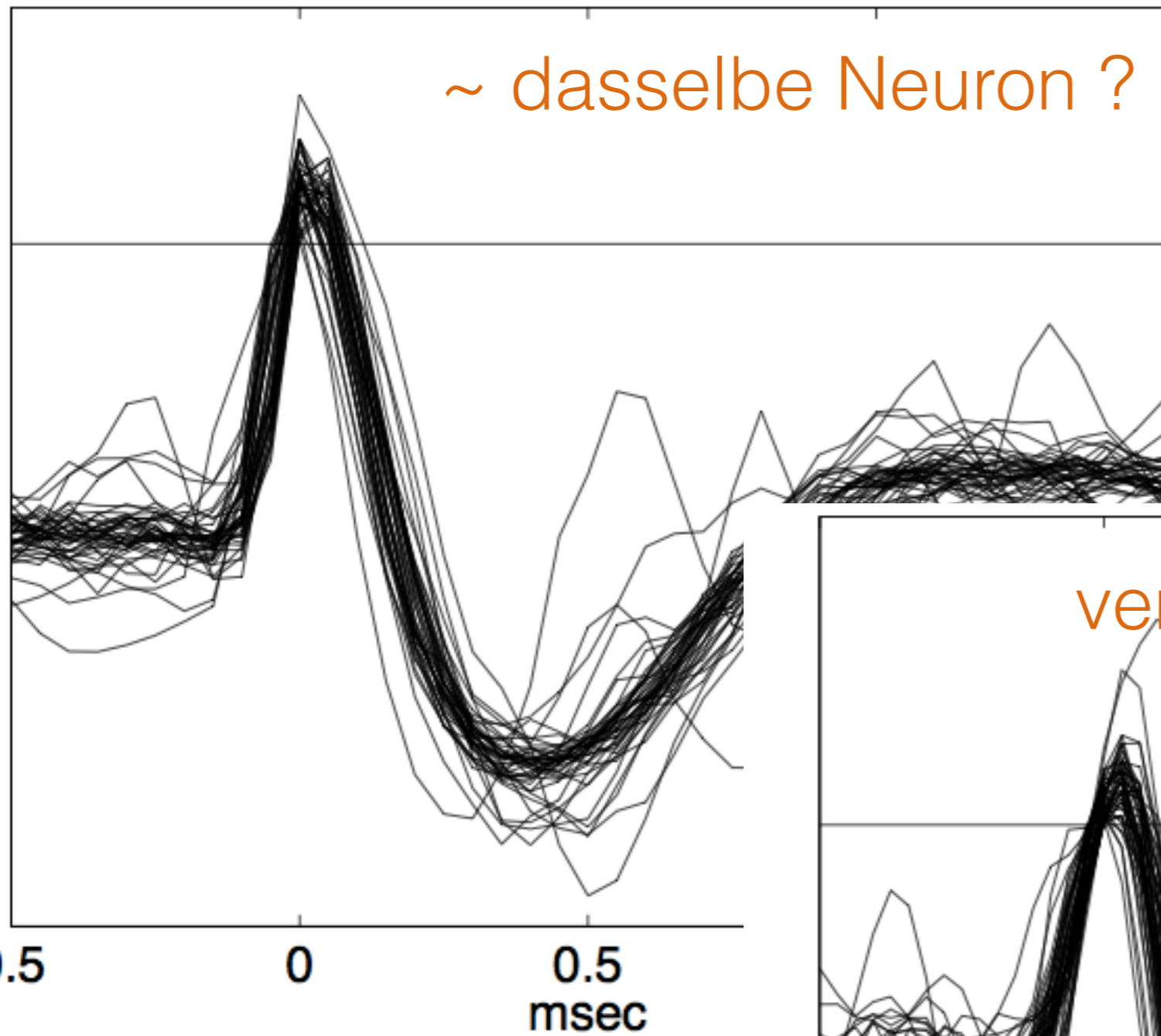
# Ergebnisse:



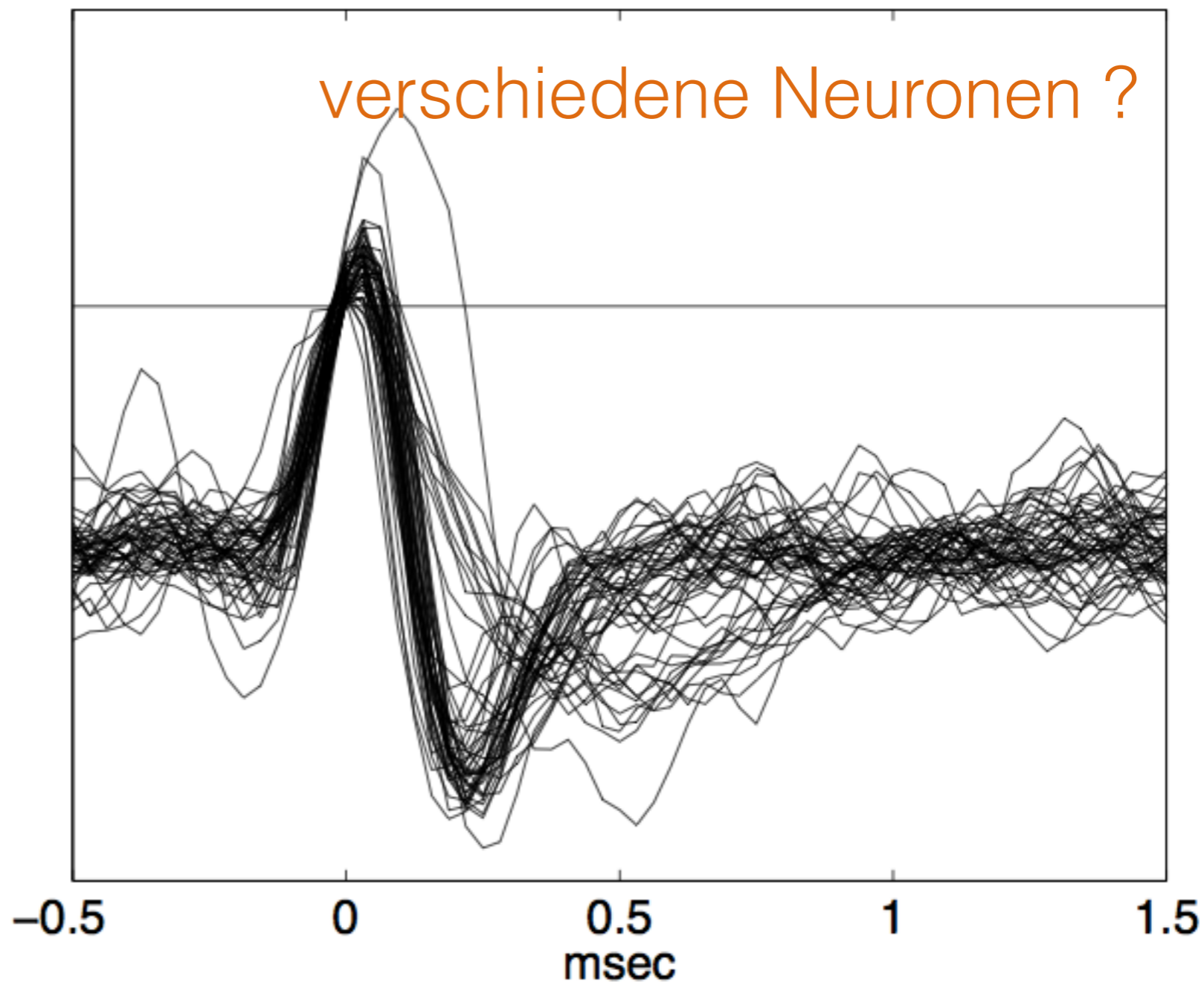
$$\lambda_1=0.61, \lambda_2=0.39$$



~ dasselbe Neuron ?



verschiedene Neuronen ?



# Anwendung auf simulierte spike-Daten:

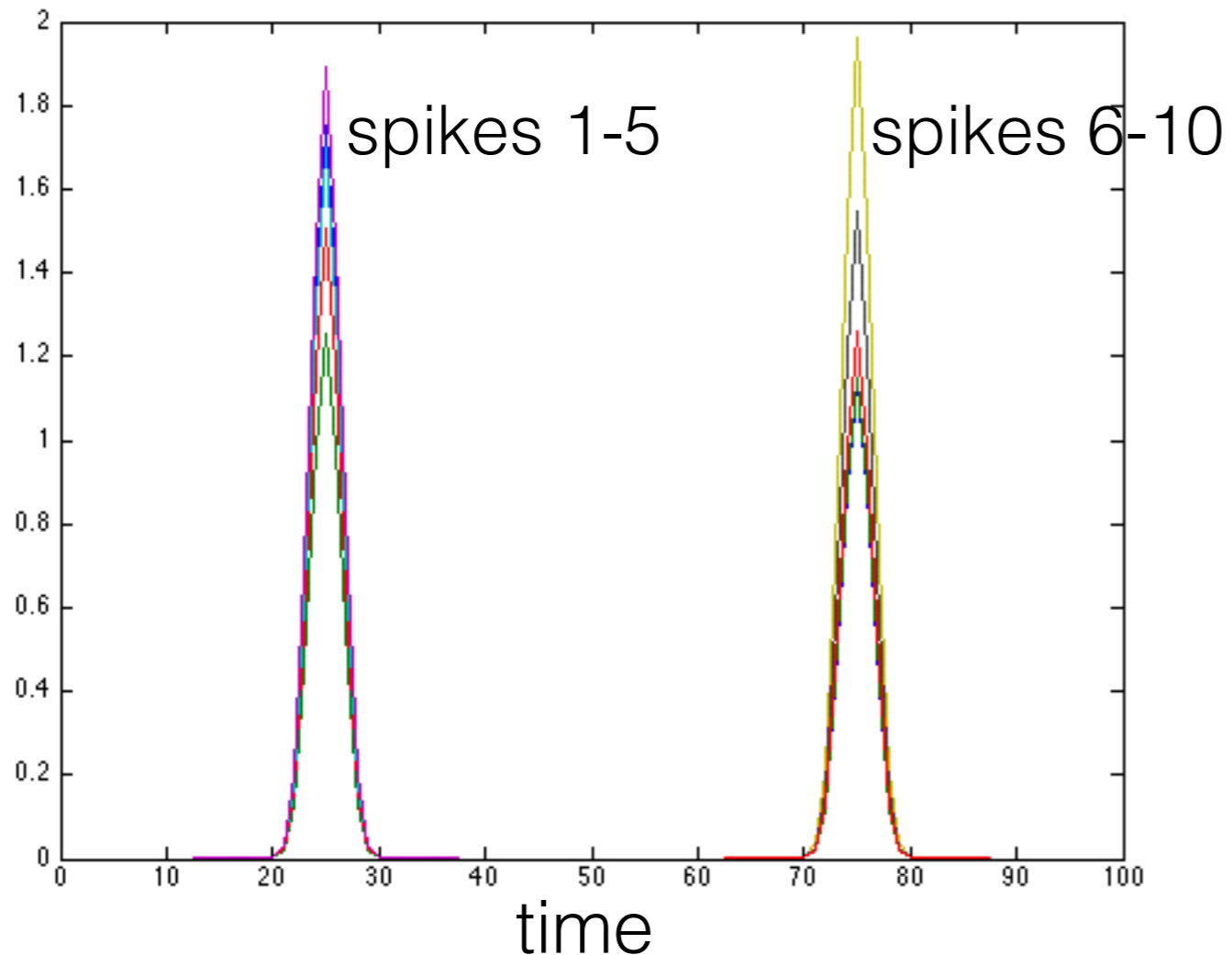
$$\mathbf{s}(n) = \sum_{i=1}^T c_i \mathbf{s}_i(n) , \quad n = 1, \dots, S , \quad \mathbf{s} \in \mathbb{R}^T$$

↑

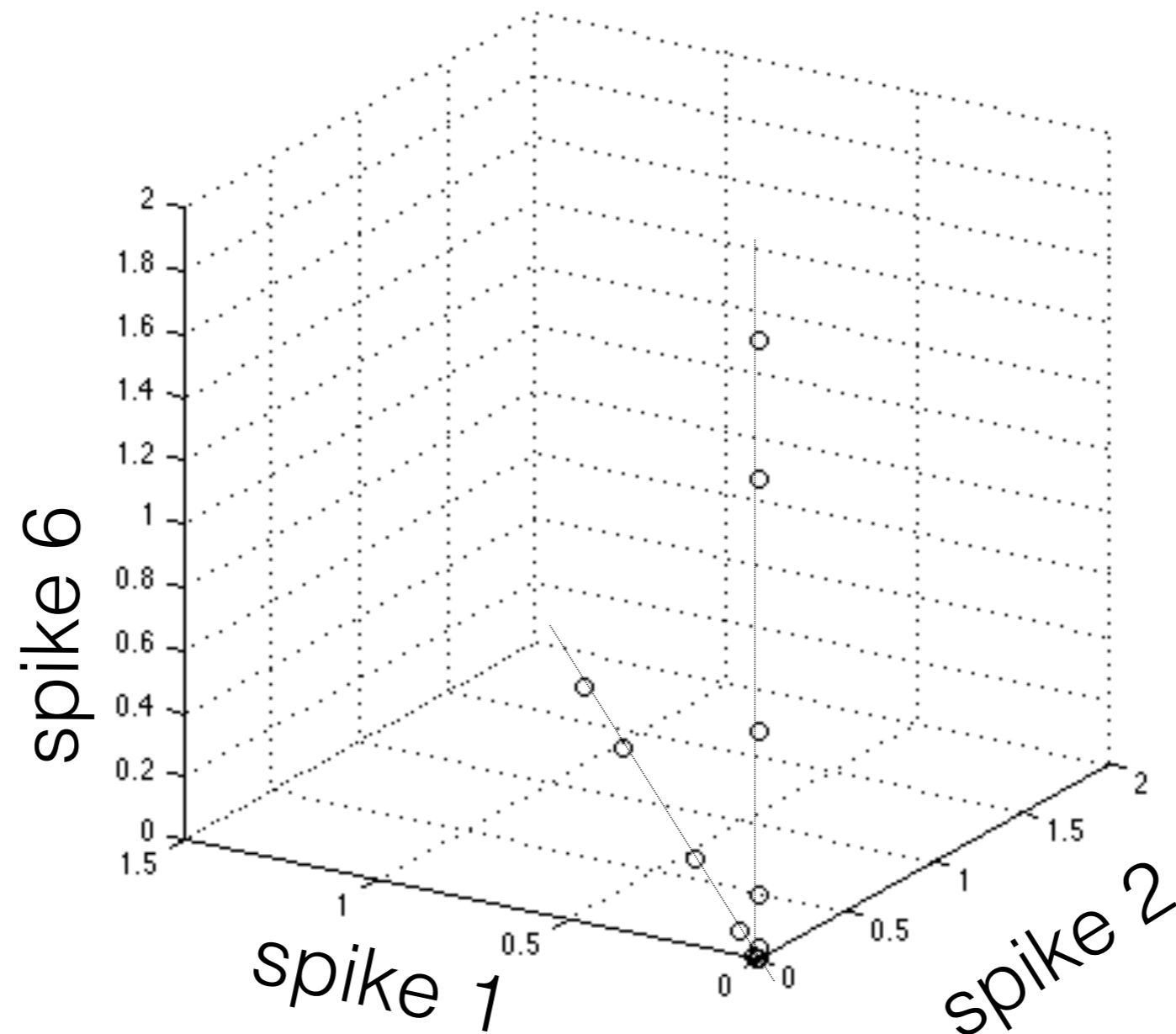
time vector of spike trains

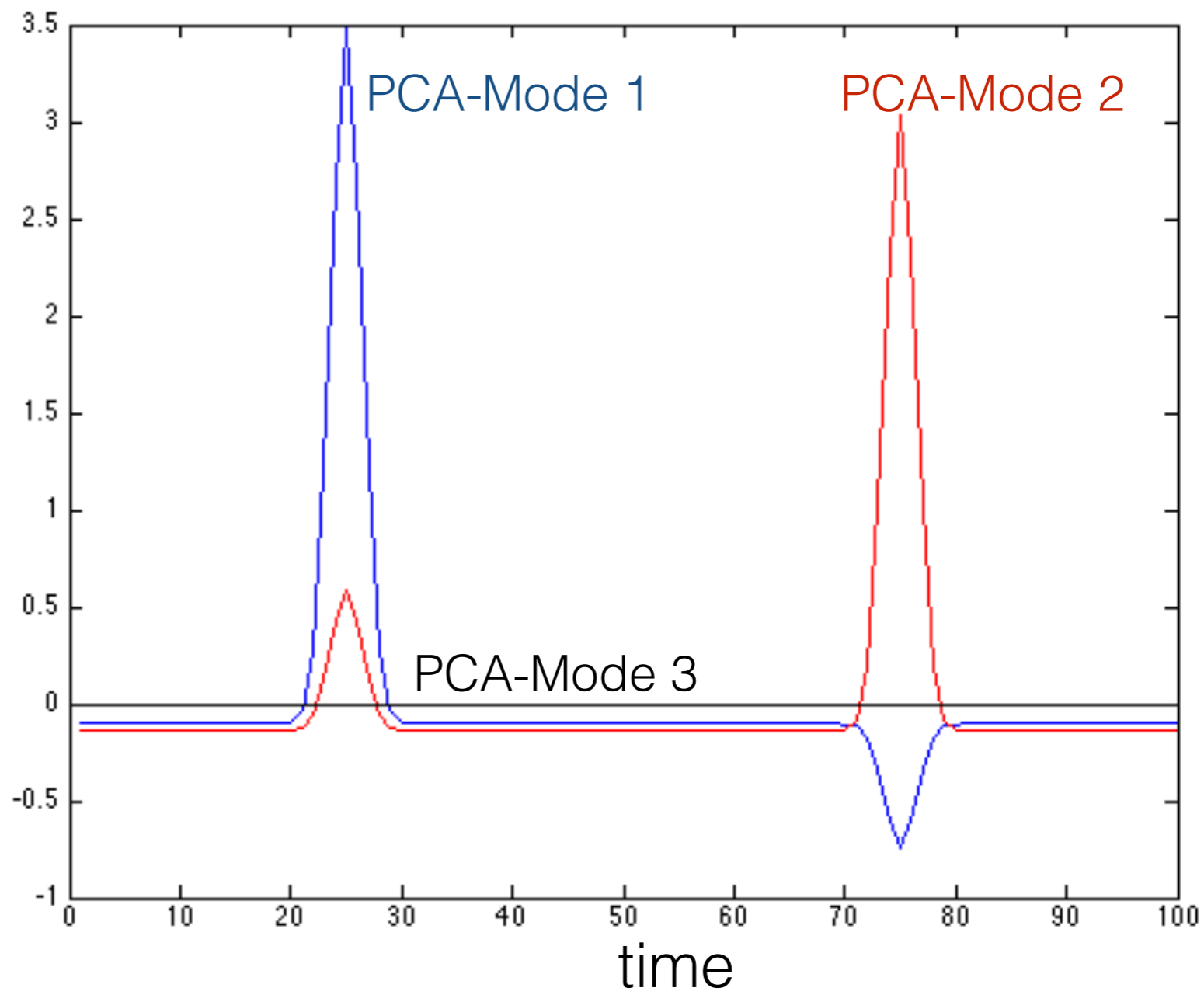
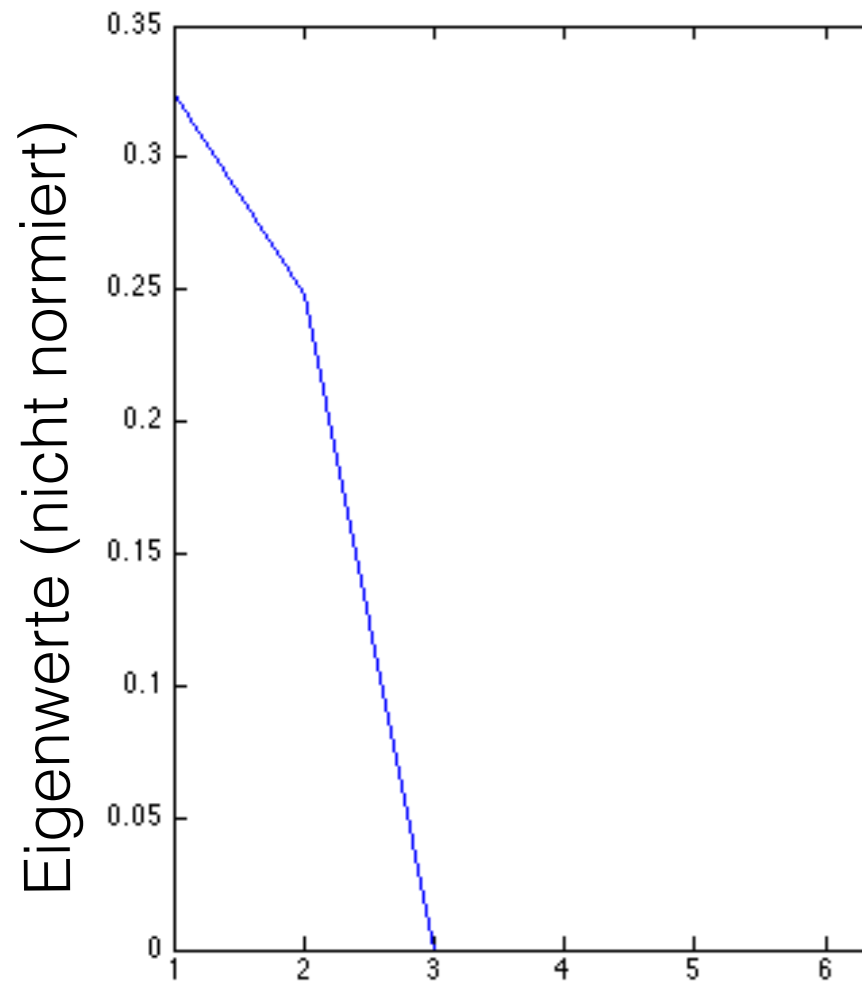
T: # time points = # modes

S: #samples , # trials



# Untermenge der simulierten Daten







## Anwendung auf reale spike-Daten:

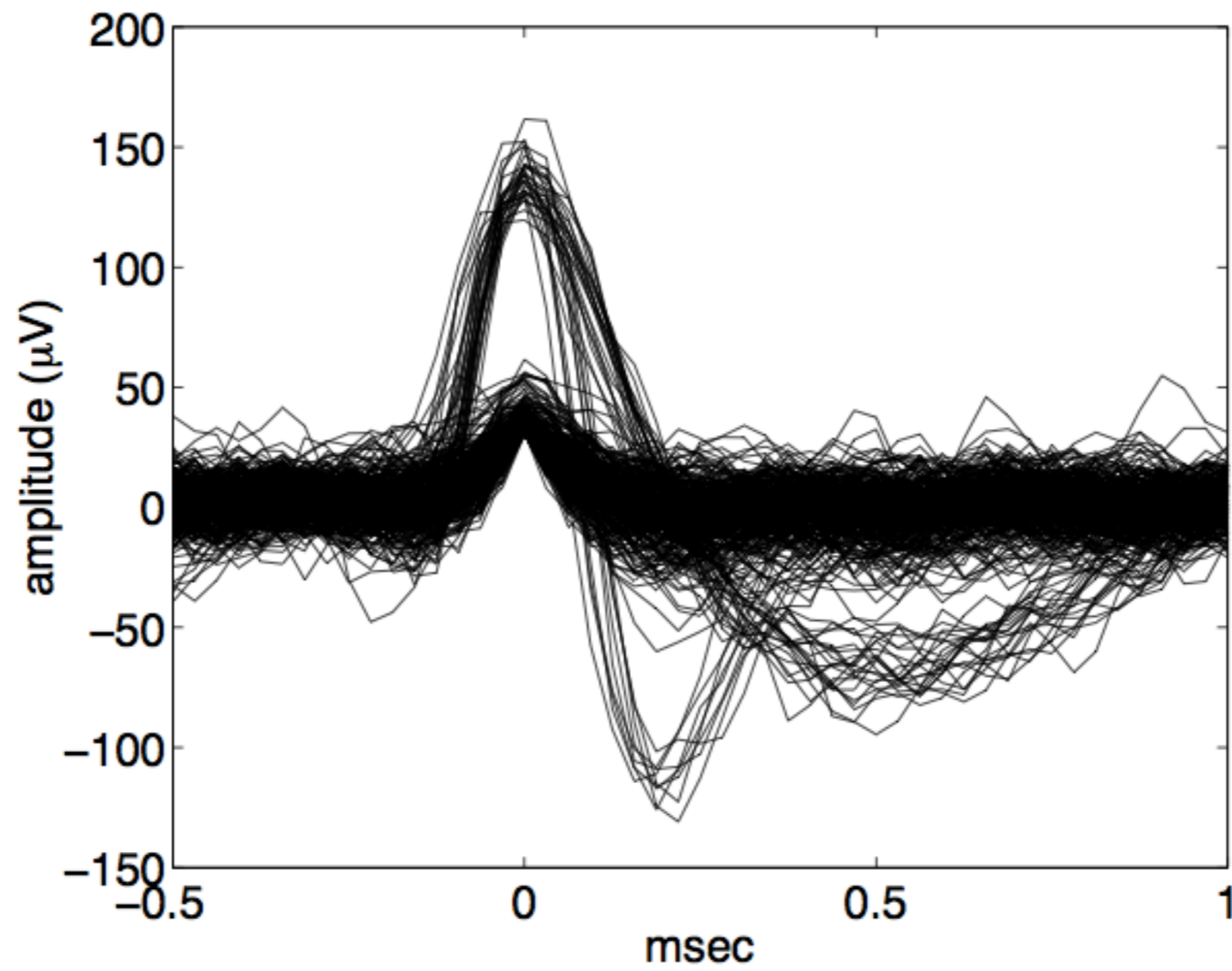
$$\mathbf{s}(n) = \sum_{i=1}^T c_i \mathbf{s}_i(n) , \quad n = 1, \dots, S , \quad \mathbf{s} \in \mathbb{R}^T$$

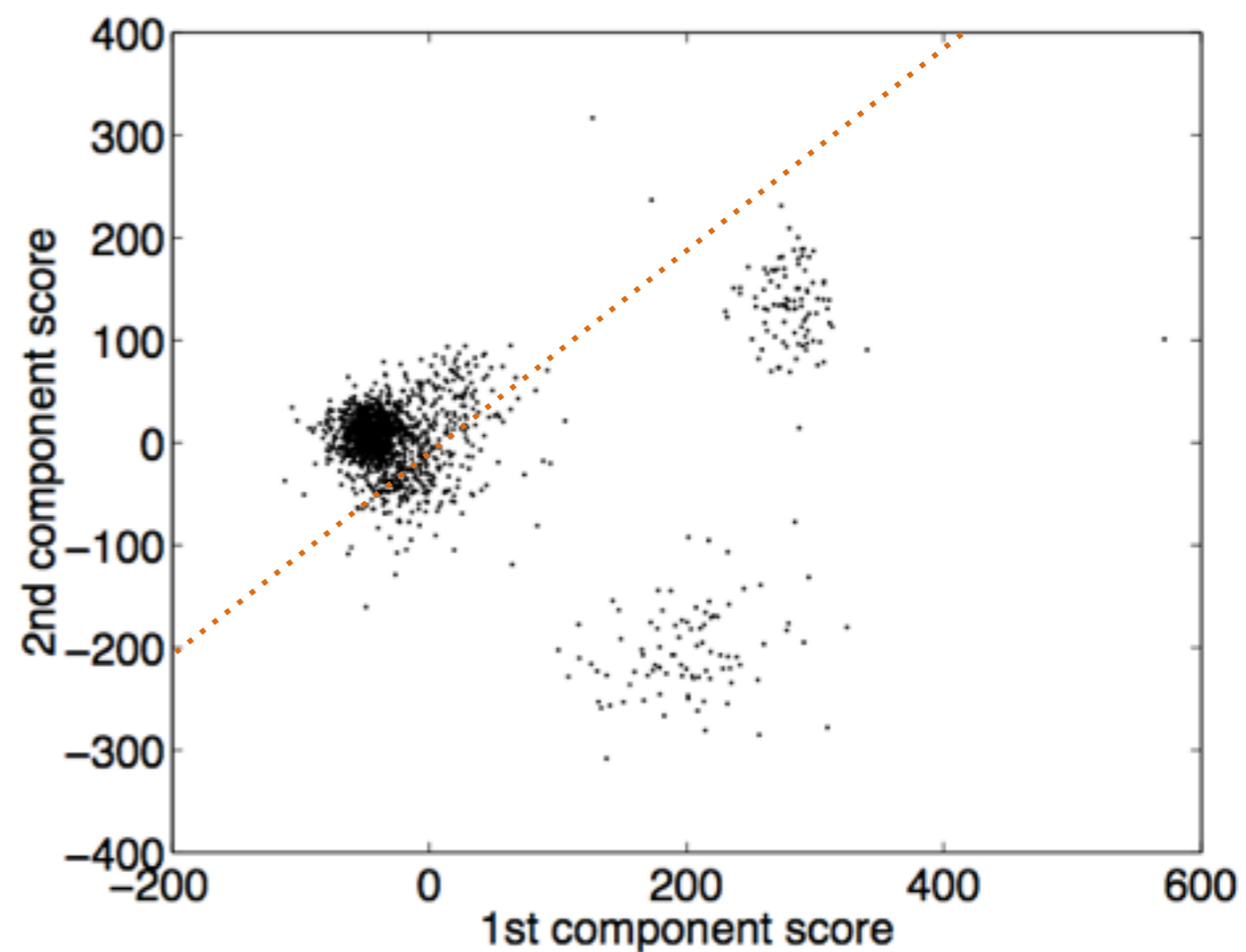
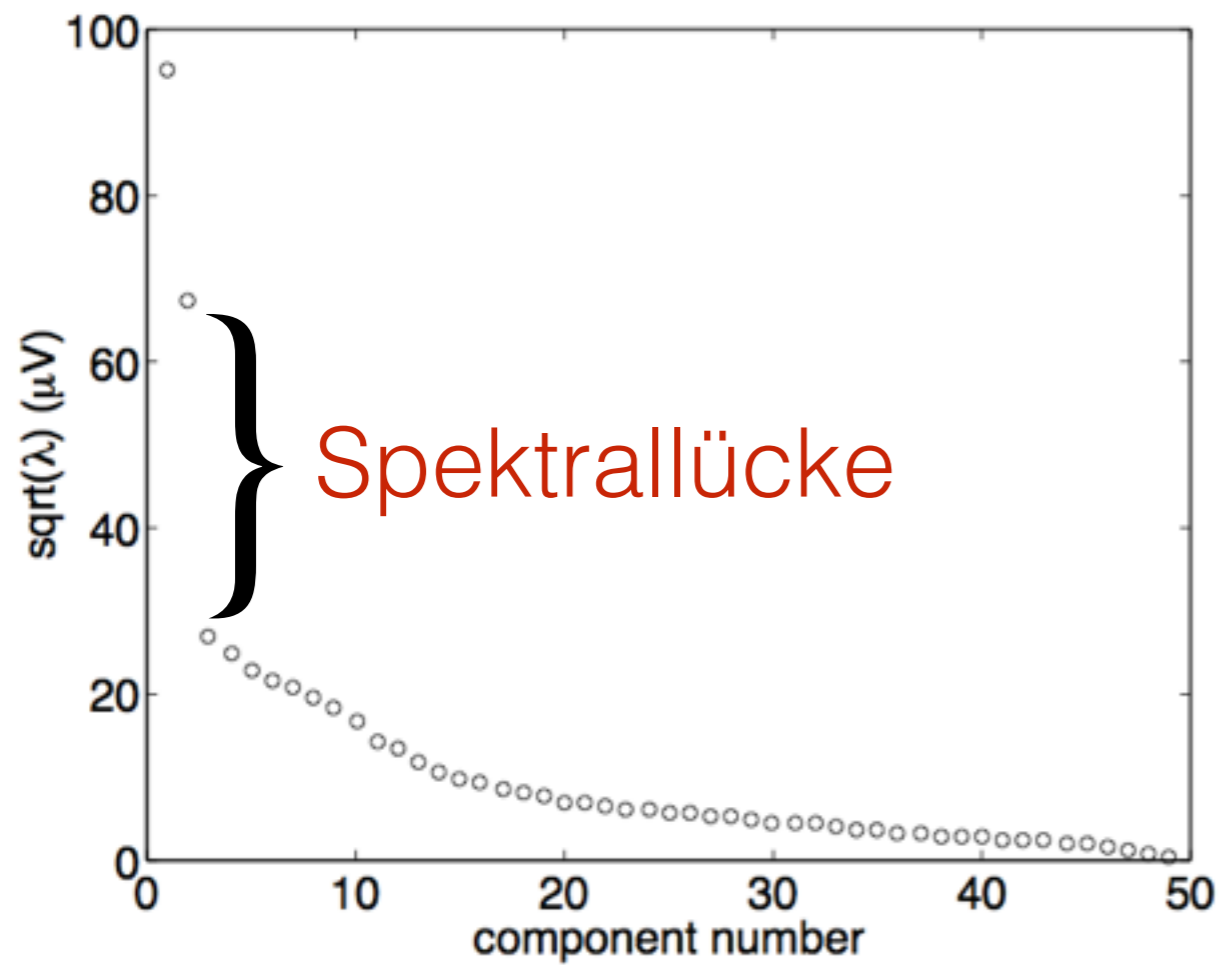
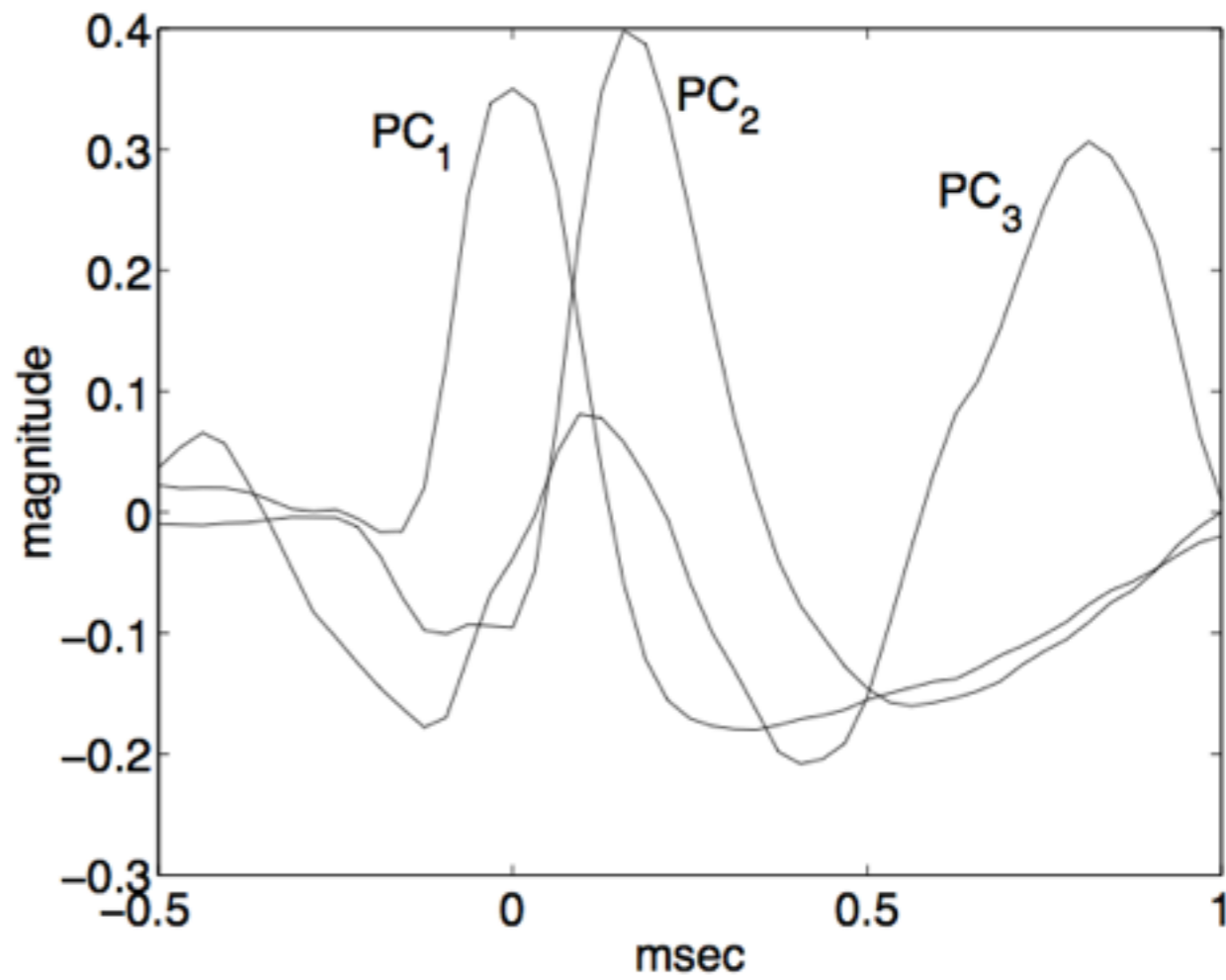
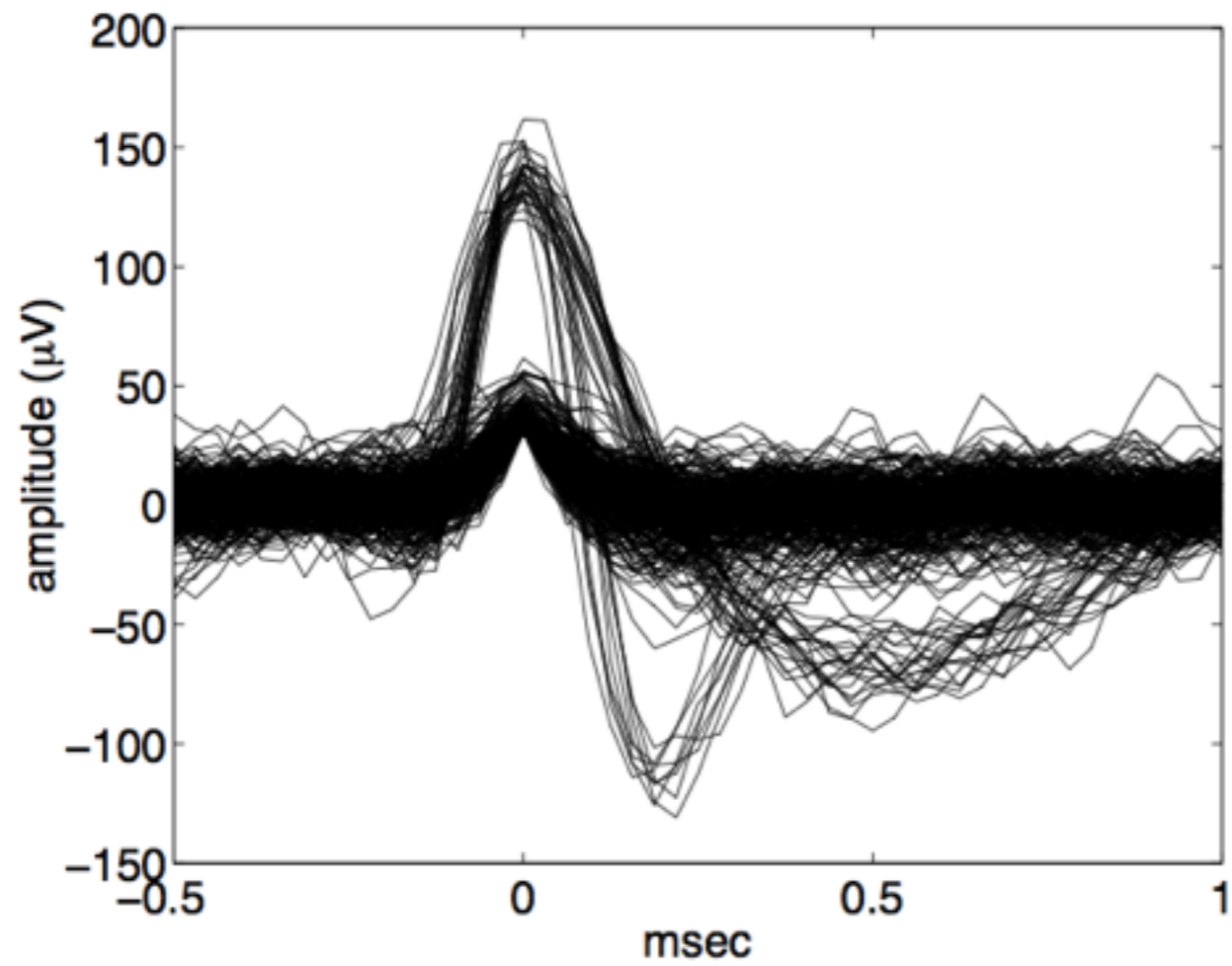
↑

time vector of spike trains

T: # time points = # modes

S: #samples , # trials





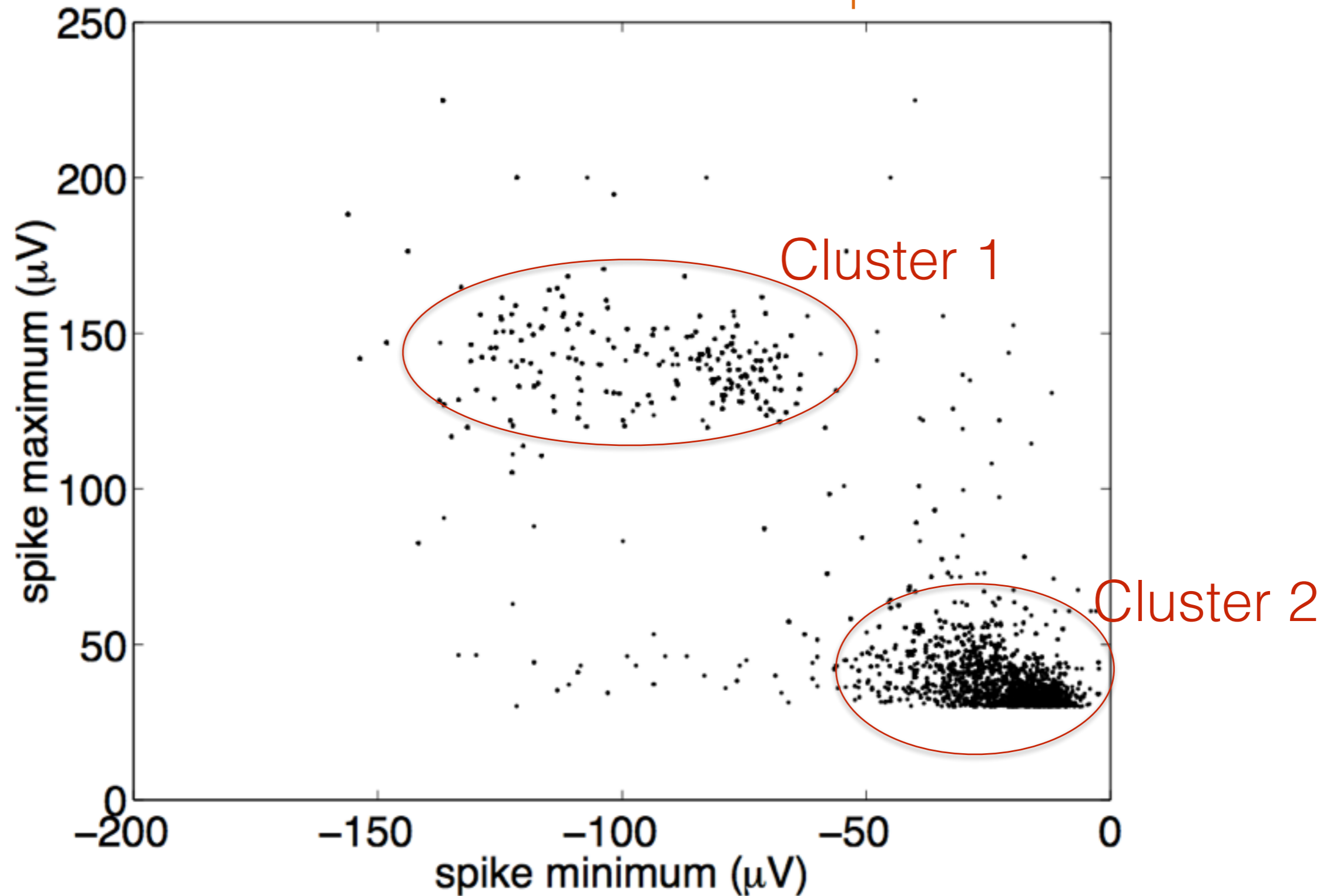
### 3. Aufgabe: Trennung von Neuronen

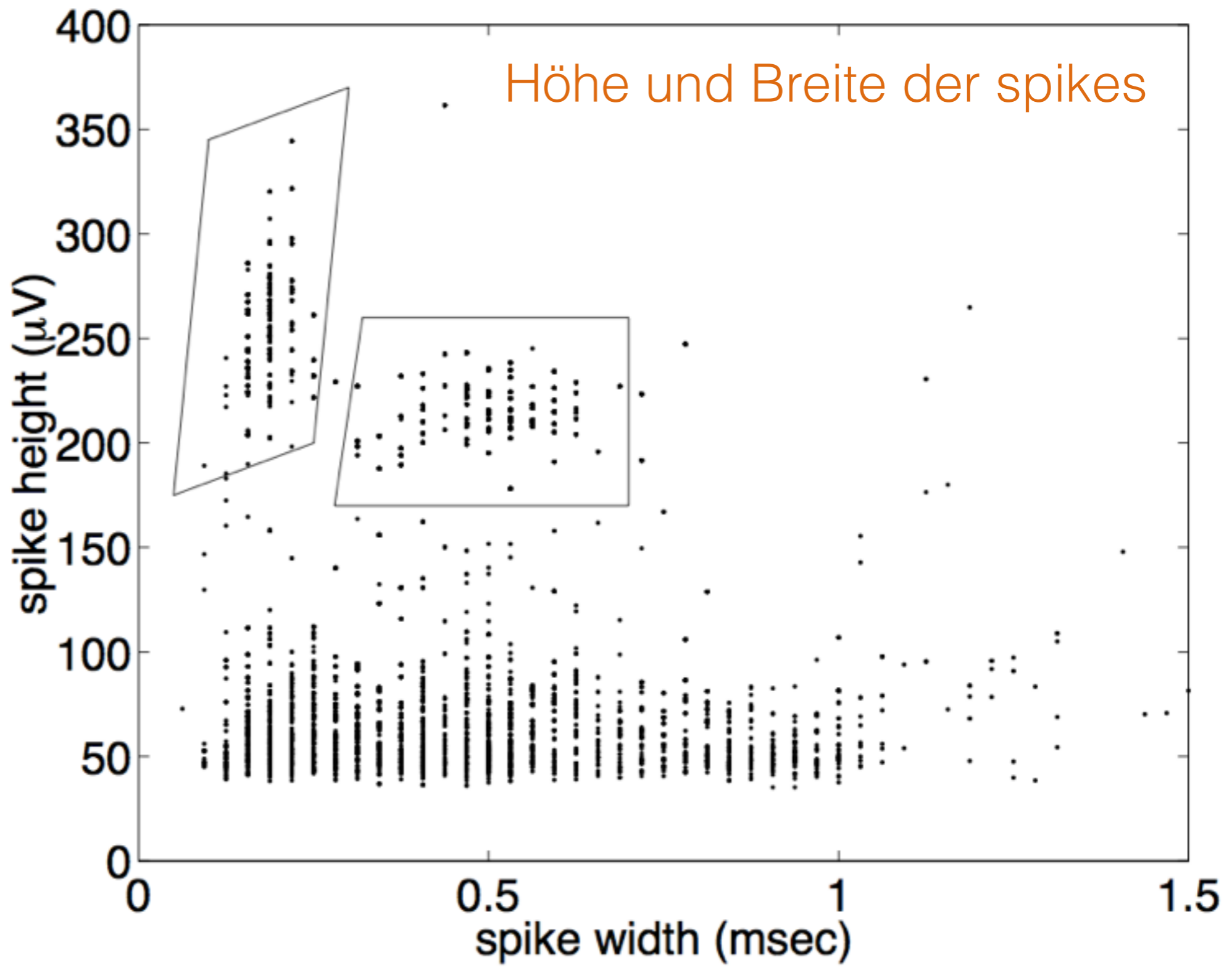
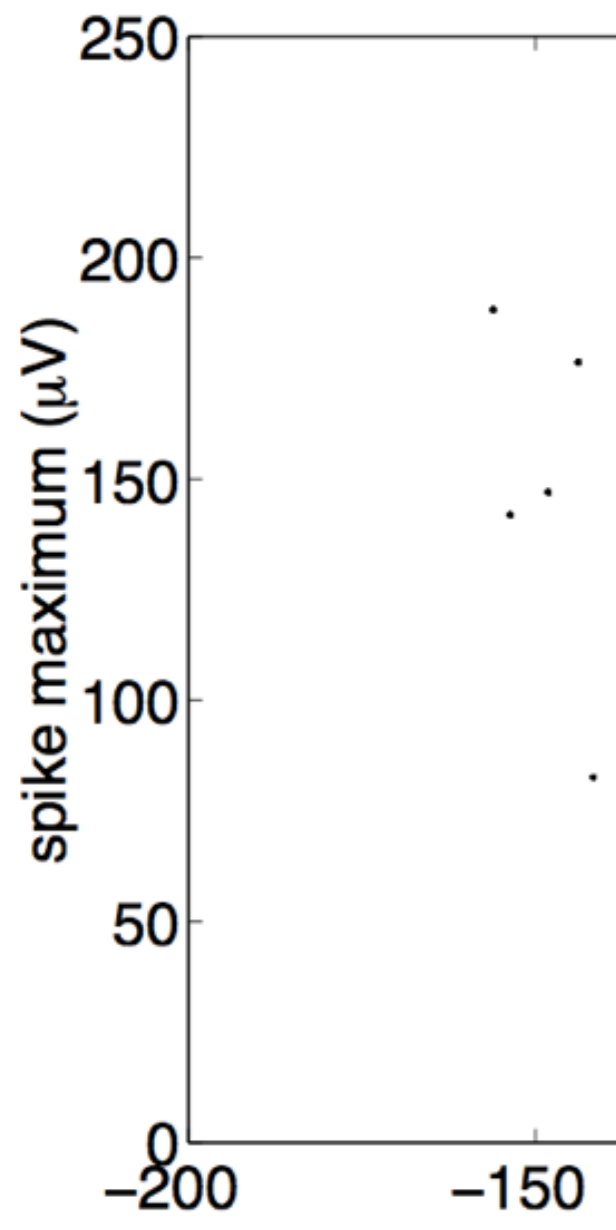
- Principal Component Analysis
- Clusteranalyse

Wie kann man die spike-Formen vergleichen ?

# Wie kann man die spike-Formen vergleichen ?

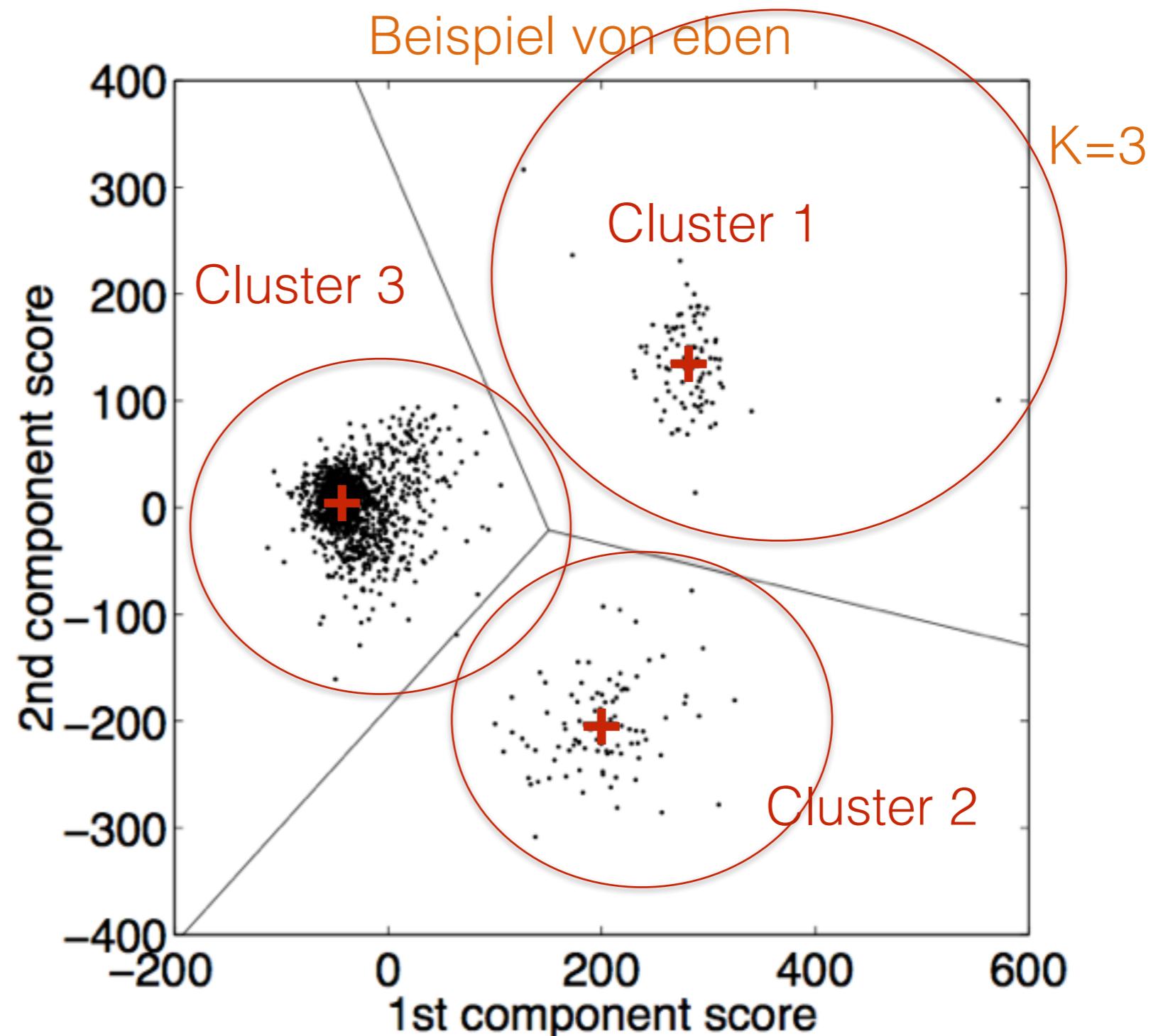
Maximum und Minimum der spikes





# einfachste Methode: k-means / nearest-neighbour

finde K cluster so, dass Abstand von Cluster-Zentren zu Datenpunkten minimal ist





....k-means / nearest-neighbour

Vorteil:

- einfache Implementierung, numerisch schnell

Nachteil:

- Anzahl von Clustern  $K$  muss gewählt werden
- Cluster sind Sphären
- *hard membership*:  
Datenpunkt gehört zu Cluster oder nicht

## effizientere Methode: Bayes'sche Klassifikation

- finde die wahrscheinlichste Klassifizierung mit K Clustern

$$p(x|c_k, \mu_k, \Sigma_k)$$

W. für Datum  $x$ , wenn  $x$  zum Cluster  $k$  mit gehört mit Wahrscheinlichkeit  $c_k$ . Cluster  $k$  hat den Mittelwert  $\mu$  und die Kovarianz  $\Sigma$

$\theta_k = \{\mu_k, \Sigma_k\}$  steht für die Clustereigenschaften von Cluster  $k$

$$p(x|\theta_{1:K}) = \sum_{k=1}^K p(x|c_k, \theta_k) p(c_k) \quad \sum_{k=1}^K p(c_k) = 1$$

W. für Datum  $x$  für einen Satz von bestimmten K Clustern

## ... Bayes'sche Klassifikation

$$p(c_k|x, \theta_{1:K})p(x|\theta_{1:K}) = p(x|c_k, \theta_{1:K})p(c_k)$$



gilt allgemein

W. für  $c_k$  falls  $x$  in Cluster  $k$  liegt; gesuchte Größe

$$p(c_k|x, \theta_{1:K}) = \frac{p(x|c_k, \theta_{1:K})p(c_k)}{p(x|\theta_{1:K})}$$

Bayes' Regel

... Bayes'sche Klassifikation

$$p(x_{1:N}|\theta_{1:K}) = \prod_{n=1}^N p(x_n|c_k, \theta_{1:K})$$

↑  
W. für den ganzen Datensatz für einen Satz  
von Clusterparameter

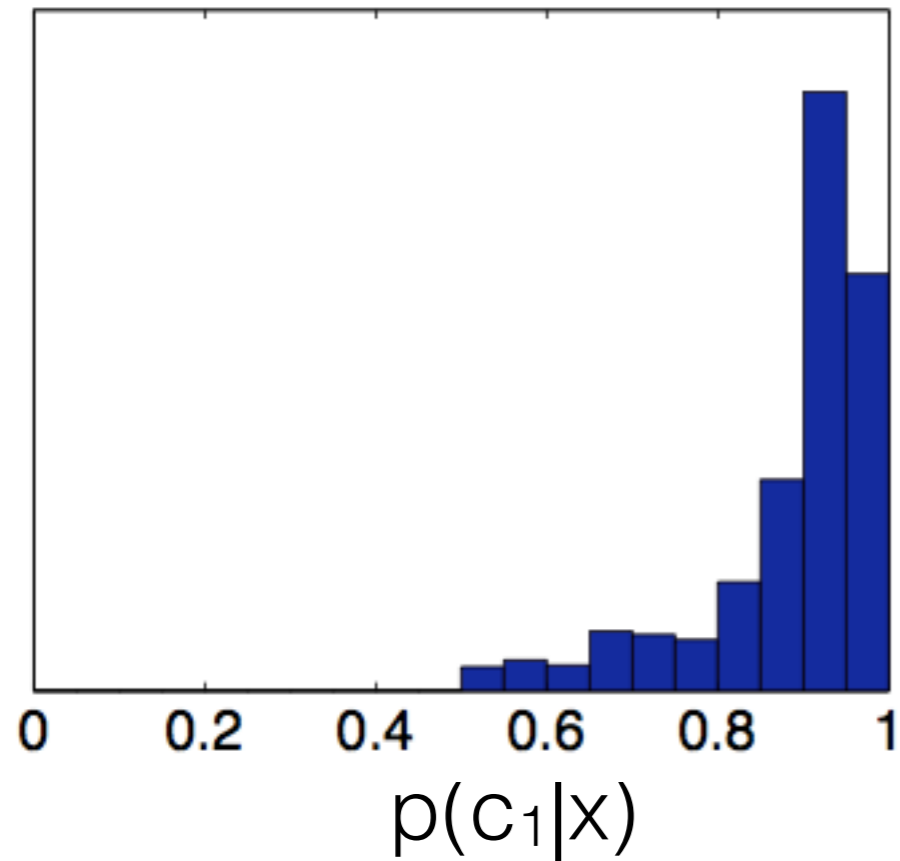
Maximieren von  $p(x_{1:N}|\theta_{1:K})$  ergibt optimalen Satz  
von Clusterparametern und somit mit Bayes' Regel  
**optimale Clusterwahrscheinlichkeiten**

Um Clustergrenzen mittels  $p(c_k|x, \theta_{1:K})$  zu bestimmen,  
benötigt man noch die Konfidenz für die Resultate, also  
eine künstliche Schwelle.

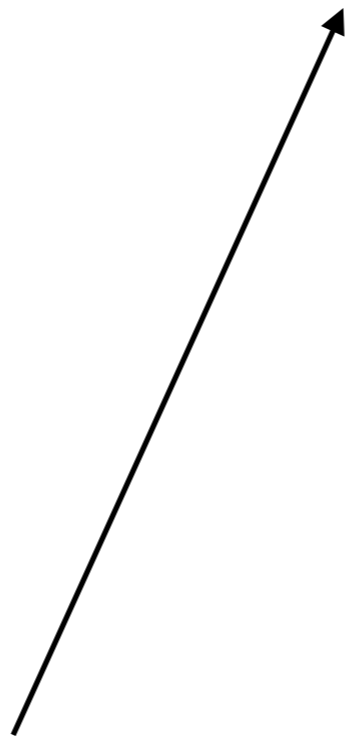
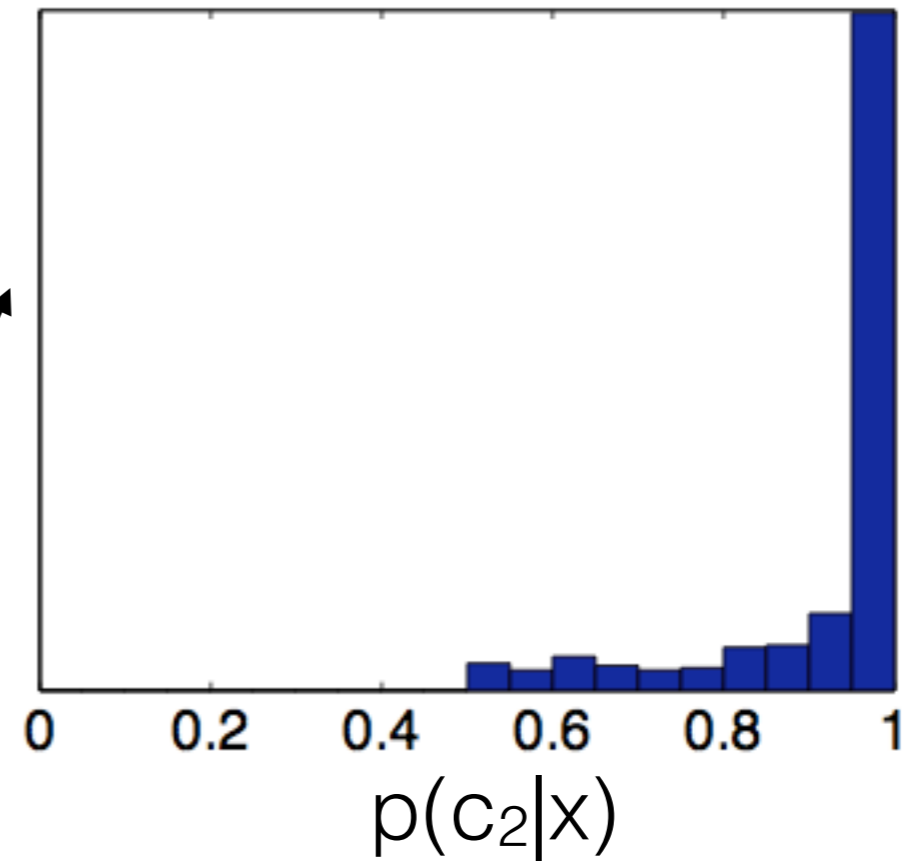
Anwendung auf reale spike-Daten  
von gerade eben

# statistisches Ergebnis

## Cluster 1



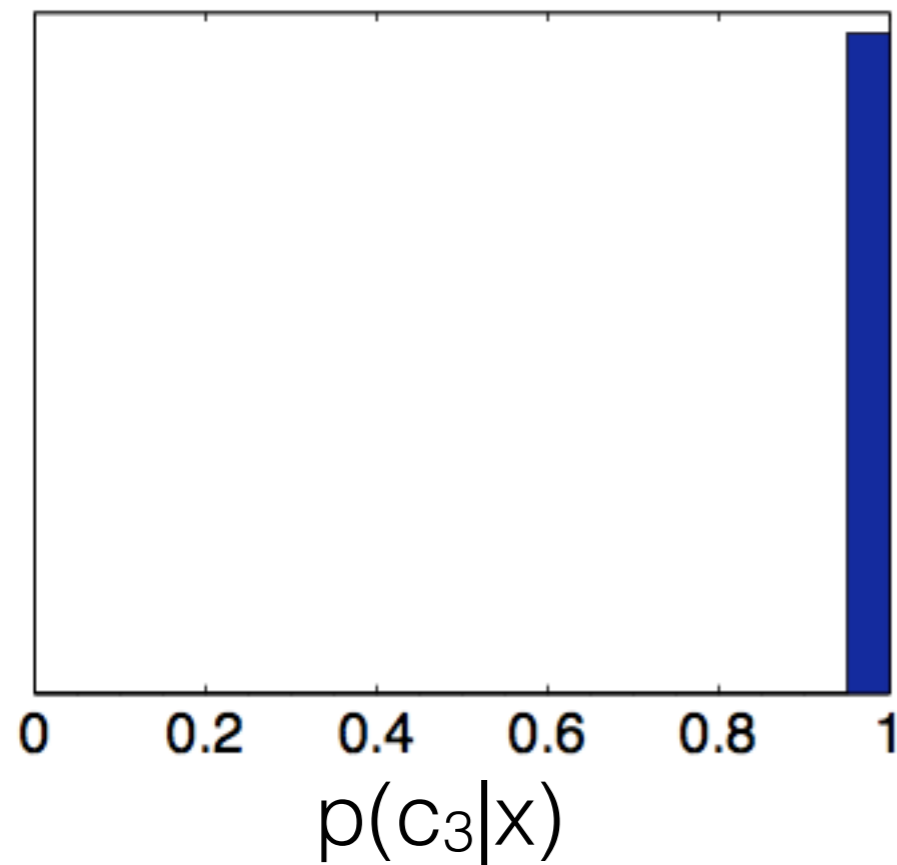
## Cluster 2



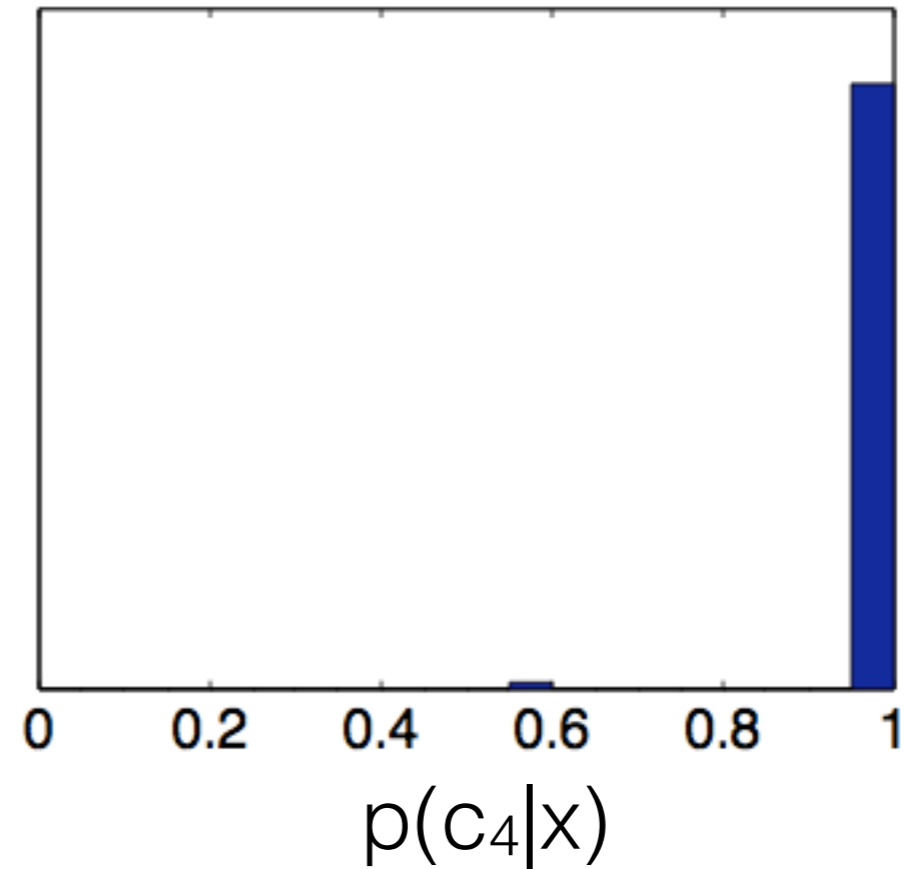
Anzahl von Datenpunkten, die mit Wahrscheinlichkeit  $p(c_k|x)$  zum Cluster  $k$  gehören

.... statistisches Ergebnis

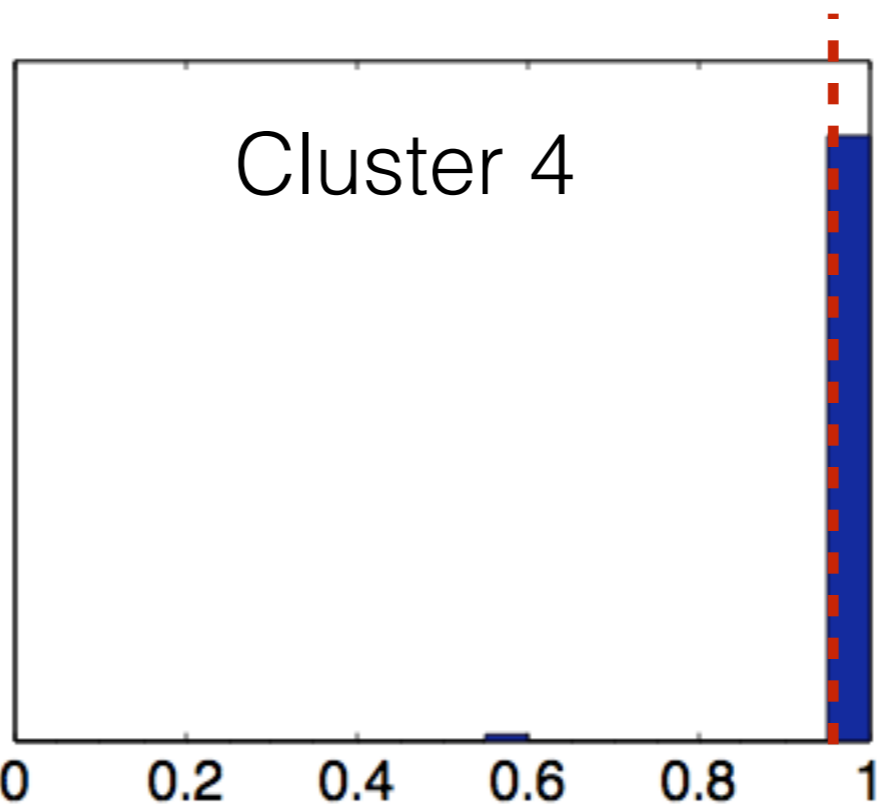
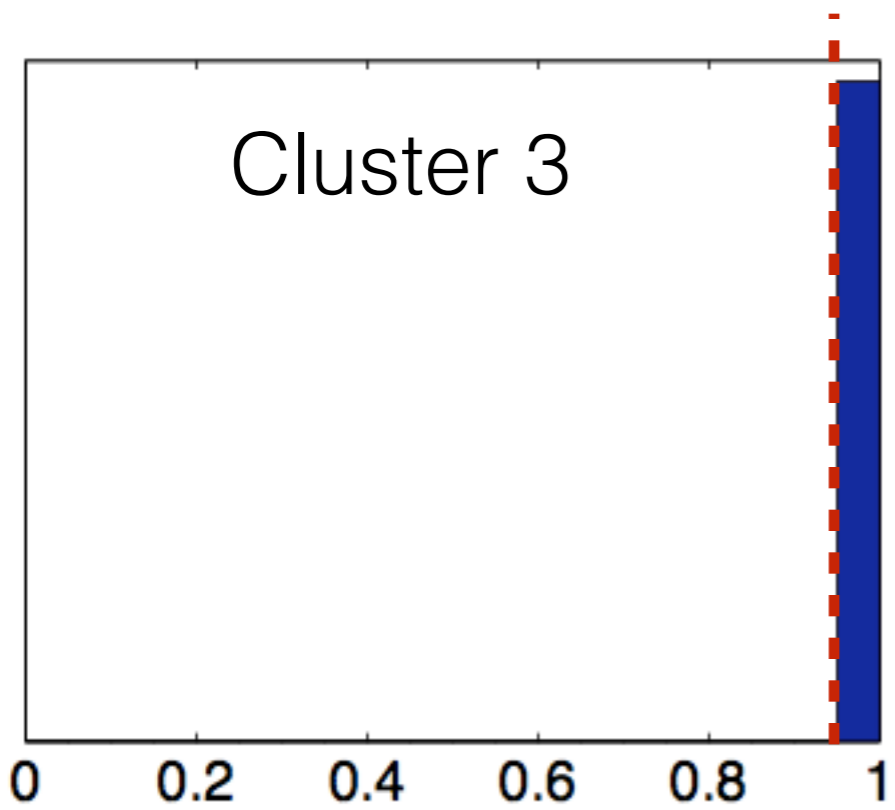
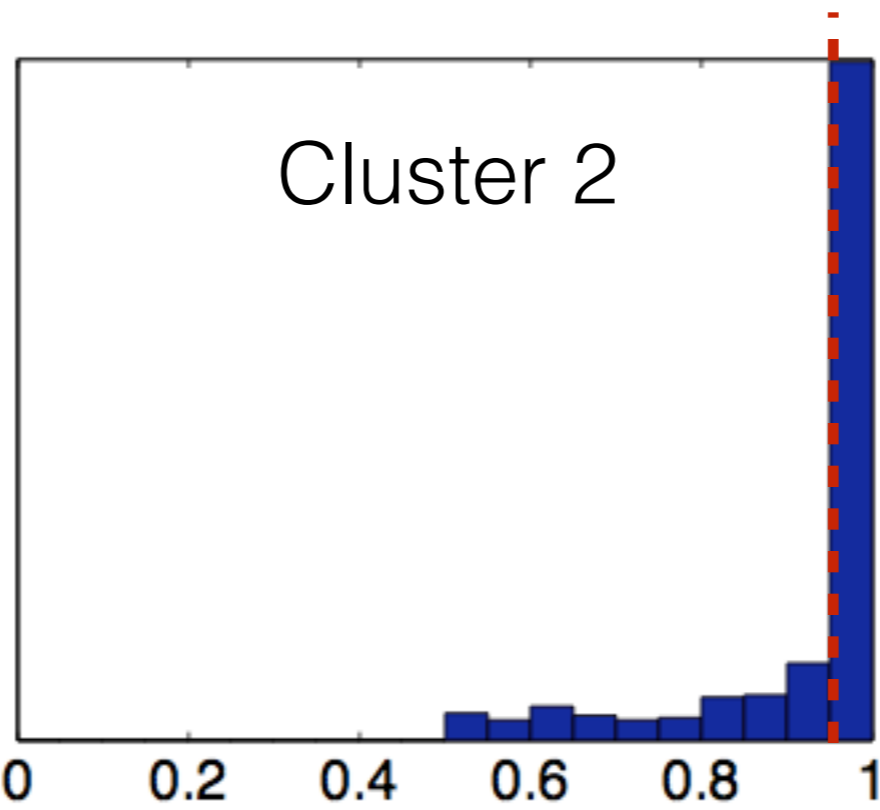
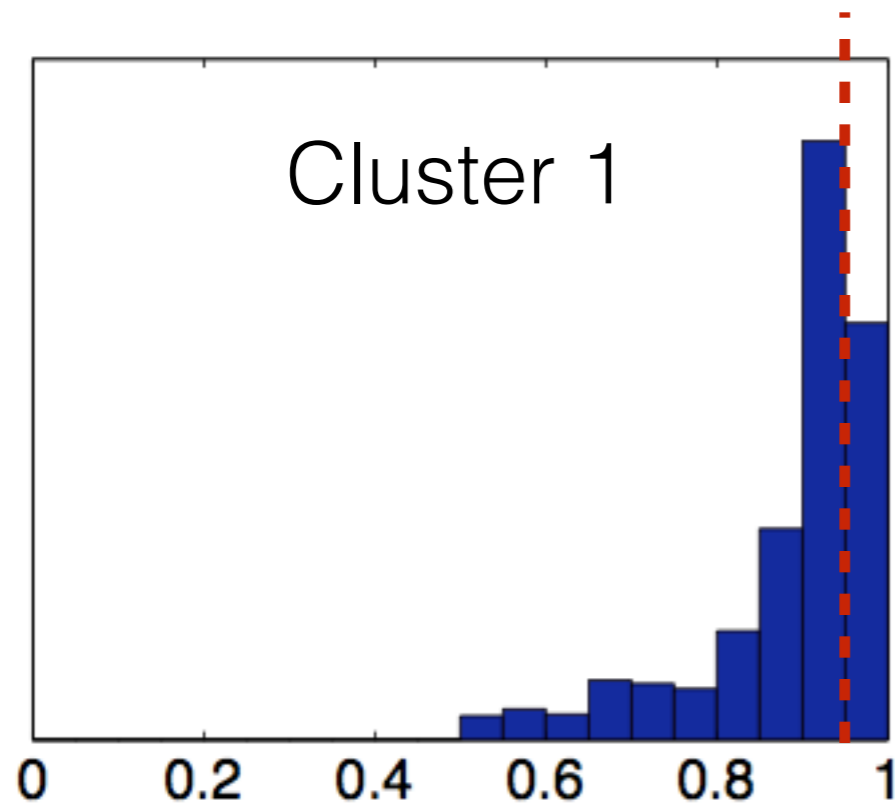
Cluster 3



Cluster 4

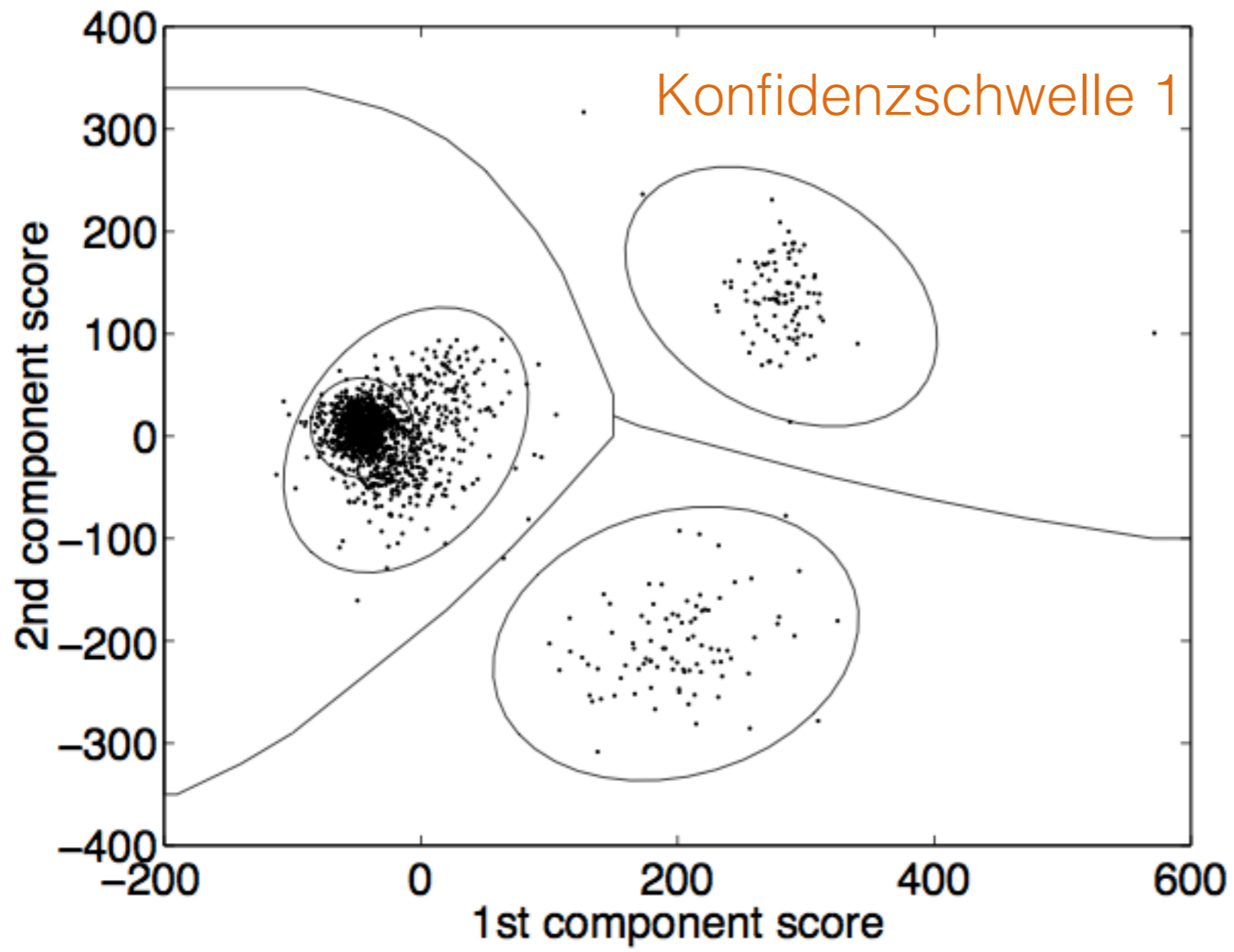


Die Datenpunkte hier gehören mit fast-Sicherheit zum jeweiligen Cluster 3 oder 4.

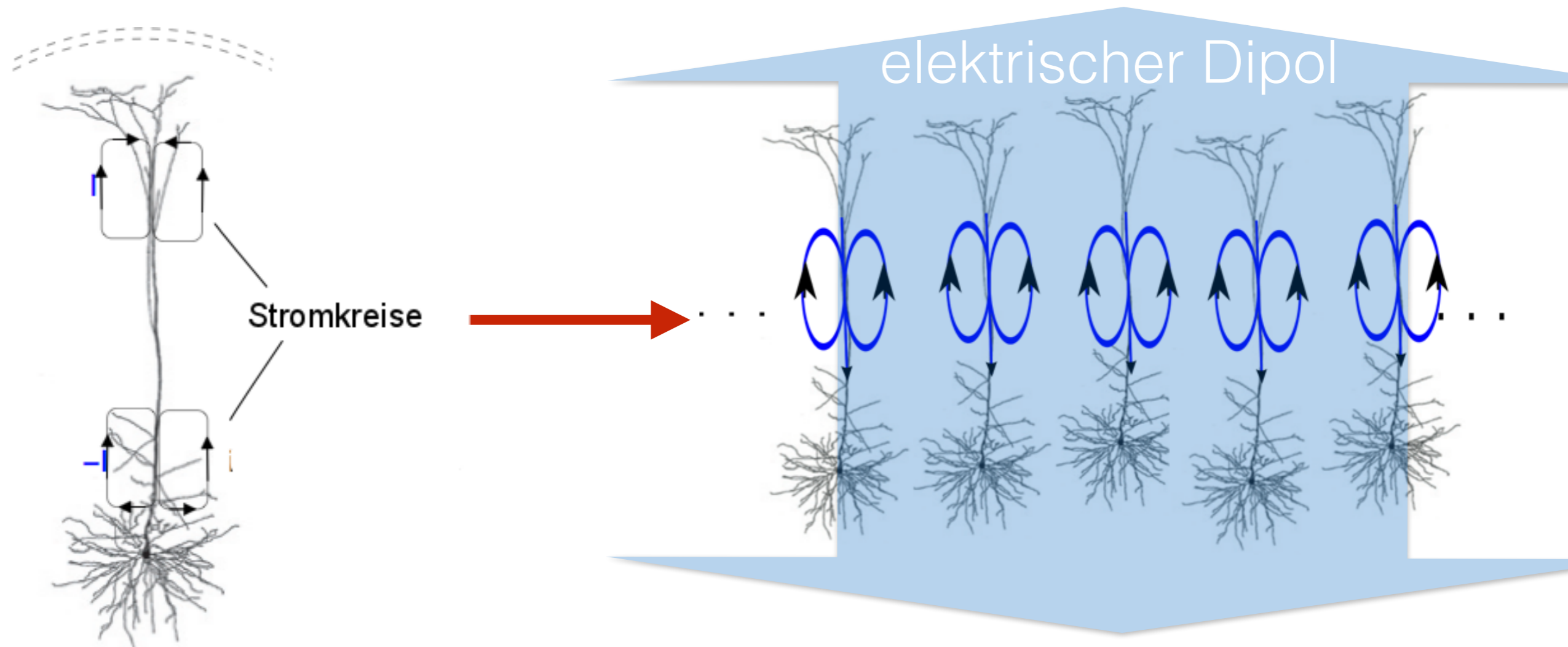


Beispiel:  
Konfidenz 95%





# Wechselwirkung zwischen Neuronen ?



**Axonale und elektromagnetische WW  
führen zur Synchronisation**

nun zu den Lokalen Feldpotentialen.....

# Kann man synaptische Stromquellen trennen ?

ja, mittels **current-source density estimation**

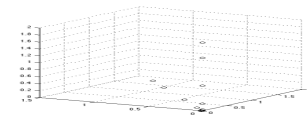
Elektrodynamik des neuronalen Gewebes:

$$\nabla \cdot \mathbf{j} + \frac{\partial \rho}{\partial t} = 0$$

Ladungserhaltung

$$\mathbf{j} = \sigma \mathbf{E}$$

Ohmscher Leiter



↑  
LFP

für homogenes Gewebes:

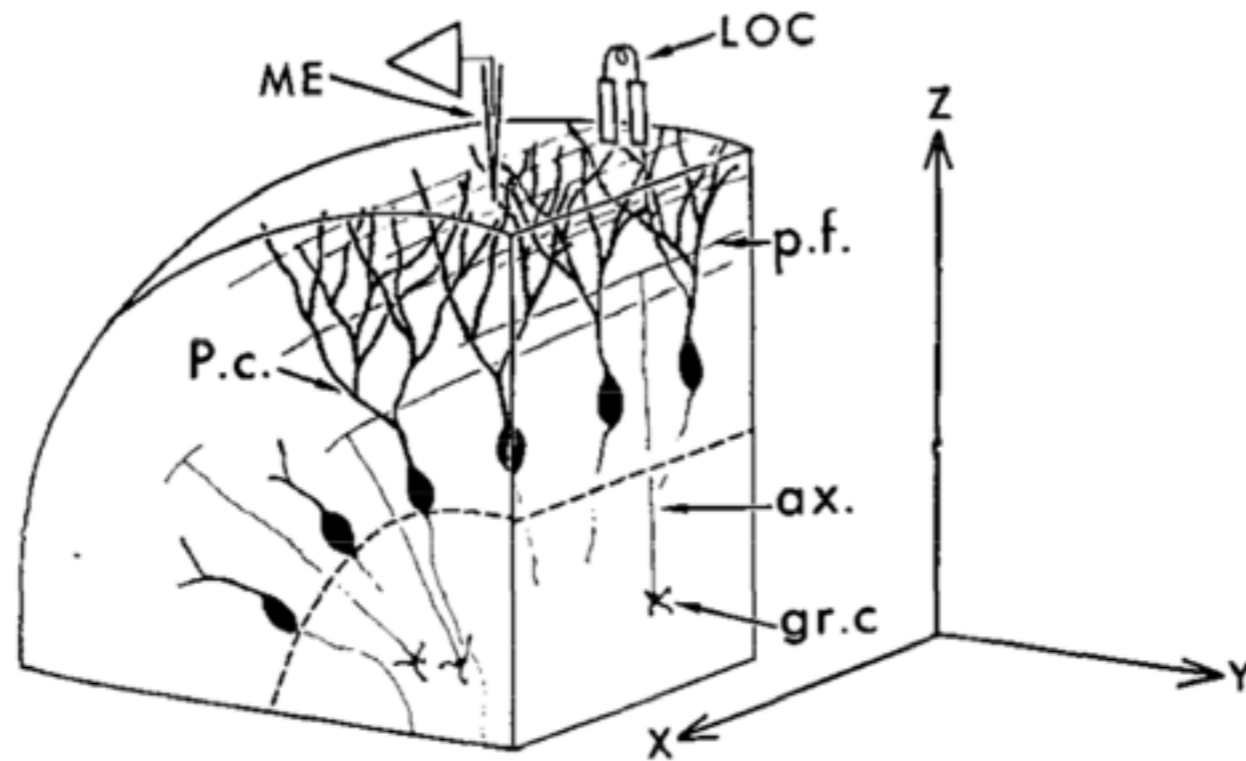
$$\nabla \mathbf{j} = \sigma \nabla \mathbf{E} = -\sigma \Delta V$$



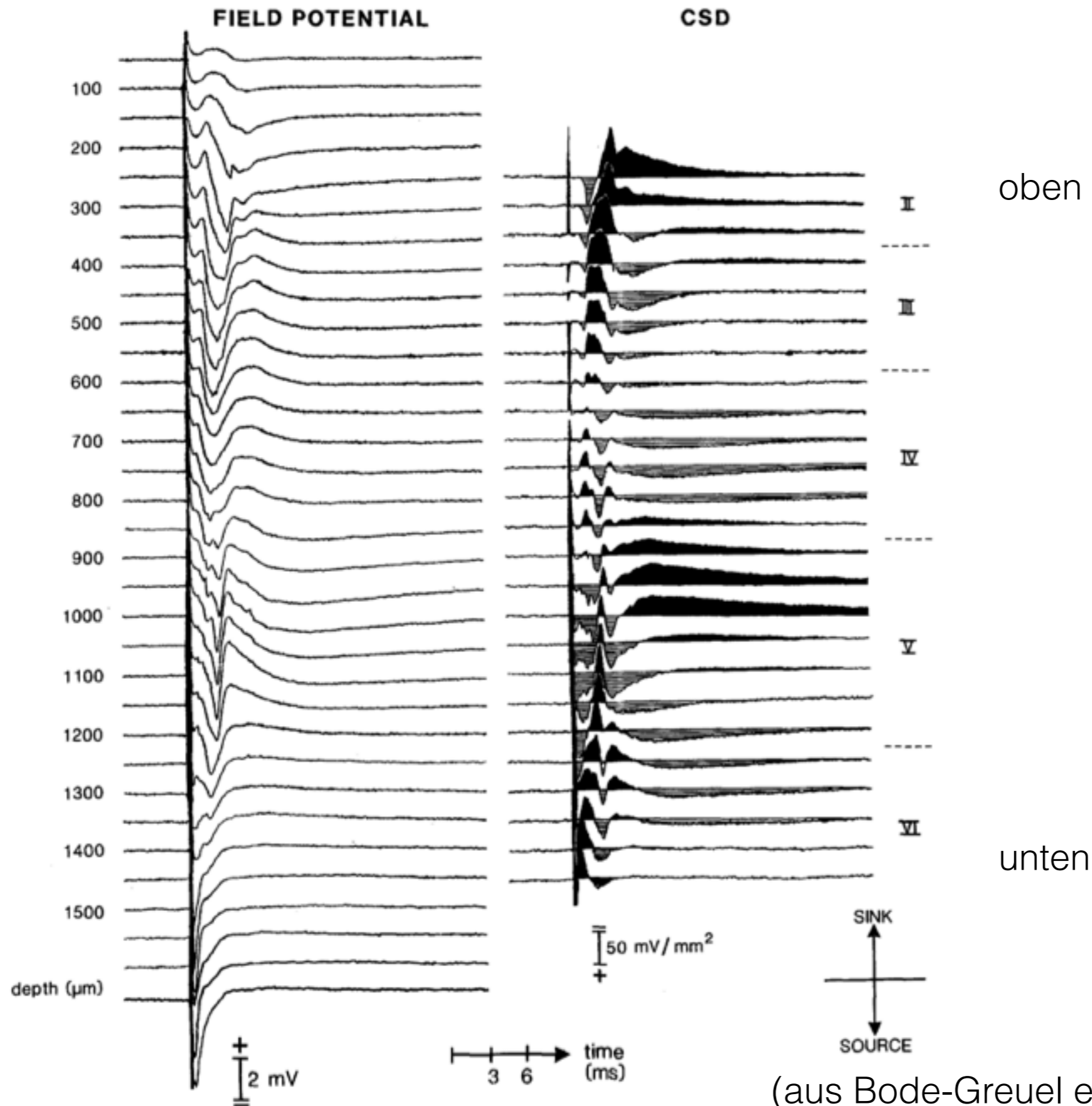
Gradient des Stromflusses ~ current source density (CSD)

CSD:

$$I_m(\mathbf{x}, t) = -\sigma \left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \right) V(\mathbf{x}, t)$$



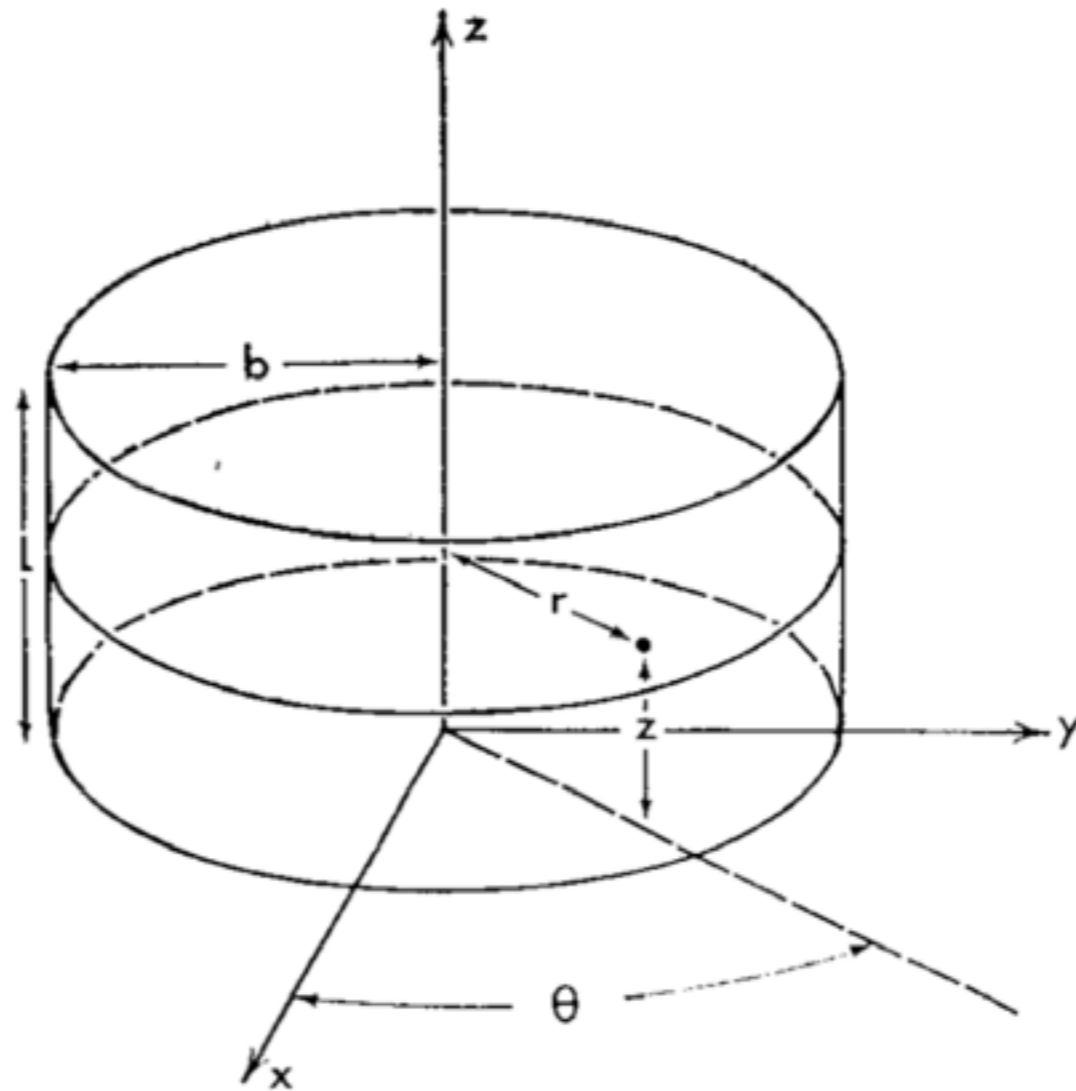
# Typische Messung: laterale Elektroden entlang der vertikalen Achse



gemessen im  
visuellen Cortex der Katze

(aus Bode-Greuel et al., Exp. Brain Res. (1987))

Beispiel:

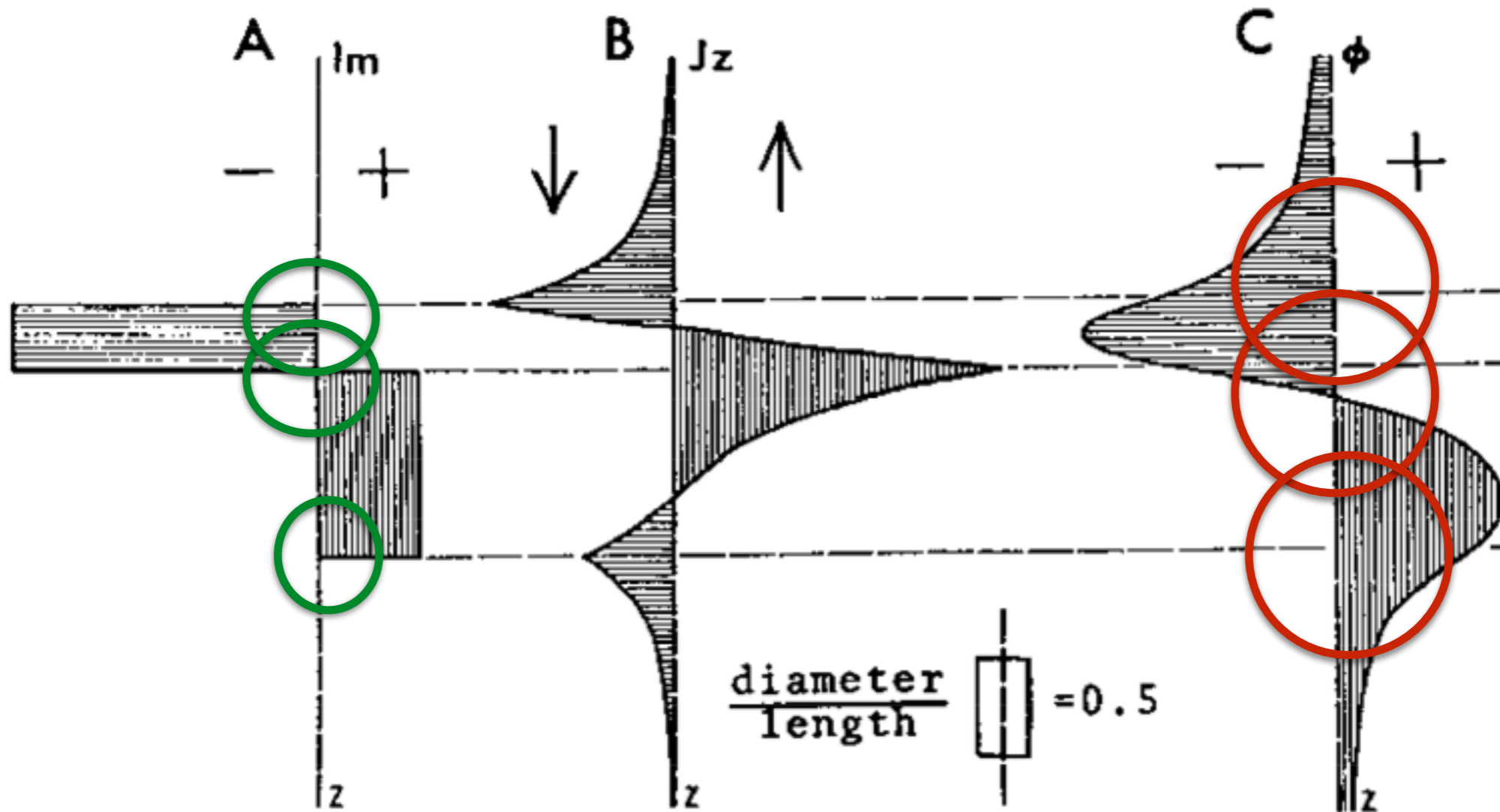


Zylindergeometrie

$$\left( \frac{\partial^2}{\partial r^2} + \frac{1}{r} \frac{\partial}{\partial r} + \frac{1}{r^2} \frac{\partial^2}{\partial \phi^2} + \frac{\partial^2}{\partial z^2} \right) V(r, \phi, z, t) = - \frac{I_m(r, \phi, z, t)}{\sigma}$$

$$I_m = +1 \quad 0 < z < 0.75L \quad 0 < r < b$$

$$= -3 \quad 0.75L < z < L \quad 0 < r < b$$



(aus Nicholson and Freeman, J. Neurophysiol. (1975))



Man kann also den **Ort** der **synaptischen Stromquellen schätzen**, doch **räumliche Auflösung** ist **schlecht**.

Man kann die **Anzahl** der **spikenden Neuronen schätzen**, doch **keine räumliche Lokalisierung** möglich.

jetzt kann man also berechnen:

- **Synchronisation** zwischen einzelnen Neuronen
- **Synchronisation** zwischen einzelnen spike trains und den LFPs

**man kann nur die Summe aller synaptischen Ströme betrachten (LFP)**