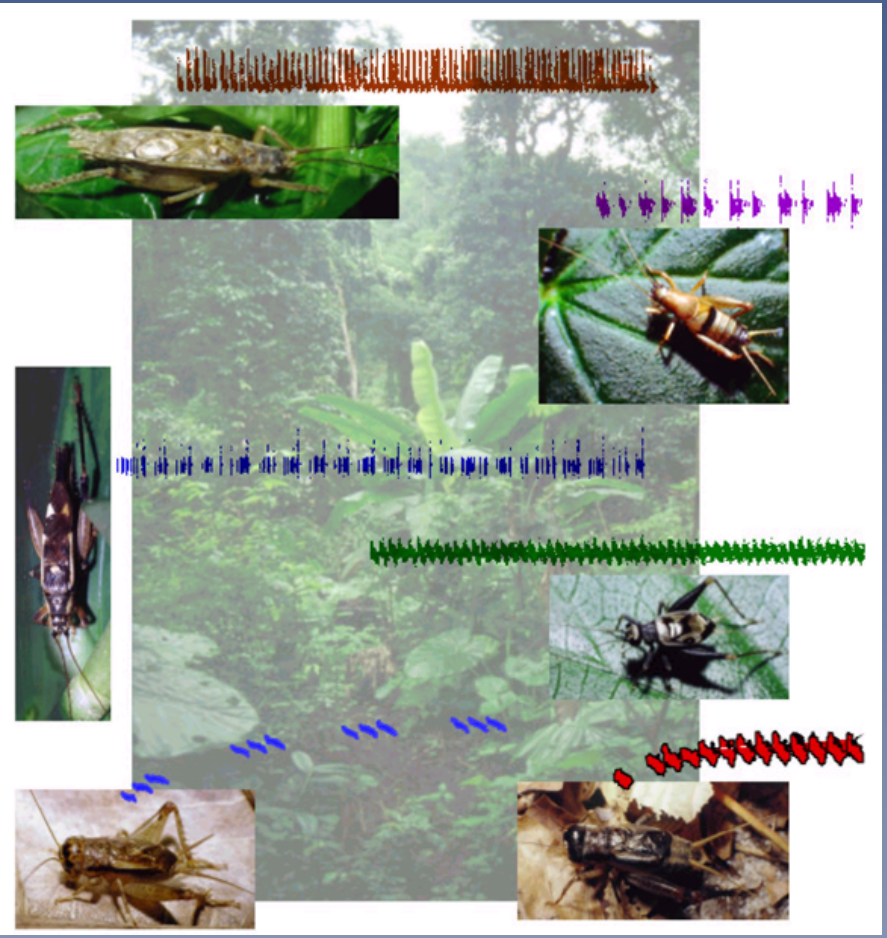
Temporale Fusion neuronaler Netze zur Klassifikation bioakustischer Zeitreihen



Christian Dietrich

Gliederung

- **Neuroinformatik**
  - ⋆ Allgemeines
  - ⋆ RBFs und deren Initialisierung
- **Bioakustik der Orthopteren**
  - ⋆ Lautäußerung (Stridulation)
  - ⋆ Signalstruktur
  - ⋆ Ablauf der vollautomatischen Klassifikation
  - ⋆ Anwendungsbereiche und Motivation
- **Mehrklassifikatorsysteme und Zeitreihenklassifikation**
  - ⋆ Statische Ansätze
  - ⋆ Adaptive Ansätze (Ensemble Training)

Informatik - Neurobiologie

Informatik - Neurobiologie

- Informatik
  - ⋆ John von Neumann (1903-1957) Konzept: Sequentielles Konzept eines universellen, programmgesteuerten Automaten

Informatik - Neurobiologie

- Informatik
  - ⋆ John von Neumann (1903-1957) Konzept: Sequentielles Konzept eines universellen, programmgesteuerten Automaten
  - ⋆ Parallele Systeme: Diese bestehen aus wenigen parallel arbeitenden Prozessoren

Informatik - Neurobiologie

- Informatik
  - ⋆ John von Neumann (1903-1957) Konzept: Sequentielles Konzept eines universellen, programmgesteuerten Automaten
  - ⋆ Parallele Systeme: Diese bestehen aus wenigen parallel arbeitenden Prozessoren
- Neurobiologie
  - ⋆ Massiv parallel arbeitende Gebilde: Das menschliche Gehirn besteht aus 10^{10} Neuronen ohne Programmierungsprobleme, Synchronisations-Probleme, Scheduling-Probleme und Protokollebenen.

Informatik - Neurobiologie

- Informatik
  - ⋆ John von Neumann (1903-1957) Konzept: Sequentielles Konzept eines universellen, programmgesteuerten Automaten
  - ⋆ Parallele Systeme: Diese bestehen aus wenigen parallel arbeitenden Prozessoren
- Neurobiologie
  - ⋆ Massiv parallel arbeitende Gebilde: Das menschliche Gehirn besteht aus 10^{10} Neuronen ohne Programmierungsprobleme, Synchronisations-Probleme, Scheduling-Probleme und Protokollebenen.
  - ⋆ Taktfrequenz: Die einzelnen biologischen Verarbeitungselemente haben eine 100.000 kleinere Taktfrequenz als moderne Mikroprozessoren.

Informatik - Neurobiologie

- Informatik
  - ⋆ John von Neumann (1903-1957) Konzept: Sequentielles Konzept eines universellen, programmgesteuerten Automaten
  - ⋆ Parallele Systeme: Diese bestehen aus wenigen parallel arbeitenden Prozessoren
- Neurobiologie
  - ⋆ Massiv parallel arbeitende Gebilde: Das menschliche Gehirn besteht aus 10^{10} Neuronen ohne Programmierungsprobleme, Synchronisations-Probleme, Scheduling-Probleme und Protokollebenen.
  - ⋆ Taktfrequenz: Die einzelnen biologischen Verarbeitungselemente haben eine 100.000 kleinere Taktfrequenz als moderne Mikroprozessoren.
  - ⋆ Robustheit: Ein neuronales System verkraftet den Ausfall von einzelnen Elementen.

Neuroinformatik

Neuroinformatik

- Biologische Neuroinformatik

Neuroinformatik

- Biologische Neuroinformatik
  - ⋆ Die grundlegende Funktionsweise des menschlichen Gehirns ist bekannt, also werden einzelne Funktionseinheiten des menschlichen Gehirns modelliert (Ionenströme der Axone und Dendriten im Verlauf der Zeit).

Neuroinformatik

- Biologische Neuroinformatik
  - ⋆ Die grundlegende Funktionsweise des menschlichen Gehirns ist bekannt, also werden einzelne Funktionseinheiten des menschlichen Gehirns modelliert (Ionenströme der Axone und Dendriten im Verlauf der Zeit).
  - ⋆ Das Anspruchsvollste, was zur Zeit mit neuronalen Netzen modelliert wird sind die menschlichen Peripherieleistungen: Bilderkennung, Sprachverarbeitung und Motoriksteuerung.

Neuroinformatik

- Biologische Neuroinformatik
  - ⋆ Die grundlegende Funktionsweise des menschlichen Gehirns ist bekannt, also werden einzelne Funktionseinheiten des menschlichen Gehirns modelliert (Ionenströme der Axone und Dendriten im Verlauf der Zeit).
  - ⋆ Das Anspruchsvollste, was zur Zeit mit neuronalen Netzen modelliert wird sind die menschlichen Peripherieleistungen: Bilderkennung, Sprachverarbeitung und Motoriksteuerung.
  - ⋆ Der Aufbau und die Funktionsweise der Sinnesorgane (Augen und Ohren) ist von besonderer Bedeutung!

Neuroinformatik

- Mustererkennung

Neuroinformatik

- Mustererkennung
  - ⋆ Die biologischen Neuronen werden stark vereinfacht und strukturiert.

Neuroinformatik

- Mustererkennung
  - ⋆ Die biologischen Neuronen werden stark vereinfacht und strukturiert.
  - ⋆ Mathematische Modelle aus der Funktionsapproximation, dem maschinellen Lernen und der Statistik dienen als Grundlage.

Neuroinformatik

- Mustererkennung
  - ⋆ Die biologischen Neuronen werden stark vereinfacht und strukturiert.
  - ⋆ Mathematische Modelle aus der Funktionsapproximation, dem maschinellen Lernen und der Statistik dienen als Grundlage.
  - ⋆ Ein KNN ist definiert durch die Netzwerkarchitektur, den internen Aufbau der Berechnungseinheiten und einen Lernalgorithmus.

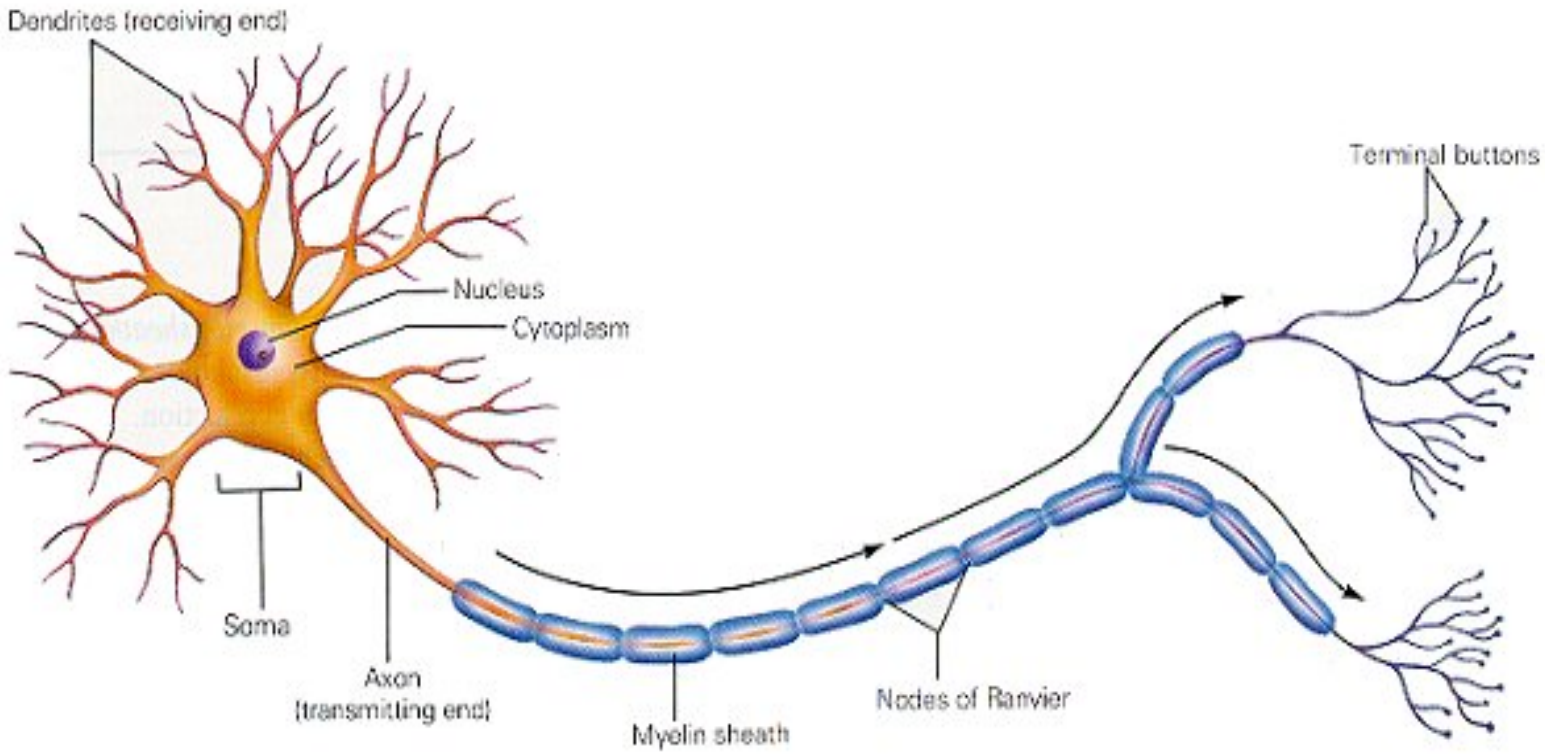
Neuroinformatik

- Mustererkennung
  - ⋆ Die biologischen Neuronen werden stark vereinfacht und strukturiert.
  - ⋆ Mathematische Modelle aus der Funktionsapproximation, dem maschinellen Lernen und der Statistik dienen als Grundlage.
  - ⋆ Ein KNN ist definiert durch die Netzwerkarchitektur, den internen Aufbau der Berechnungseinheiten und einen Lernalgorithmus.
  - ⋆ Anwendungsbereiche
    - Mustererkennung und Klassifikation
    - Funktionsapproximation
    - Vektorquantisierung

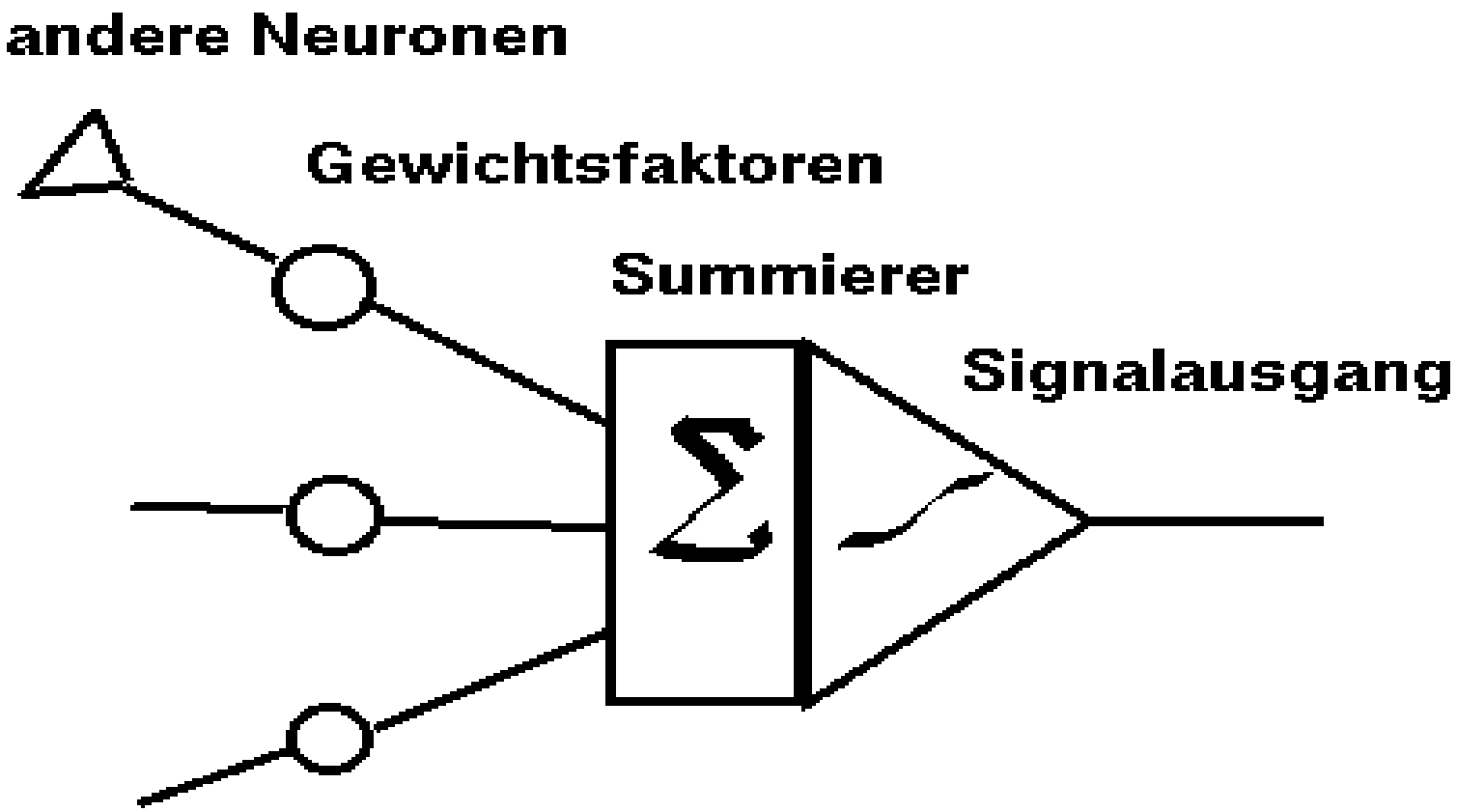
Ein biologisches Neuron

The Major Structures of the Neuron

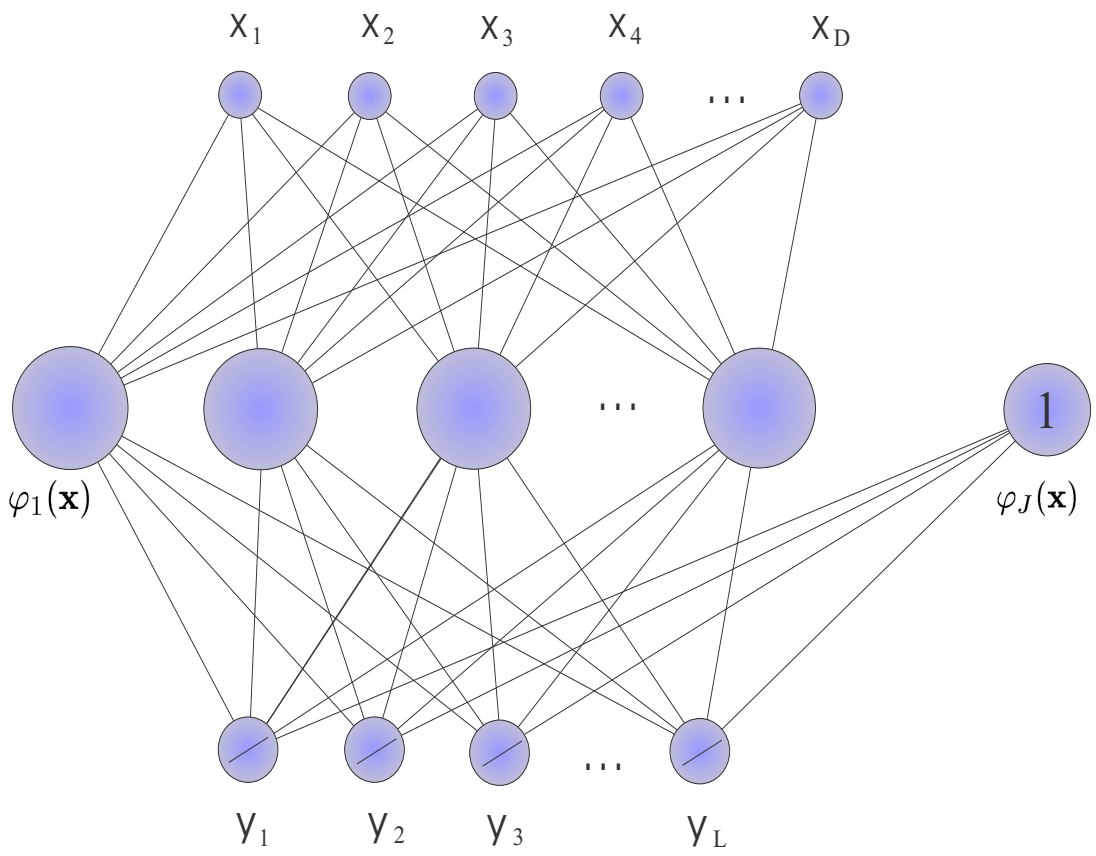
The neuron receives nerve impulses through its dendrites. It then sends the nerve impulses through its axon to the terminal buttons where neurotransmitters are released to stimulate other neurons.



Ein künstliches Neuron



Netze aus radialen Basisfunktionen



Netze aus radialen Basisfunktionen

Axonales Potential:

$$\varphi_j(\mathbf{x}) = \exp\Big(-\sum_{d=1}^{D} \frac{||\mathbf{x}_d - \mathbf{c}_{jd}||_2^2}{2\sigma_{jd}^2}\Big)$$

Netzausgabe:

$$\mathbf{y}_l = \sum_{j=0}^{J} \mathbf{w}_{jl}\varphi_j(\mathbf{x})$$

Netze aus radialen Basisfunktionen

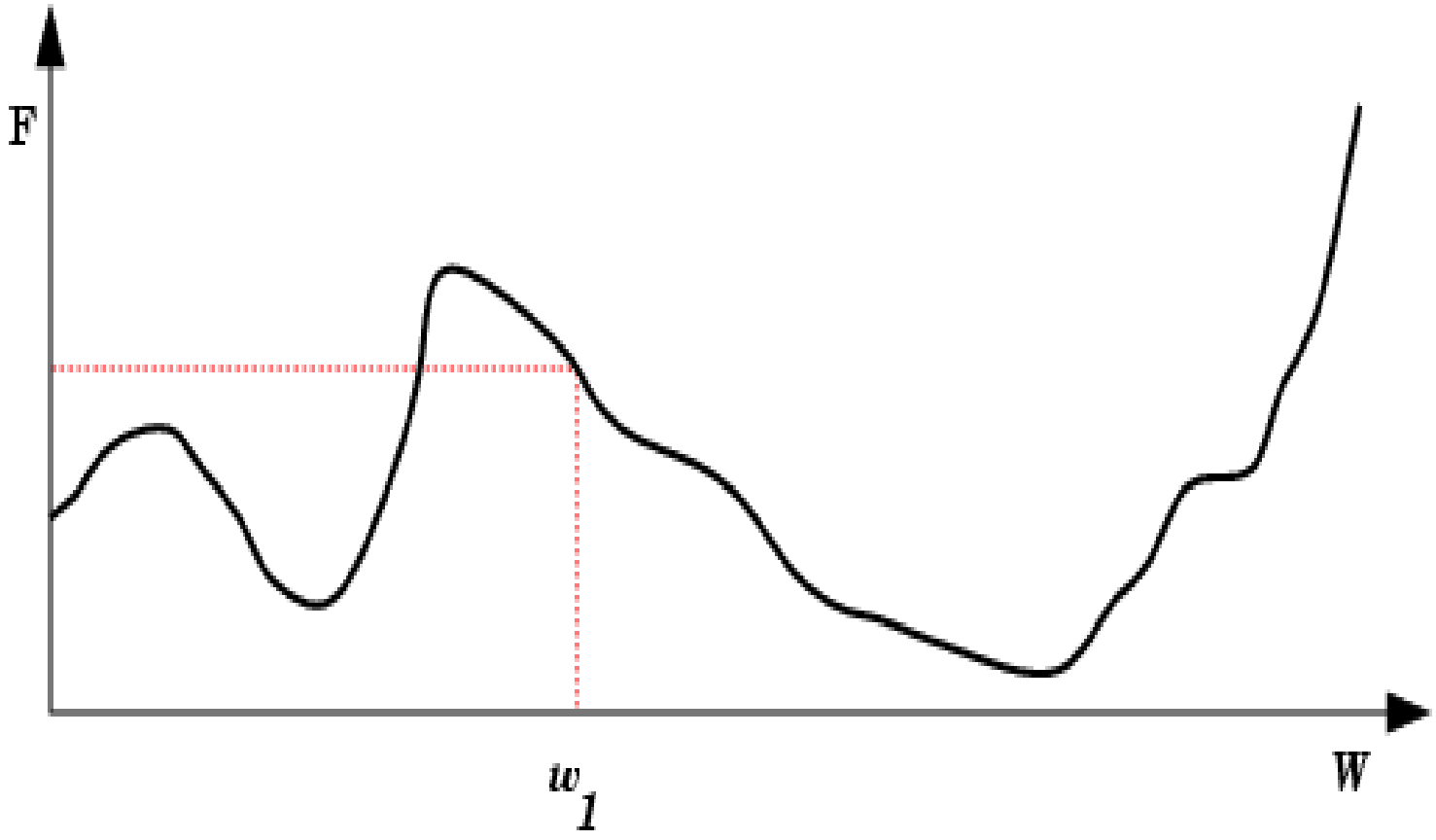
Lernen:

$$\mathbf{w}_{jl} = \mathbf{w}_{jl} - \eta_1 \frac{\partial E}{\partial \mathbf{w}_{jl}} = \mathbf{w}_{jl} + \eta_1 \sum_{\mu} (\hat{\mathbf{y}}_l^{\mu} - \mathbf{y}_l^{\mu}) \varphi_j(\mathbf{x}^{\mu}).$$

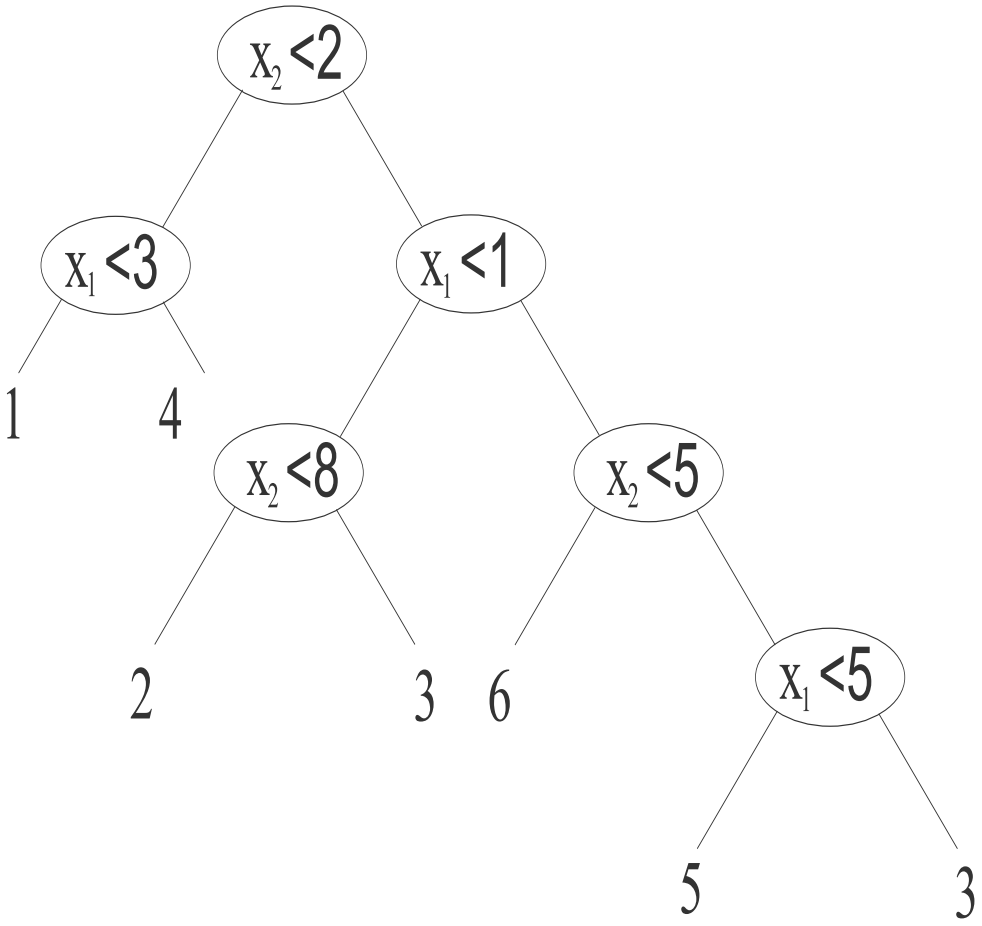
$$\mathbf{c}_{jd} = \mathbf{c}_{jd} - \eta_2 \frac{\partial E}{\partial \mathbf{c}_{jd}} = \mathbf{c}_{jd} + \eta_2 \sum_{\mu} \frac{\mathbf{x}_i^{\mu} - \mathbf{c}_{jd}^{\mu}}{\sigma_{jd}^2} \varphi_j(\mathbf{x}^{\mu}) \sum_{l=1}^{L} (\hat{\mathbf{y}}_l^{\mu} - \mathbf{y}_l^{\mu}) \mathbf{w}_{jl}$$

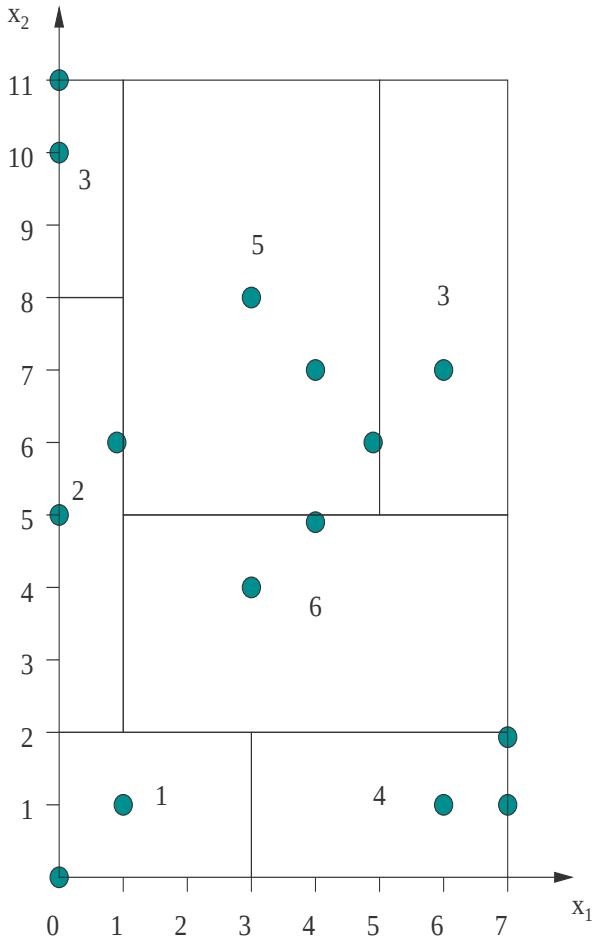
$$\sigma_{jd} = \sigma_{jd} - \eta_3 \frac{\partial E}{\partial \sigma_{jd}} = \sigma_{jd} + \eta_3 \sum_{\mu} \frac{(\mathbf{x}_i^{\mu} - \mathbf{c}_{jd}^{\mu})^2}{\sigma_{jd}^3} \varphi_j(\mathbf{x}^{\mu}) \sum_{l=1}^{L} (\hat{\mathbf{y}}_l^{\mu} - \mathbf{y}_l^{\mu}) \mathbf{w}_{jl}$$

Parameteranpassung durch Gradientenabstieg



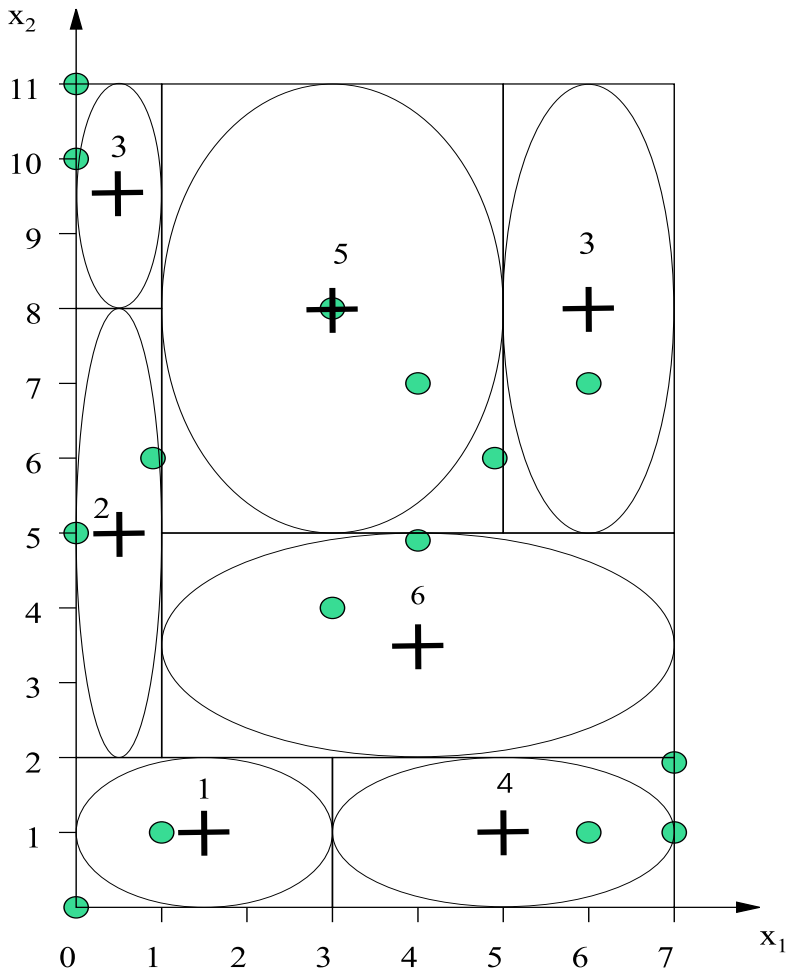
Initialisierung con RBF Netzen mit Entscheidungsbäumen



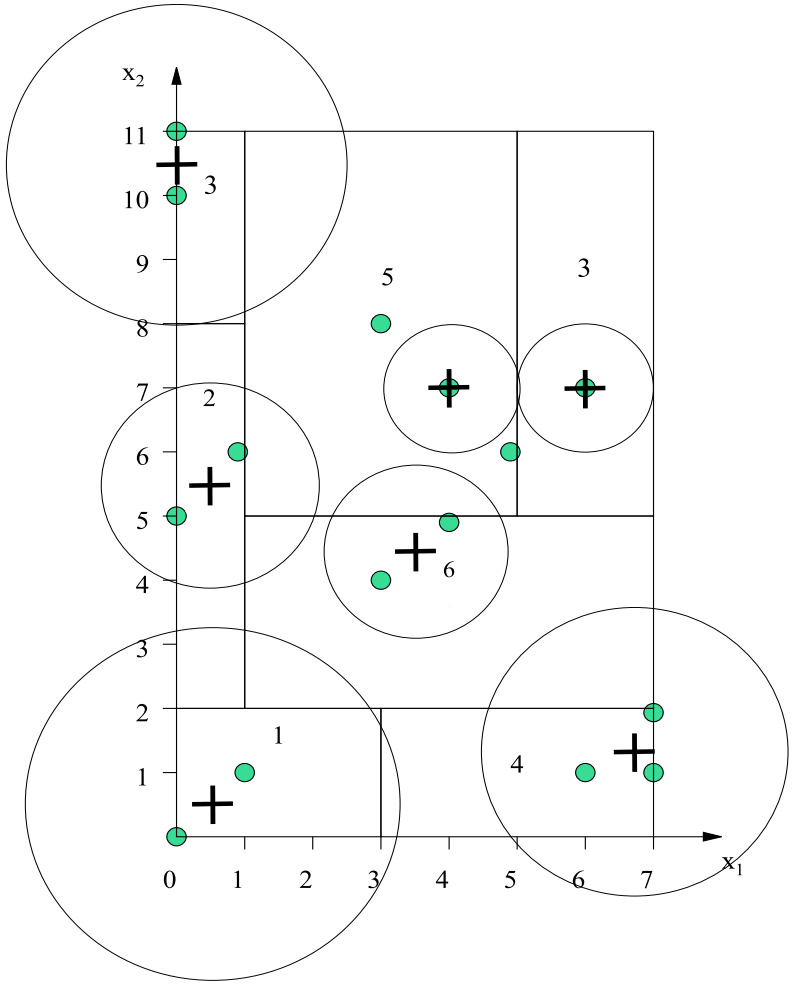


Initialisierung von RBF Netzen mit Entscheidungsbäumen

[Kubat 98]



[Schwenker, Dietrich 00]



Bioakustik

Bioakustik

- Die Bioakustik ist ein interdisziplinäres Gebiet, das viele Parallelen zur Sprachverarbeitung aufweist.

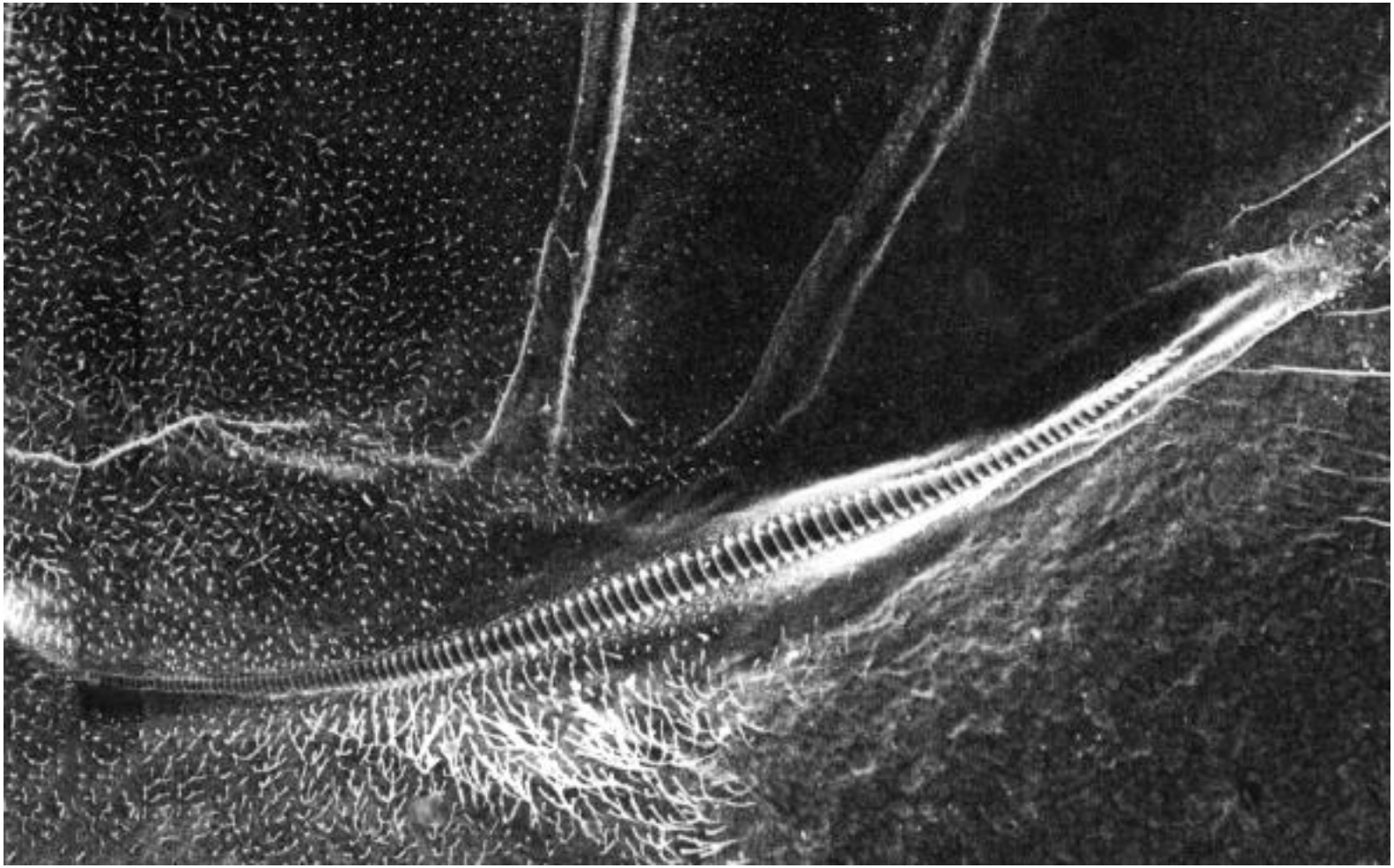
Bioakustik

- Die Bioakustik ist ein interdisziplinäres Gebiet, das viele Parallelen zur Sprachverarbeitung aufweist.
- Die Erzeugung sowie die Wahrnehmung der akustischen Signale werden in beiden Disziplinen genau untersucht.

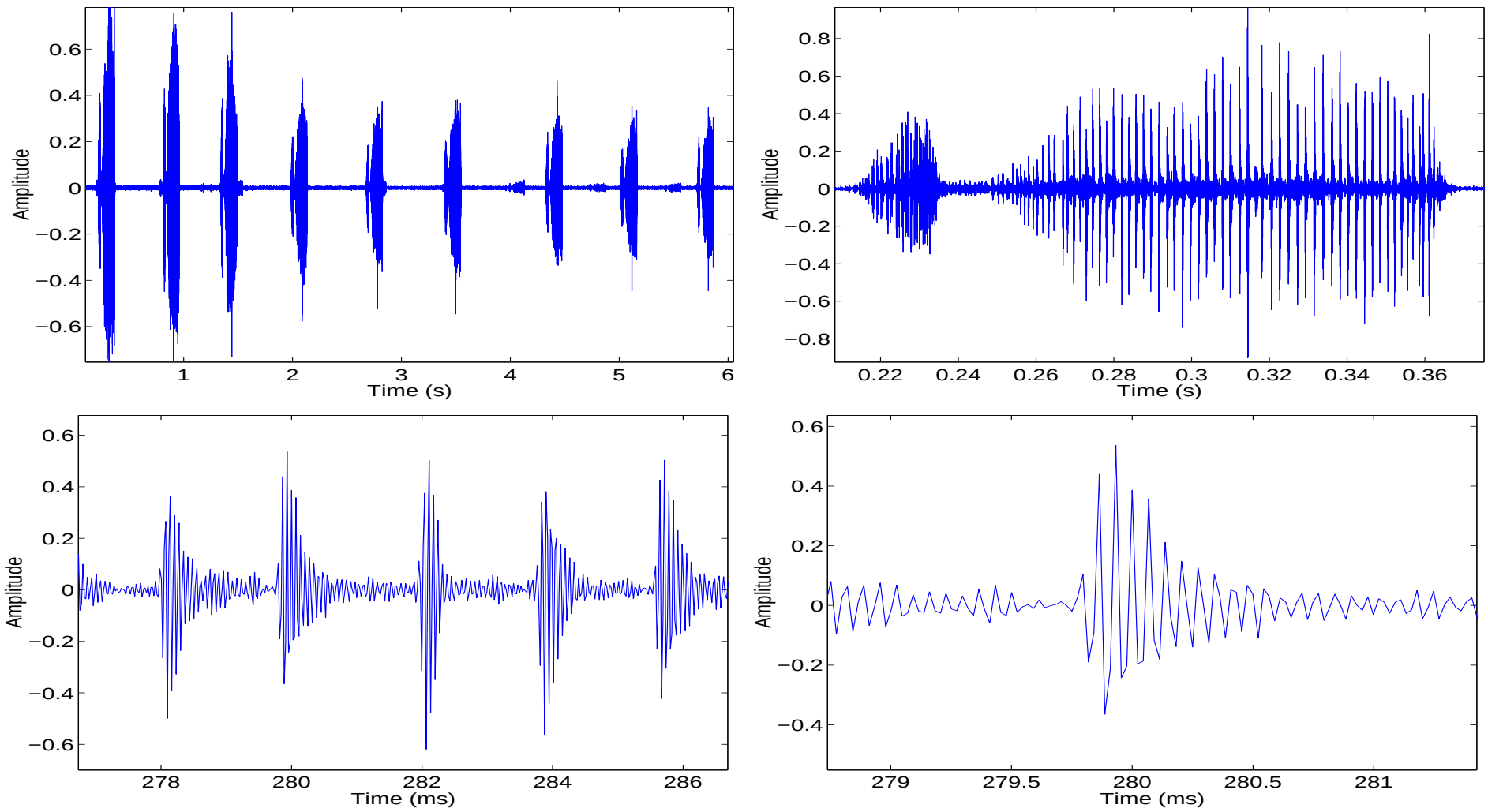
Bioakustik

- Die Bioakustik ist ein interdisziplinäres Gebiet, das viele Parallelen zur Sprachverarbeitung aufweist.
- Die Erzeugung sowie die Wahrnehmung der akustischen Signale werden in beiden Disziplinen genau untersucht.
- Aus dieser Untersuchung lassen sich Merkmale für die Erkennung akustischer Signale und deren Klassifikation ableiten.

Stridulationsleiste der *Zvenella yunnana*

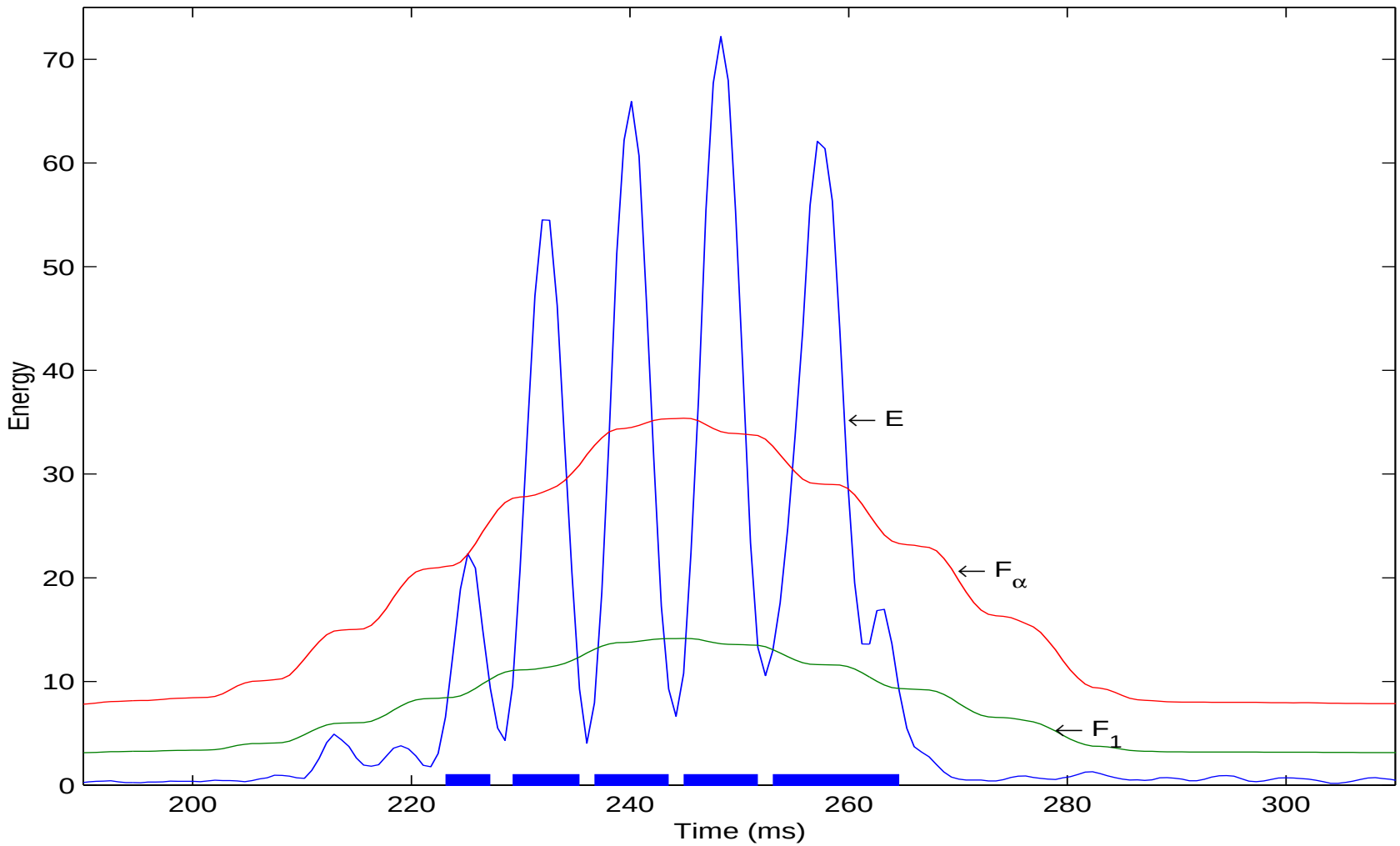


Signalstruktur: *Ephippiger ephippiger*



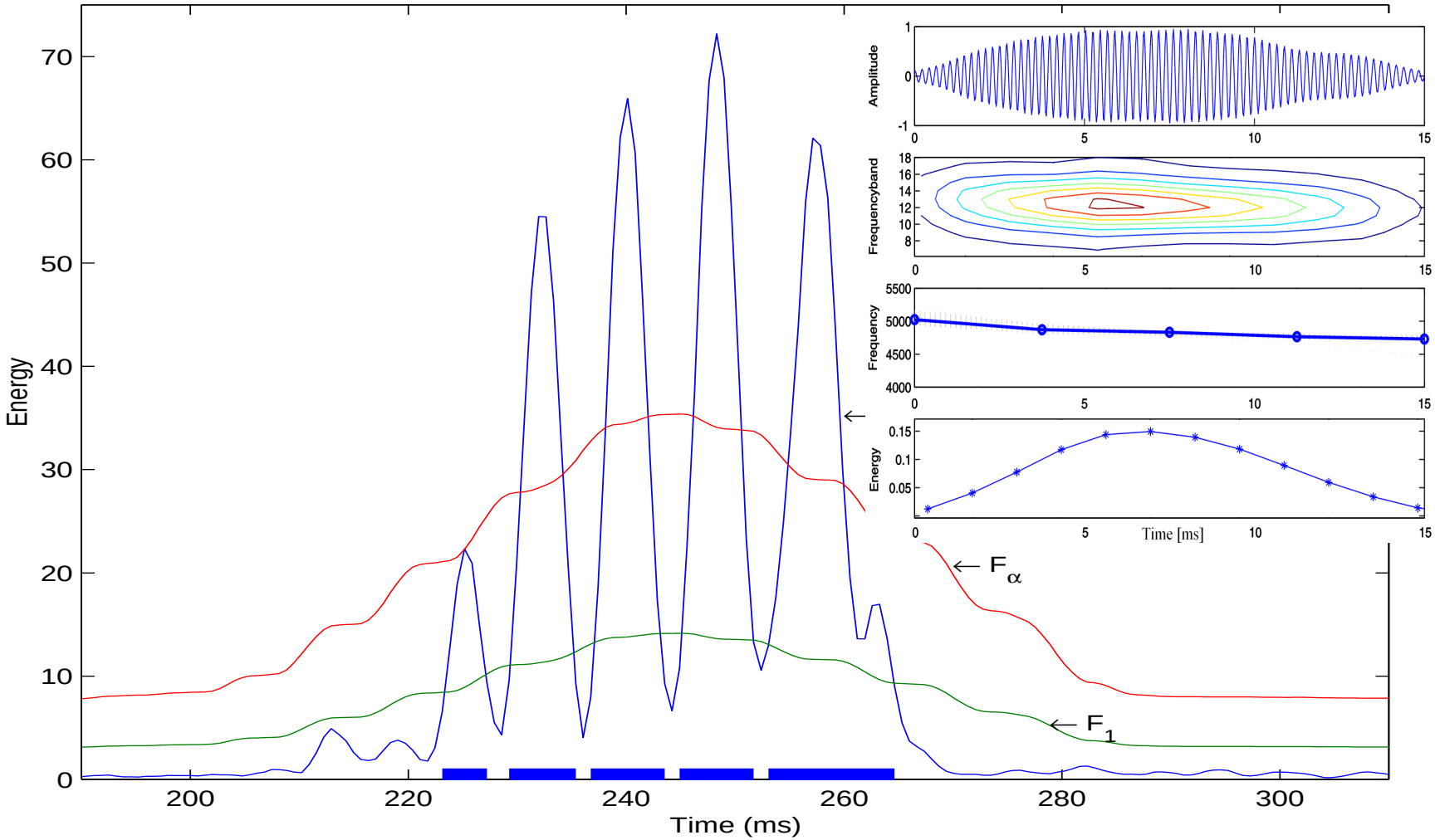
Signalsegmentierung - Pulsdetektion: *Zvenella transversa*

Idee: [Rabiner, Sambur 75]

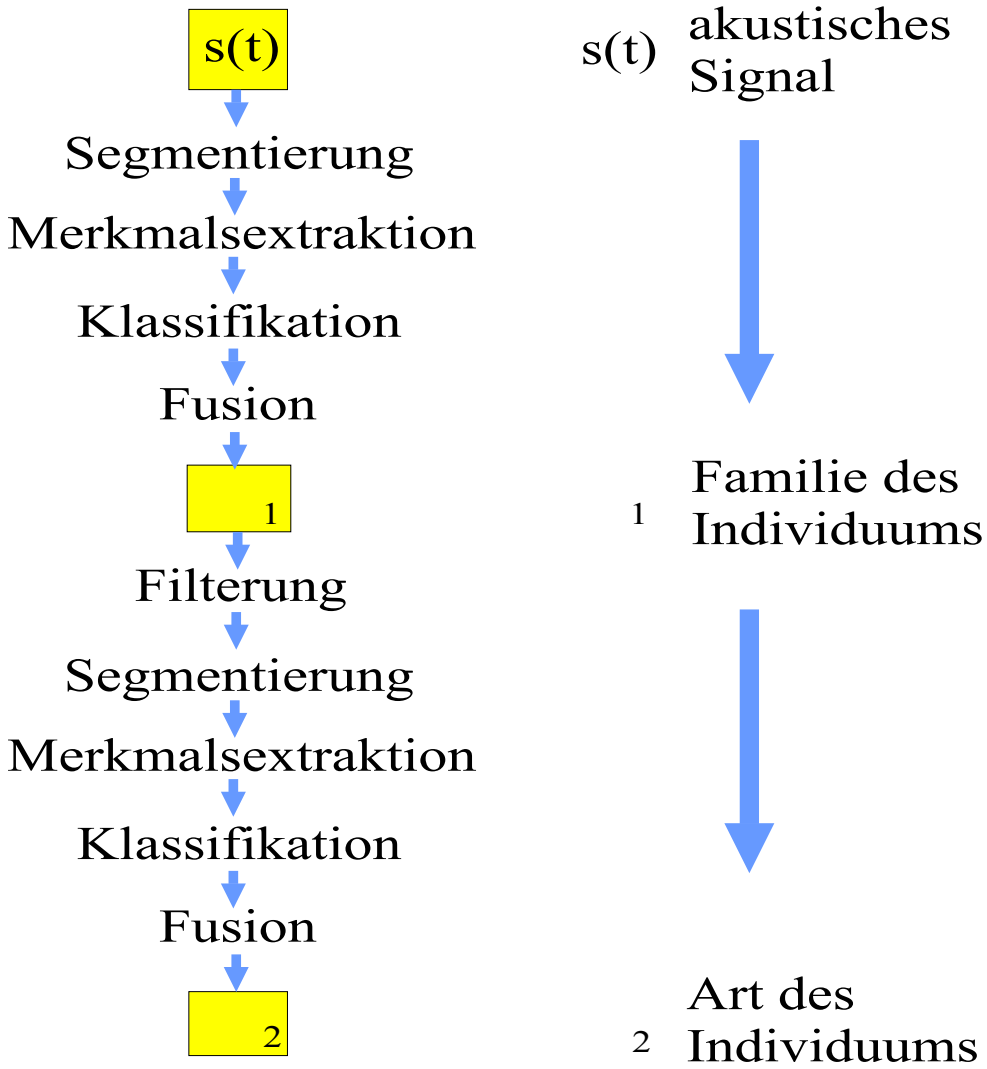


Signalsegmentierung - Pulsdetektion: *Zvenella transversa*

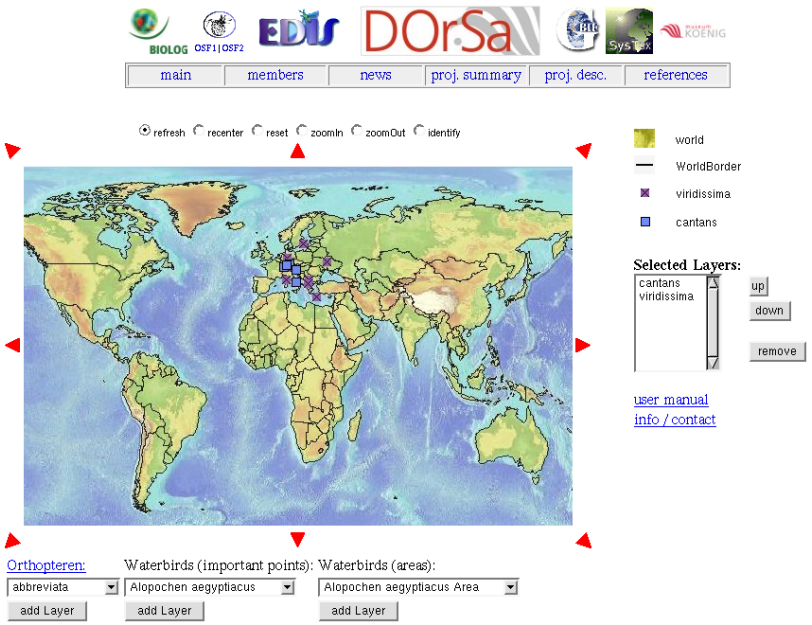
Idee: [Rabiner, Sambur 75]



Ablauf der vollautomatischen Klassifikation

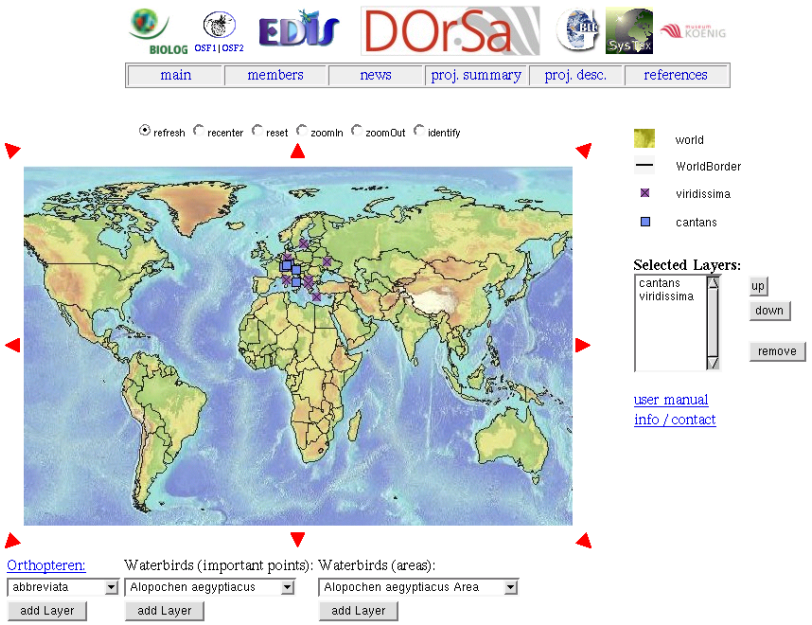


Anwendungsbereiche und Motivation



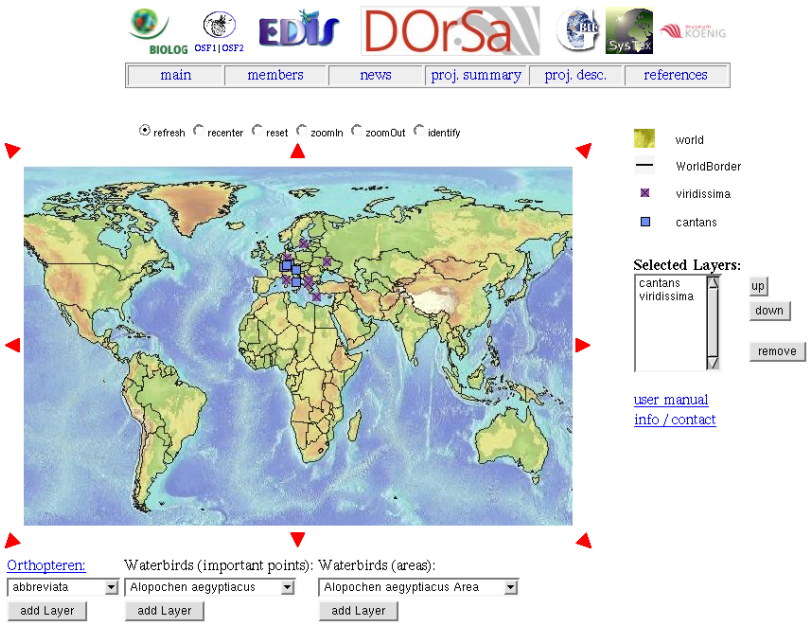
Anwendungsbereiche und Motivation

- Kartierung biologisch intakter Gebiete

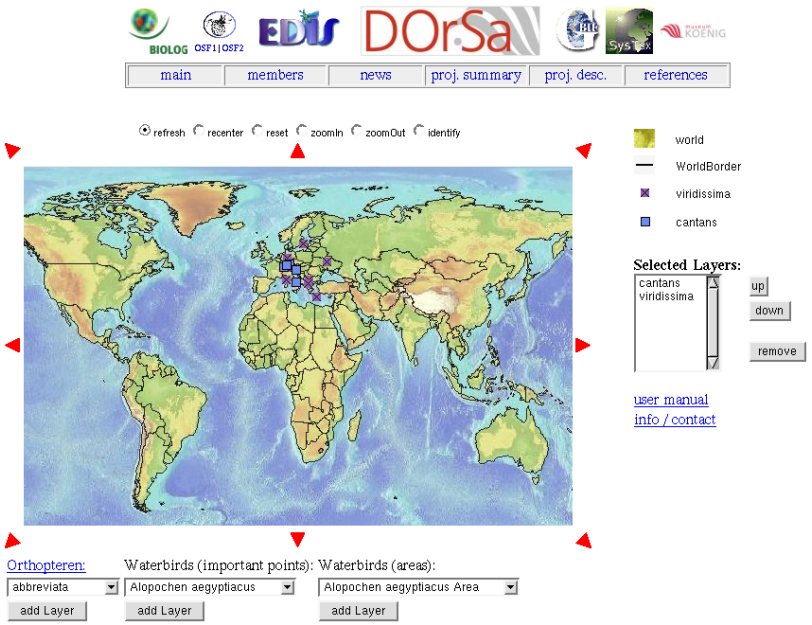


Anwendungsbereiche und Motivation

- Kartierung biologisch intakter Gebiete
- Unterstützung des Taxonomen durch
  - ⋆ Klassifikationstool
  - ⋆ automatische Extraktion von Merkmalen

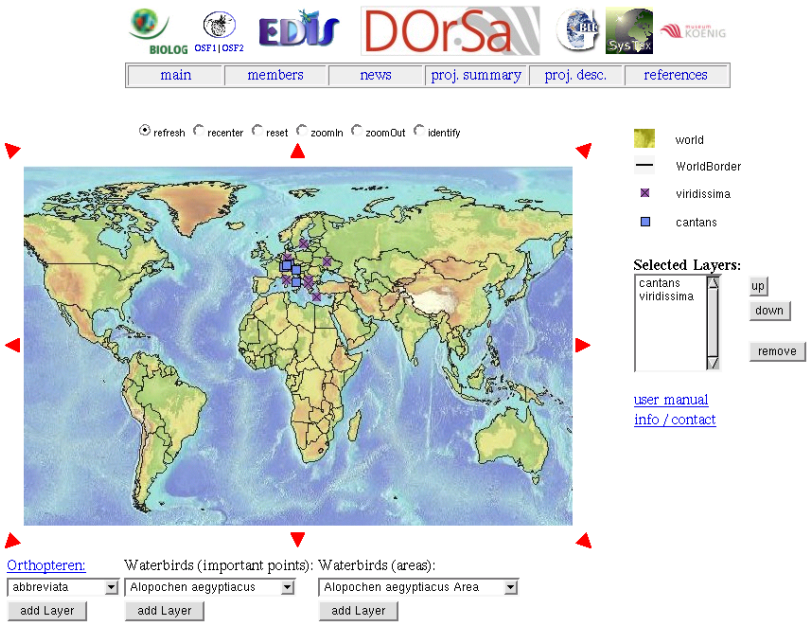


Anwendungsbereiche und Motivation



- Kartierung biologisch intakter Gebiete
- Unterstützung des Taxonomen durch
  - ⋆ Klassifikationstool
  - ⋆ automatische Extraktion von Merkmalen
- Qualitätskontrolle in großen Datenbeständen (www.dorsa.de)

Anwendungsbereiche und Motivation



- Kartierung biologisch intakter Gebiete
- Unterstützung des Taxonomen durch
  - ⋆ Klassifikationstool
  - ⋆ automatische Extraktion von Merkmalen
- Qualitätskontrolle in großen Datenbeständen (www.dorsa.de)
- Untersuchung biologischer Sachverhalte

Mehrklassifikatorsysteme - Design und Methodik

Mehrklassifikatorsysteme - Design und Methodik

Ein Klassifikatorensemble besteht aus mehreren Basisklassifikatoren, deren Einzelentscheidungen zu einer Gesamtentscheidung fusioniert werden.

Mehrklassifikatorsysteme - Design und Methodik

Ein Klassifikatorensemble besteht aus mehreren Basisklassifikatoren, deren Einzelentscheidungen zu einer Gesamtentscheidung fusioniert werden.

1. **Integration über unterschiedliche Merkmalsräume**

Mehrklassifikatorsysteme - Design und Methodik

Ein Klassifikatorensemble besteht aus mehreren Basisklassifikatoren, deren Einzelentscheidungen zu einer Gesamtentscheidung fusioniert werden.

1. **Integration über unterschiedliche Merkmalsräume**

2. Unterschiedliche Klassifikatoren (neuronale Architekturen, Parametereinstellungen und Initialisierungen)

Mehrklassifikatorsysteme - Design und Methodik

Ein Klassifikatorensemble besteht aus mehreren Basisklassifikatoren, deren Einzelentscheidungen zu einer Gesamtentscheidung fusioniert werden.

1. **Integration über unterschiedliche Merkmalsräume**

2. Unterschiedliche Klassifikatoren (neuronale Architekturen, Parametereinstellungen und Initialisierungen)

3. Unterschiedliche Trainingsdaten (Bagging, Boosting, ...)

Mehrklassifikatorsysteme - Design und Methodik

Ein Klassifikatorensemble besteht aus mehreren Basisklassifikatoren, deren Einzelentscheidungen zu einer Gesamtentscheidung fusioniert werden.

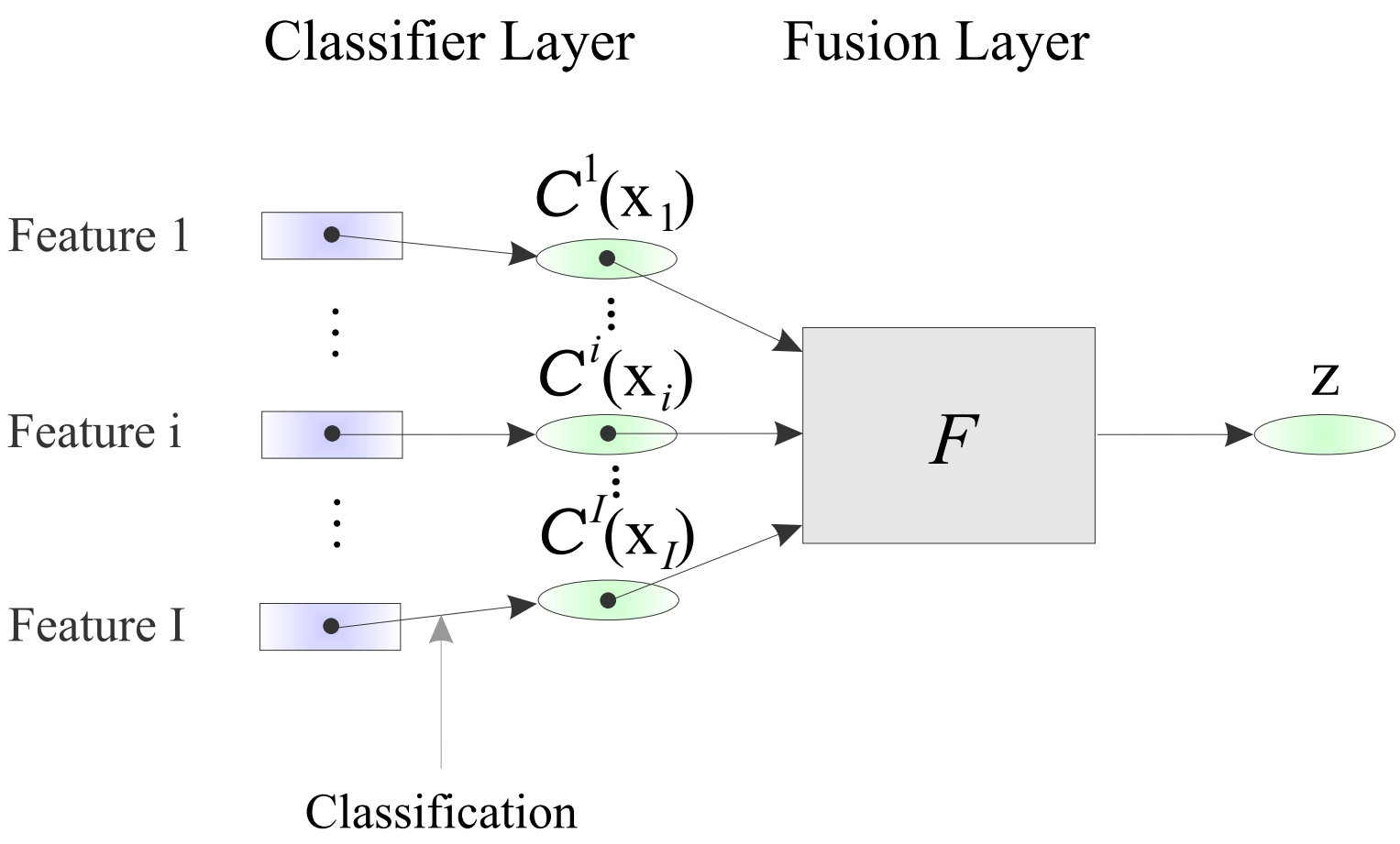
1. **Integration über unterschiedliche Merkmalsräume**

2. Unterschiedliche Klassifikatoren (neuronale Architekturen, Parametereinstellungen und Initialisierungen)

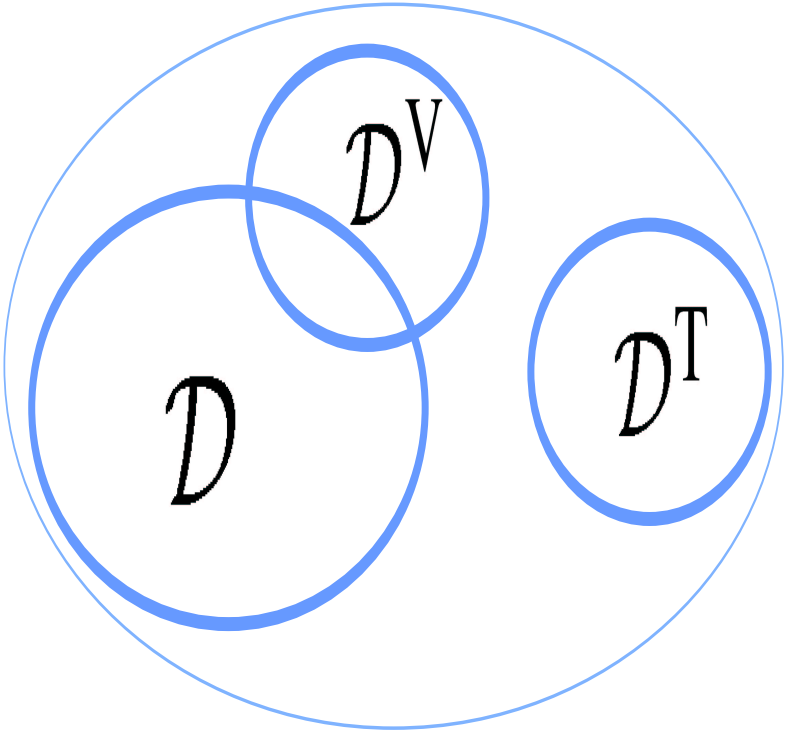
3. Unterschiedliche Trainingsdaten (Bagging, Boosting, ...)

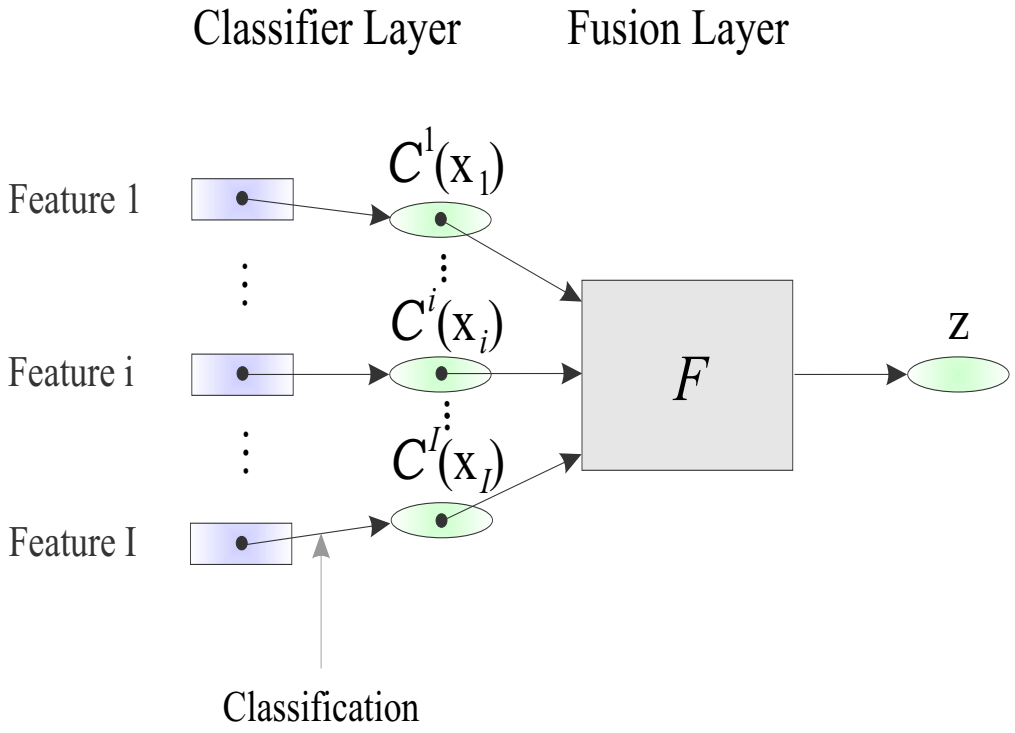
(1), (2) Methoden:
statische Methoden:
Majority Voting, Perzentile, Borda Count, probabilistische Fusion, ...
adaptive Methoden:
Behaviour Knowledge Space, Naive Bayes Fusion, Decision Templates, neuronale Fusion, Pseudoinverse Matrix, ...

Mehrklassifikatorsysteme - Integration über mehrere Merkmalsräume

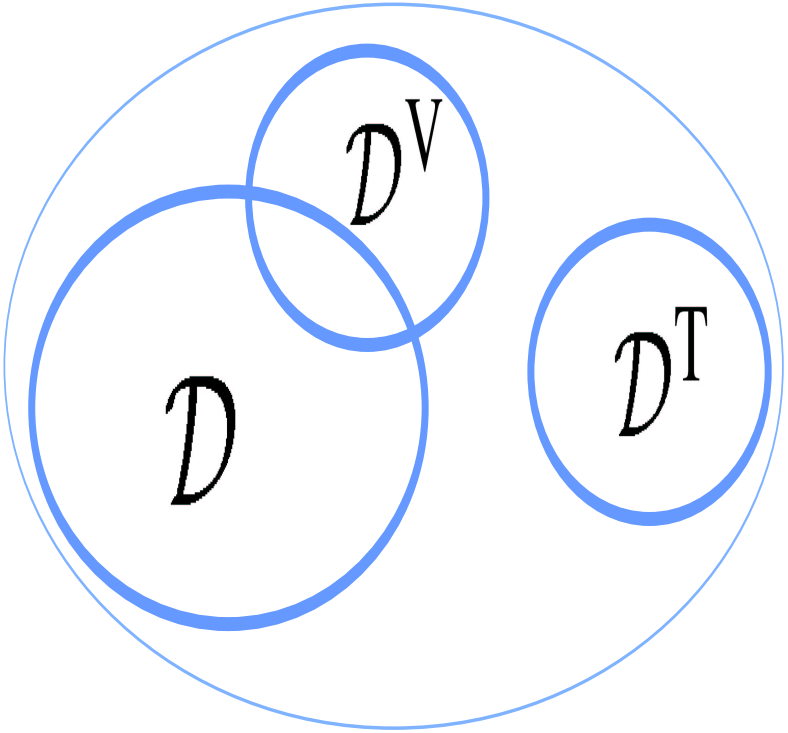


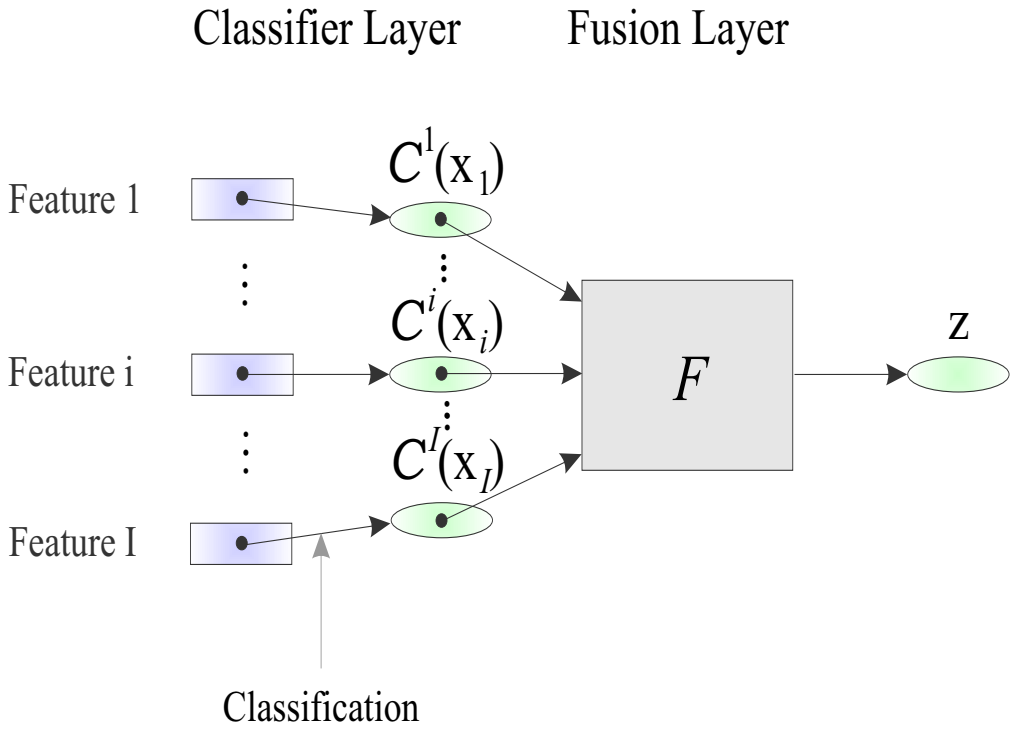
Ensemble Training





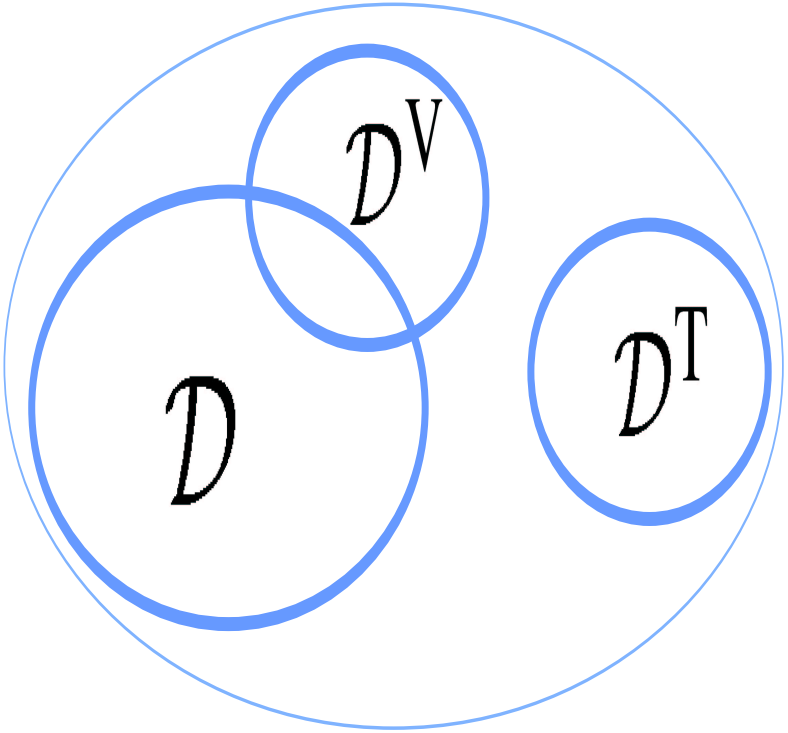
Ensemble Training

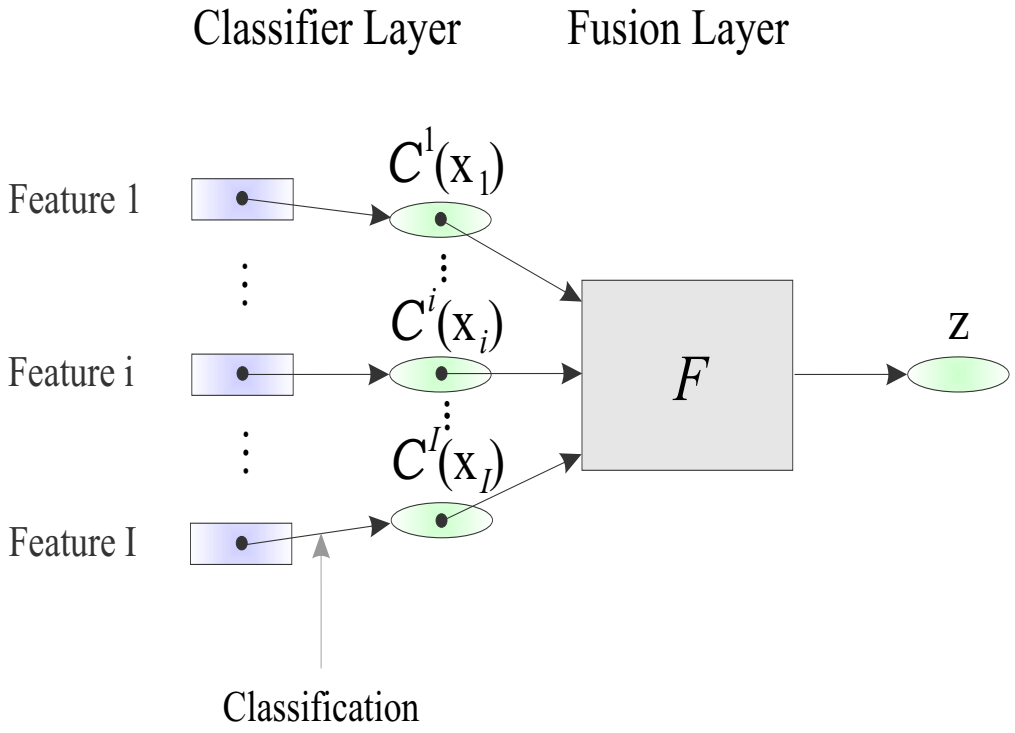




1. Training der Basisklassifikatoren mittels Trainingsdatensatz $\mathcal{D}$

Ensemble Training

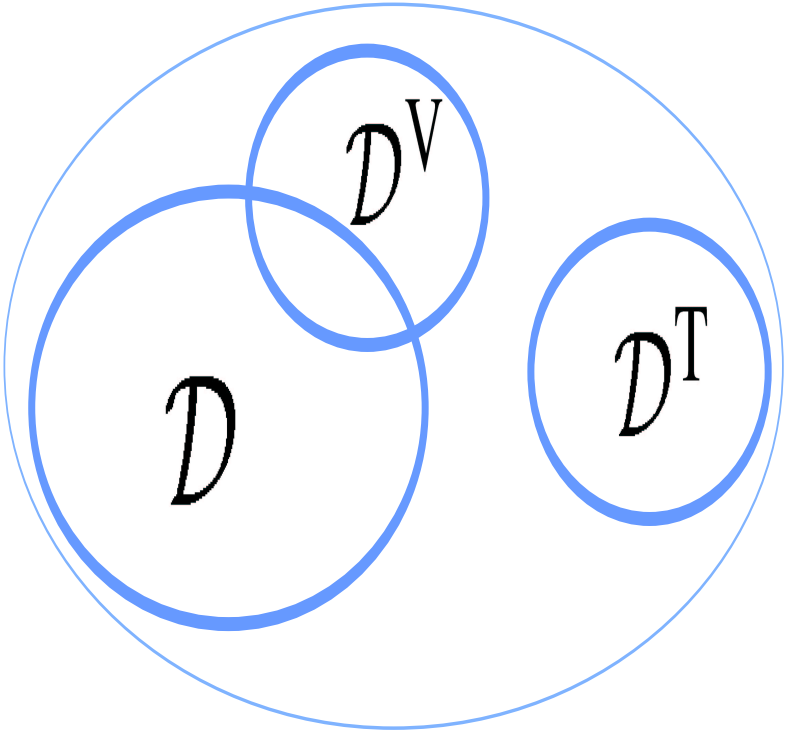


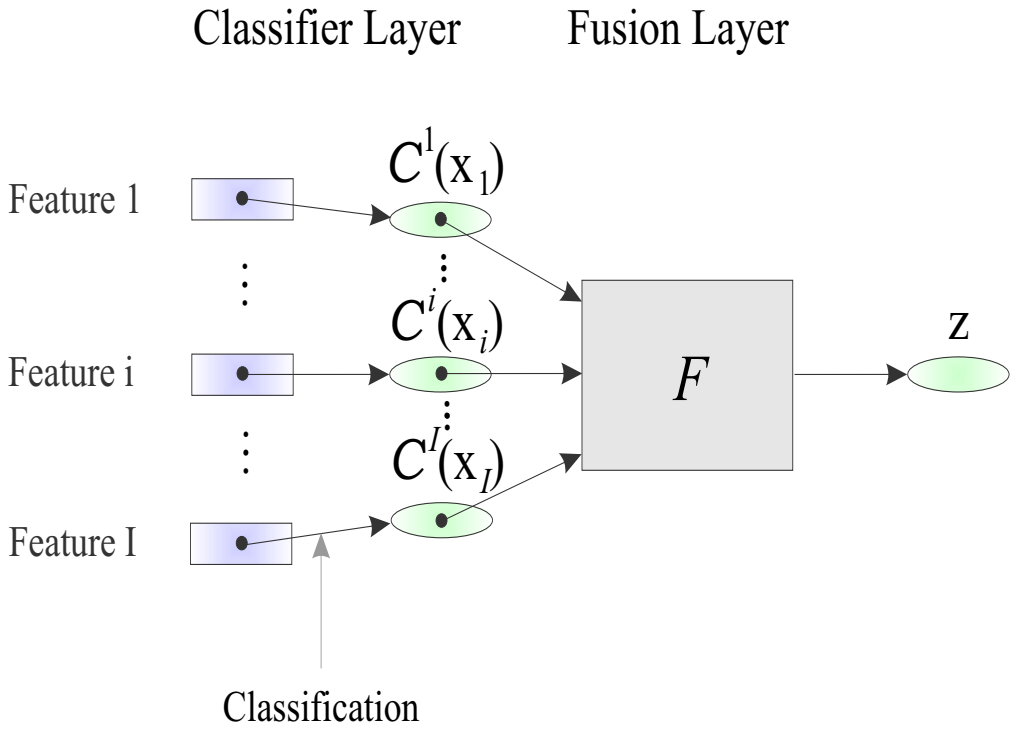


1. Training der Basisklassifikatoren mittels Trainingsdatensatz $\mathcal{D}$

2. Training der Fusionsschicht mittels Datensatz $\mathcal{D}^V$

Ensemble Training





1. Training der Basisklassifikatoren mittels Trainingsdatensatz $\mathcal{D}$
2. Training der Fusionsschicht mittels Datensatz $\mathcal{D}^V$
3. Nachtrainieren der Basisklassifikatoren mit $\mathcal{D} \cup \mathcal{D}^V$

Decision Templates - Training

[Kuncheva 95]

Decision Templates - Training

[Kuncheva 95]

Für jede Klasse in $\omega \in \Omega$ wird eine repräsentative Klassifikatorausgabe durch Mittelwertbildung der Klassifikatorausgaben für Validierungsdaten aus $\mathcal{D}_\omega^V$ berechnet:

$$\mathcal{T}^\omega := \frac{1}{|\mathcal{D}_\omega^V|} \sum_{\mu \in \mathcal{D}_\omega^V} \mathcal{C}(\mathbf{x}^\mu)$$

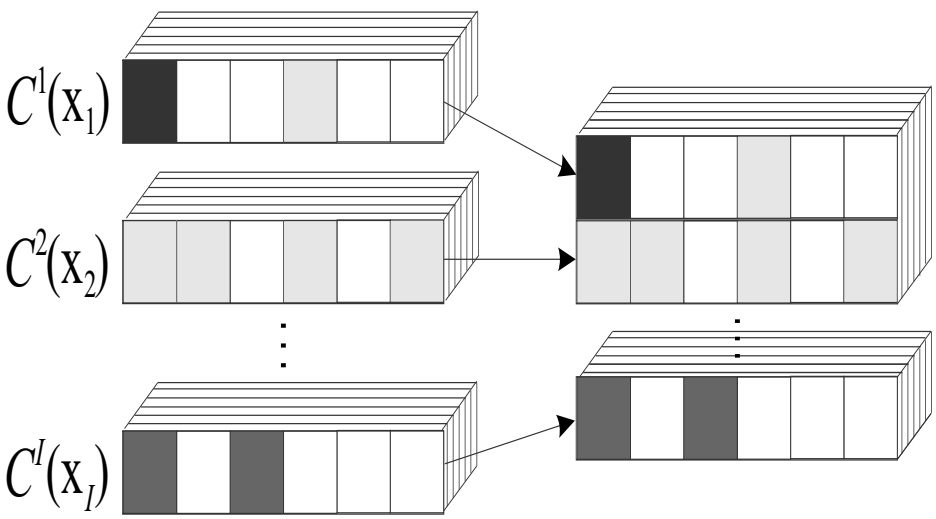
Decision Templates - Training

[Kuncheva 95]

Für jede Klasse in $\omega \in \Omega$ wird eine repräsentative Klassifikatorausgabe durch Mittelwertbildung der Klassifikatorausgaben für Validierungsdaten aus $\mathcal{D}_\omega^V$ berechnet:

$$\mathcal{T}^\omega := \frac{1}{|\mathcal{D}_\omega^V|} \sum_{\mu \in \mathcal{D}_\omega^V} \mathcal{C}(\mathbf{x}^\mu)$$

Decision Profile $\mathcal{P}$



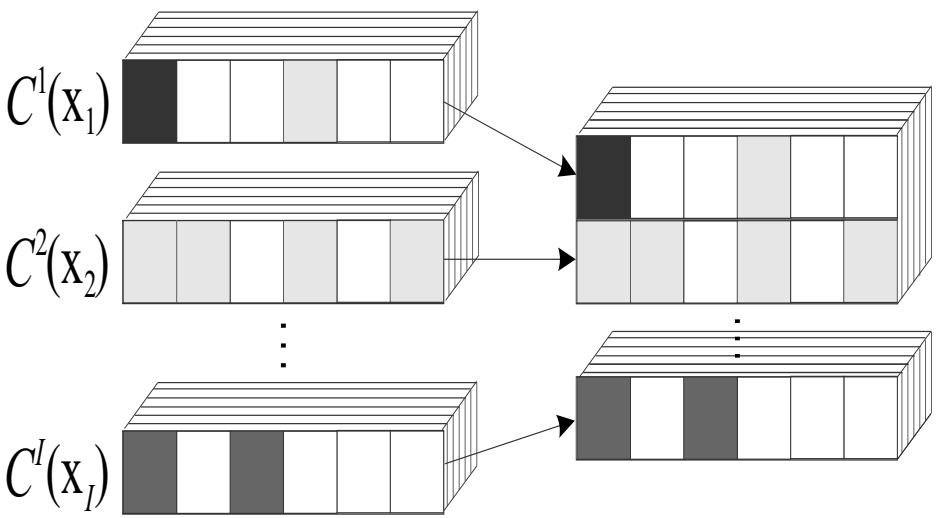
Decision Templates - Training

[Kuncheva 95]

Für jede Klasse in $\omega \in \Omega$ wird eine repräsentative Klassifikatorausgabe durch Mittelwertbildung der Klassifikatorausgaben für Validierungsdaten aus $\mathcal{D}_\omega^V$ berechnet:

$$\mathcal{T}^\omega := \frac{1}{|\mathcal{D}_\omega^V|} \sum_{\mu \in \mathcal{D}_\omega^V} \mathcal{C}(\mathbf{x}^\mu)$$

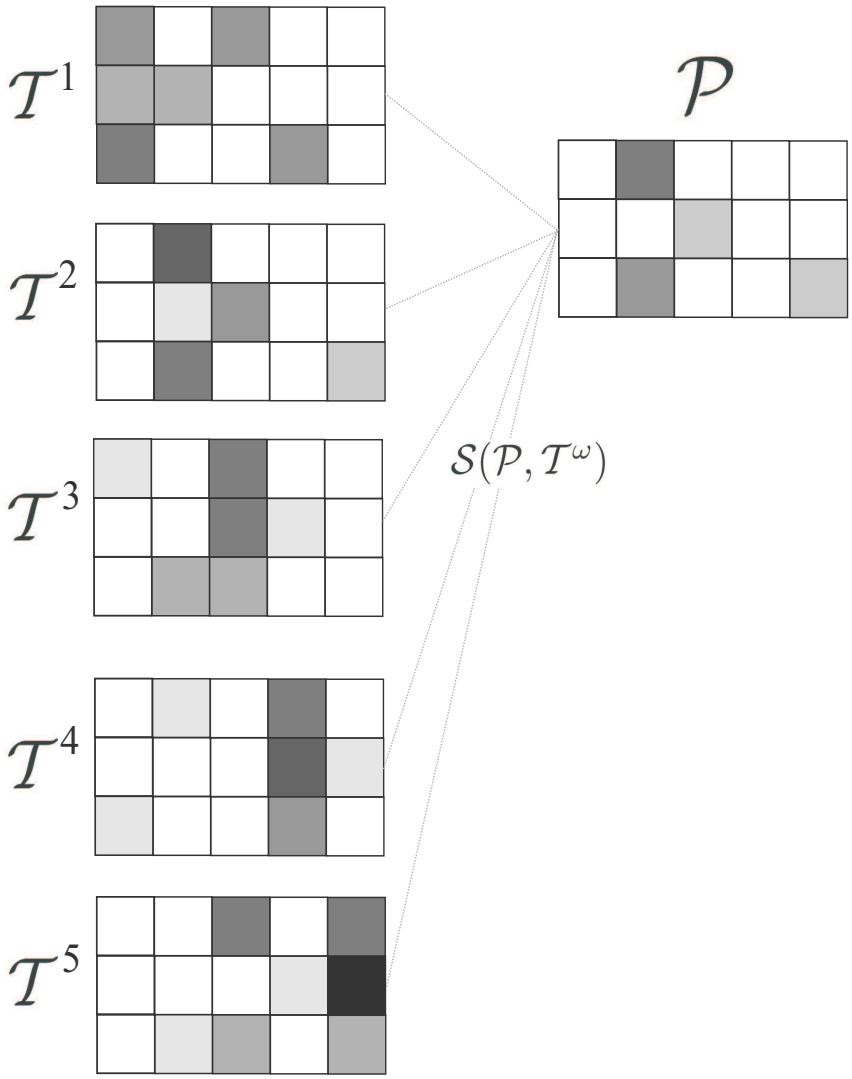
Decision Profile $\mathcal{P}$



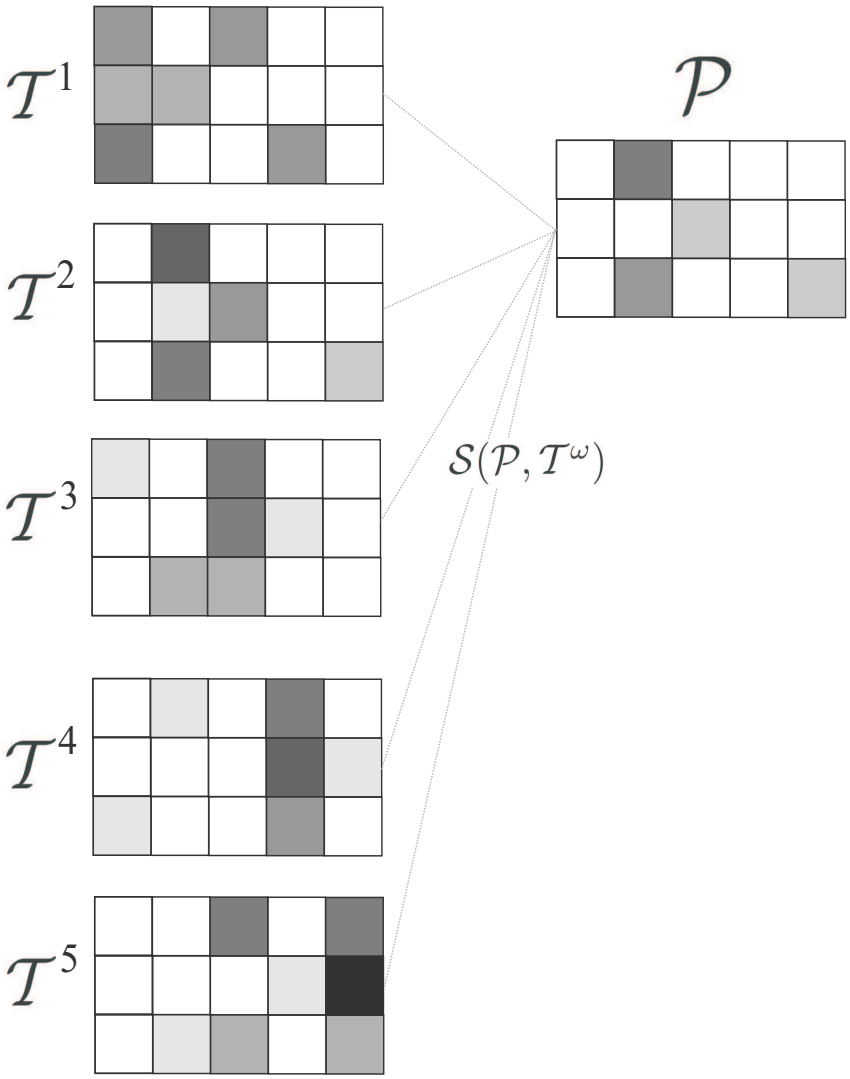
Decision Template $\mathcal{T}^\omega$

$$\mathcal{T}^\omega := \begin{bmatrix} \mathcal{T}_{1,1}^\omega & \dots & \mathcal{T}_{1,l}^\omega & \dots & \mathcal{T}_{1,L}^\omega \\ \vdots & \dots & \vdots & \dots & \vdots \\ \mathcal{T}_{i,1}^\omega & \dots & \mathcal{T}_{i,l}^\omega & \dots & \mathcal{T}_{i,L}^\omega \\ \vdots & \dots & \vdots & \dots & \vdots \\ \mathcal{T}_{I,1}^\omega & \dots & \mathcal{T}_{I,l}^\omega & \dots & \mathcal{T}_{I,L}^\omega \end{bmatrix}$$

Decision Templates - Klassifikation



Decision Templates - Klassifikation



Algorithm $\omega^* = \mathrm{DT}((\mathbf{x}_i)_{i=1}^I, (\mathcal{T}^\omega)_{\omega=1}^L)$

$\mathcal{P} = [\mathcal{C}^1(\mathbf{x}_1) \ldots \mathcal{C}^I(\mathbf{x}_I)]^T$

foreach $\omega \in \Omega$

$C_\omega = \mathcal{S}(\mathcal{P}, \mathcal{T}^\omega)$

end

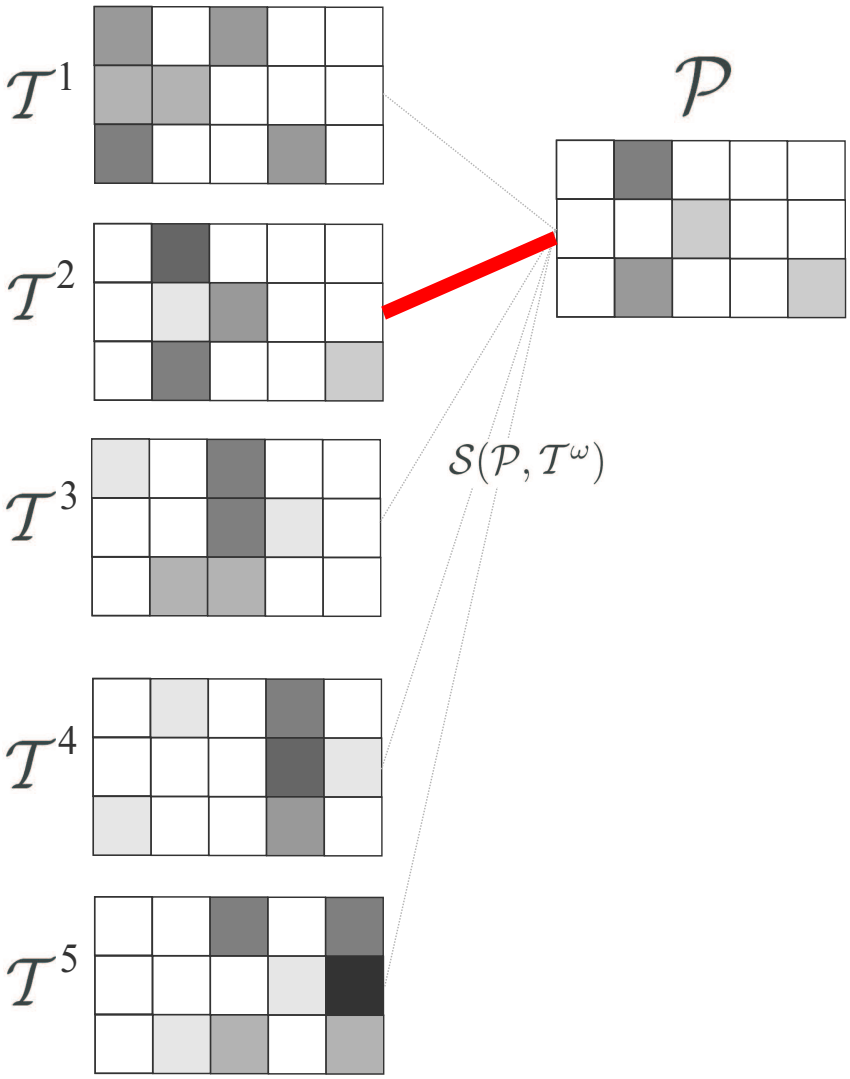
$\omega^* = \underset{\omega}{\operatorname{argmax}}(C_\omega)$

—

Algorithm DT:

Klassifikation mit Decision Templates

Decision Templates - Klassifikation



Algorithm $\omega^* = \mathrm{DT}((\mathbf{x}_i)_{i=1}^I, (\mathcal{T}^\omega)_{\omega=1}^L)$

$\mathcal{P} = [\mathcal{C}^1(\mathbf{x}_1) \ldots \mathcal{C}^I(\mathbf{x}_I)]^T$

foreach $\omega \in \Omega$

$C_\omega = \mathcal{S}(\mathcal{P}, \mathcal{T}^\omega)$

end

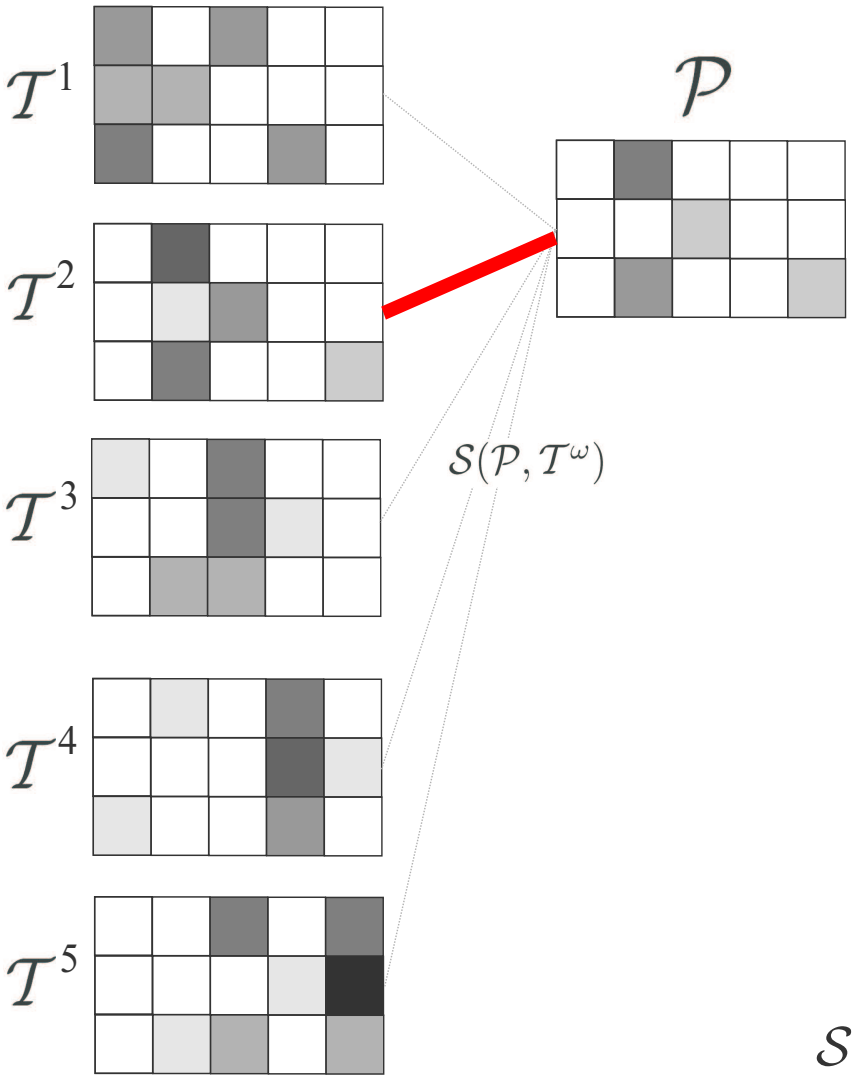
$\omega^* = \underset{\omega}{\operatorname{argmax}}(C_\omega)$

—

Algorithm DT:

Klassifikation mit Decision Templates

Decision Templates - Klassifikation



Algorithm $\omega^* = \mathrm{DT}((\mathbf{x}_i)_{i=1}^I, (\mathcal{T}^\omega)_{\omega=1}^L)$

$\mathcal{P} = [\mathcal{C}^1(\mathbf{x}_1) \ldots \mathcal{C}^I(\mathbf{x}_I)]^T$

foreach $\omega \in \Omega$

$C_\omega = \mathcal{S}(\mathcal{P}, \mathcal{T}^\omega)$

end

$\omega^* = \underset{\omega}{\operatorname{argmax}}(C_\omega)$

—

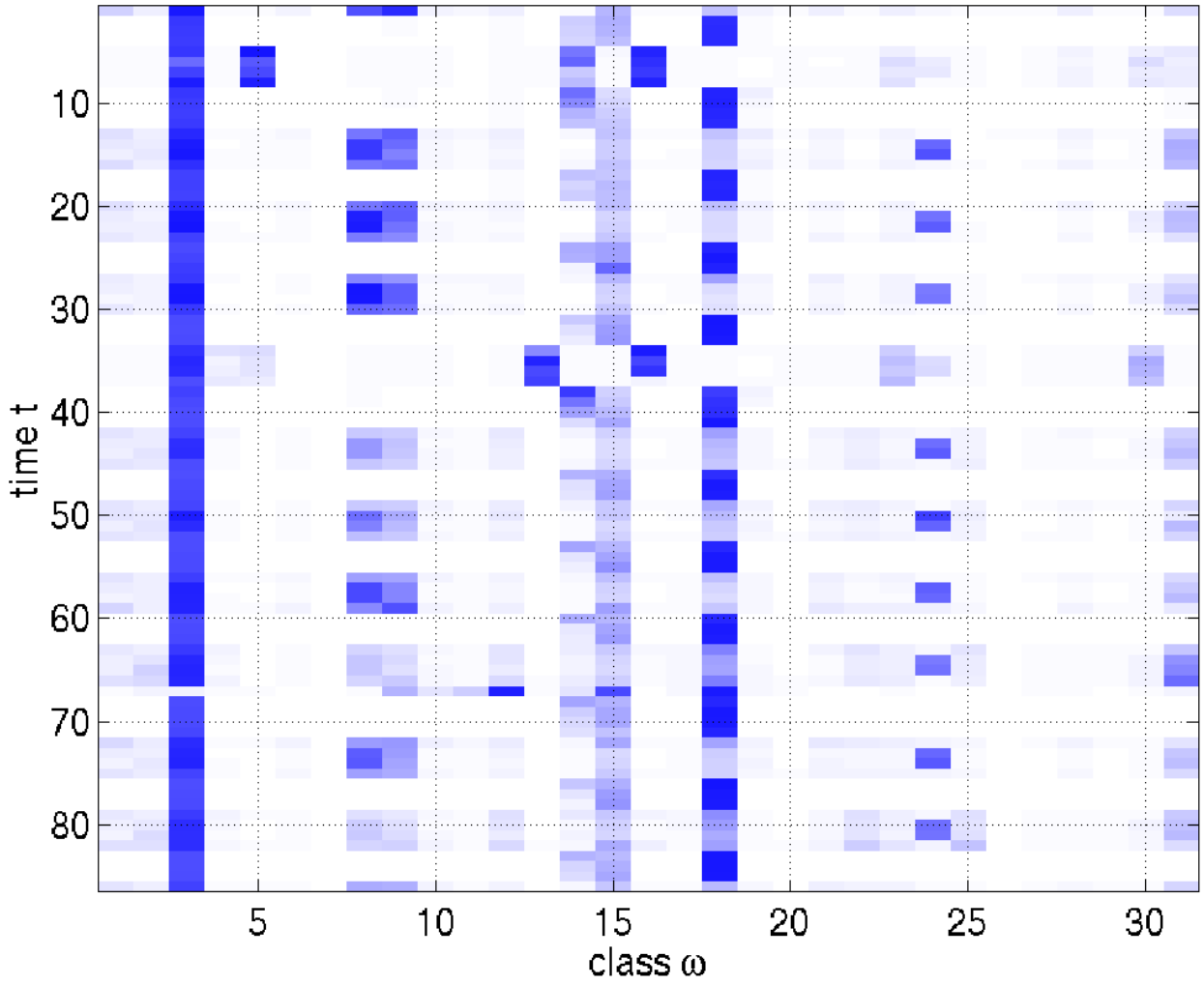
Algorithm DT:

Klassifikation mit Decision Templates

$$\mathcal{S}(\mathcal{P}, \mathcal{T}^\omega) = 1 - \frac{1}{2I} \sum_{i=1}^{I} ||\mathcal{P}_{i,\cdot} - \mathcal{T}^\omega_{i,\cdot}|| \quad \in [0, 1]$$

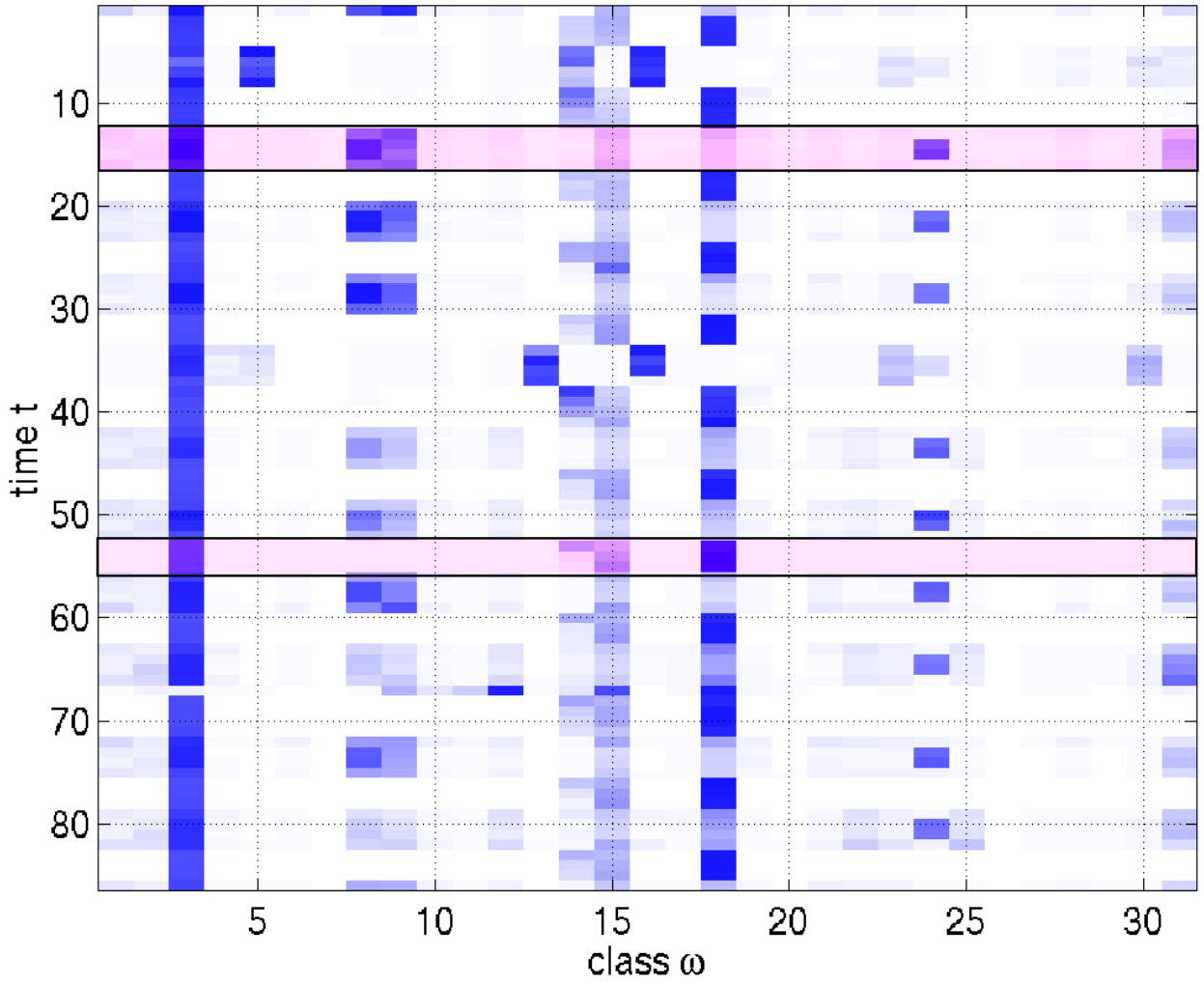
Multiple Decision Templates (MDT) - Motivation

Multiple Decision Templates (MDT) - Motivation



Aclodes chamocoru

Multiple Decision Templates (MDT) - Motivation



Aclodes chamocoru

Multiple Decision Templates (MDT) - Motivation

- Bei der Klassifikation von Zeitreihen treten typischerweise temporale Verwechslungen auf, die durch Standard Decision Template Methoden nicht hinreichend repräsentiert werden können. [Dietrich, Schwenker, Palm 02]

Multiple Decision Templates (MDT) - Motivation

- Bei der Klassifikation von Zeitreihen treten typischerweise temporale Verwechslungen auf, die durch Standard Decision Template Methoden nicht hinreichend repräsentiert werden können. [Dietrich, Schwenker, Palm 02]

- Die temporale Variation der Klassifikatorausgaben kann jedoch durch mehrere Decision Templates pro Klasse dargestellt werden.

MDT - Training

CDT: Clustered Decision Templates

Clusterung der Decision Profiles mit k-means über alle Daten der Klasse ω

MDT - Training

CDT: Clustered Decision Templates

Clusterung der Decision Profiles mit k-means über alle Daten der Klasse ω

k DTs pro Klasse

MDT - Training

CDT: Clustered Decision Templates

Clusterung der Decision Profiles mit k-means über alle Daten der Klasse ω

k DTs pro Klasse

TDT: Temporal Decision Templates

Zeitliche Mittelung statt Clusterung

MDT - Training

CDT: Clustered Decision Templates

Clusterung der Decision Profiles mit k-means über alle Daten der Klasse ω

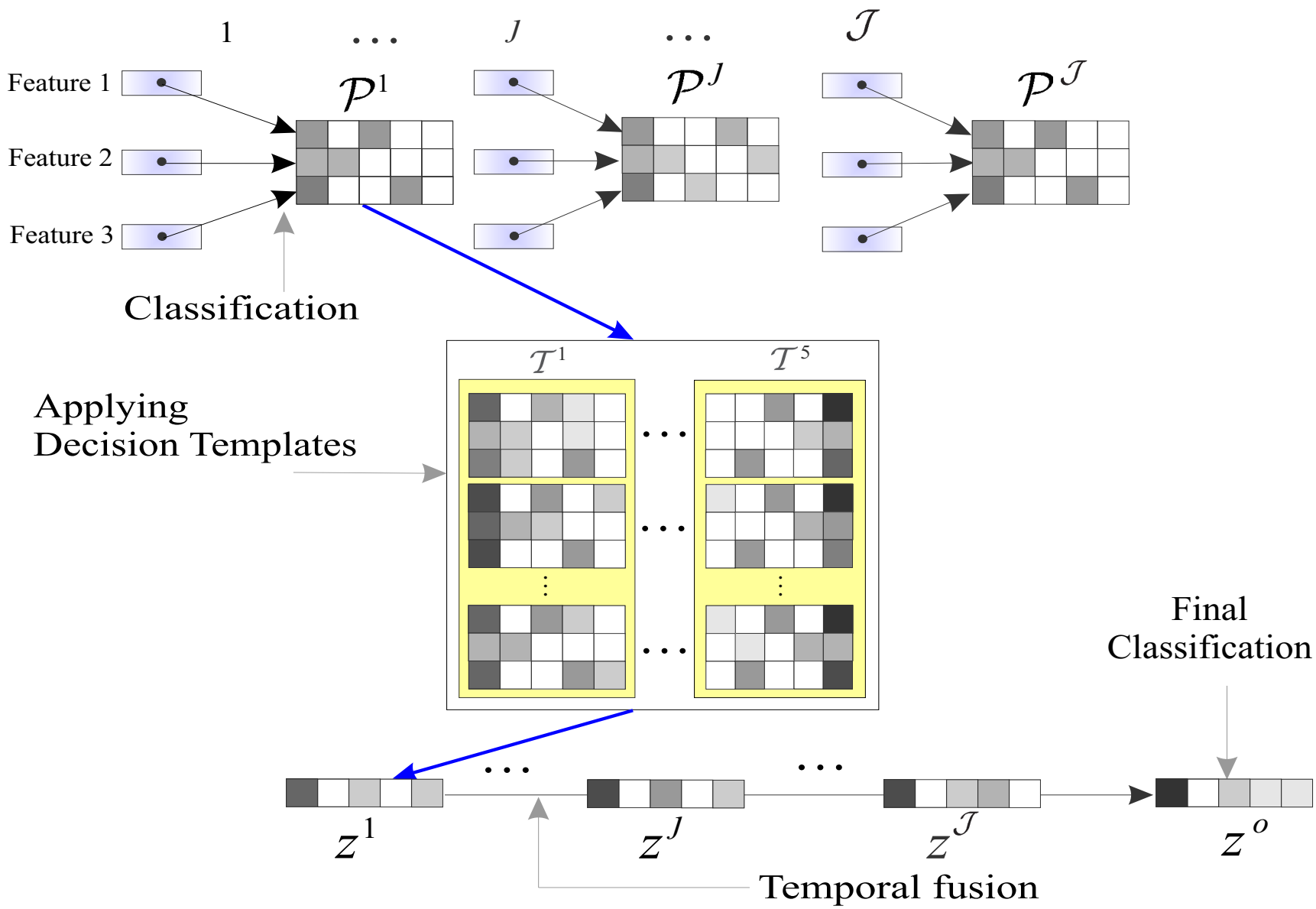
k DTs pro Klasse

TDT: Temporal Decision Templates

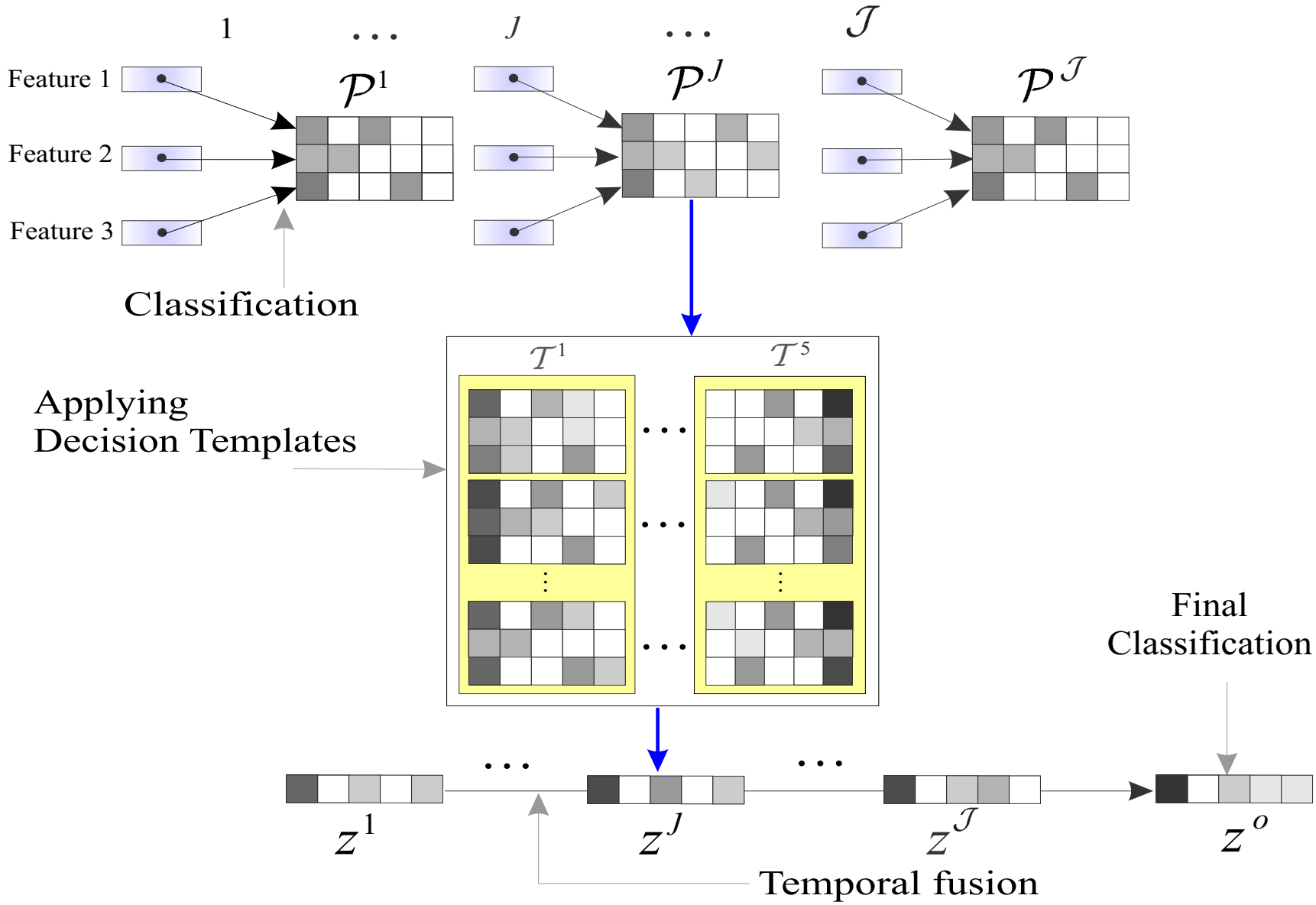
Zeitliche Mittelung statt Clusterung

$|\mathcal{D}_\omega^V|$ DTs pro Klasse

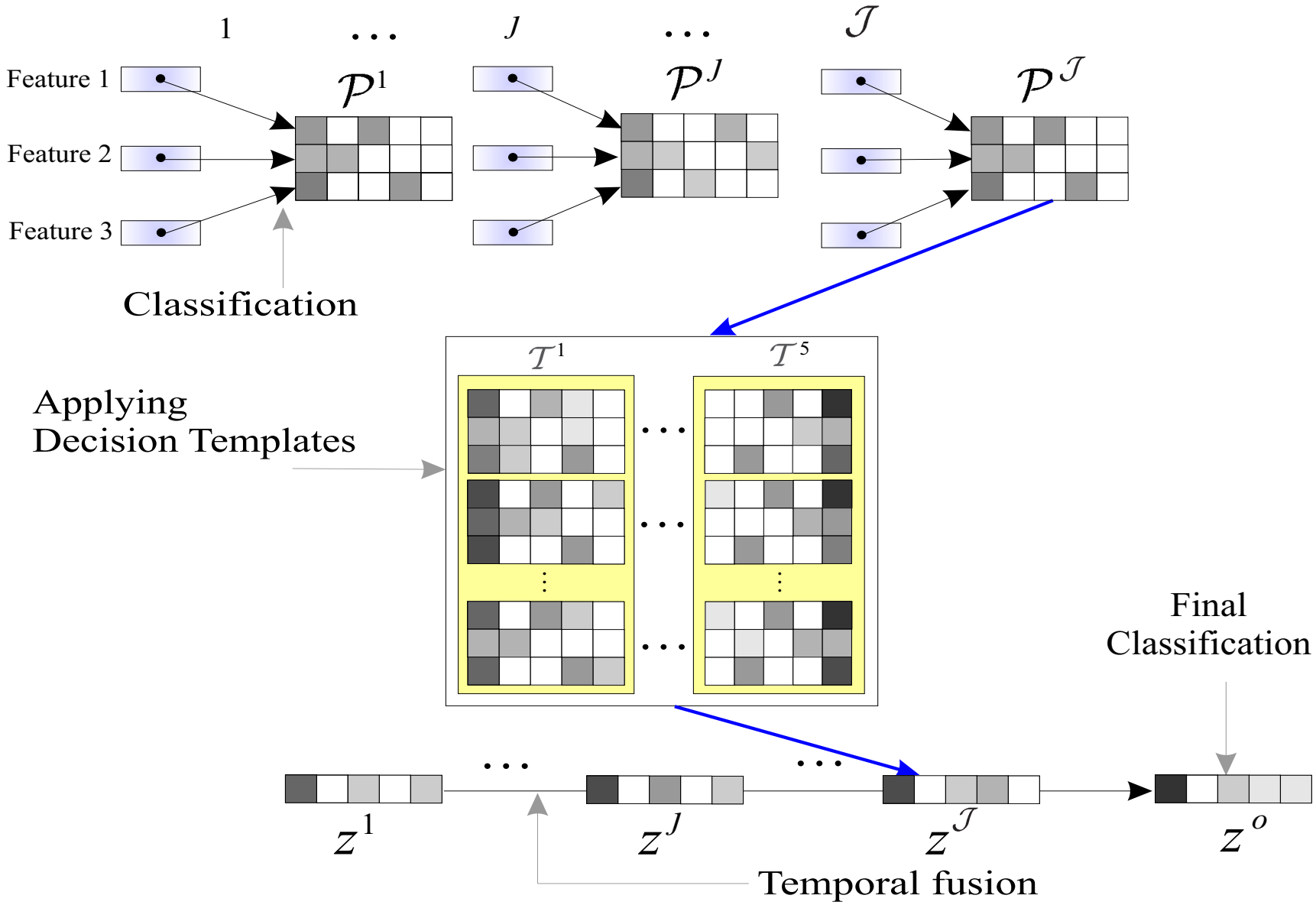
MDT - Klassifikation



MDT - Klassifikation



MDT - Klassifikation



Ergebnisse

Ergebnisse

Neuroinformatik

- Einordnung von Decision Templates in gängige Verfahren der Neuroinformatik (Assoziativspeicher, Naive Bayes Fusion und Pseudoinverse Matrix)

Ergebnisse

Neuroinformatik

- Einordnung von Decision Templates in gängige Verfahren der Neuroinformatik (Assoziativspeicher, Naive Bayes Fusion und Pseudoinverse Matrix)
- Algorithmen zur Initialisierung von Radialen Basis Funktionen Netzwerke mit Entscheidungsbäumen [Kubat 98], [Schwenker, Dietrich 00]

Ergebnisse

Bioakustik

- Verfahren zur automatischen Klassifikation von Orthopteren, basierend auf der biologischen Systematik und der Fusion über mehrere Zeitskalen [Dietrich, Schwenker, Riede, Palm 01]

Ergebnisse

Bioakustik

- Verfahren zur automatischen Klassifikation von Orthopteren, basierend auf der biologischen Systematik und der Fusion über mehrere Zeitskalen [Dietrich, Schwenker, Riede, Palm 01]
- Segmentierung und Extraktion biologisch relevanter Merkmale

Ergebnisse

Bioakustik

- Verfahren zur automatischen Klassifikation von Orthopteren, basierend auf der biologischen Systematik und der Fusion über mehrere Zeitskalen [Dietrich, Schwenker, Riede, Palm 01]
- Segmentierung und Extraktion biologisch relevanter Merkmale
- Prototyp eines Gesangerkennungssystems zur Klassifikation von Orthopteren in C/C++, basierend auf der Matlab C++-Library

Ergebnisse

Bioakustik

- Verfahren zur automatischen Klassifikation von Orthopteren, basierend auf der biologischen Systematik und der Fusion über mehrere Zeitskalen [Dietrich, Schwenker, Riede, Palm 01]
- Segmentierung und Extraktion biologisch relevanter Merkmale
- Prototyp eines Gesangerkennungssystems zur Klassifikation von Orthopteren in C/C++, basierend auf der Matlab C++-Library
- Bewertung akustischer Merkmale durch vollautomatische Merkmalsselektion

Ergebnisse

Bioakustik

- Verfahren zur automatischen Klassifikation von Orthopteren, basierend auf der biologischen Systematik und der Fusion über mehrere Zeitskalen [Dietrich, Schwenker, Riede, Palm 01]
- Segmentierung und Extraktion biologisch relevanter Merkmale
- Prototyp eines Gesangerkennungssystems zur Klassifikation von Orthopteren in C/C++, basierend auf der Matlab C++-Library
- Bewertung akustischer Merkmale durch vollautomatische Merkmalsselektion

Zeitreihenklassifikation mit MCS

- Vergleich von Fusionsarchitekturen zur Klassifikation von Zeitreihen (CFT-Architektur als statische Referenzarchitektur zu DTs) [Dietrich, Schwenker, Palm 01]

Ergebnisse

Bioakustik

- Verfahren zur automatischen Klassifikation von Orthopteren, basierend auf der biologischen Systematik und der Fusion über mehrere Zeitskalen [Dietrich, Schwenker, Riede, Palm 01]
- Segmentierung und Extraktion biologisch relevanter Merkmale
- Prototyp eines Gesangerkennungssystems zur Klassifikation von Orthopteren in C/C++, basierend auf der Matlab C++-Library
- Bewertung akustischer Merkmale durch vollautomatische Merkmalsselektion

Zeitreihenklassifikation mit MCS

- Vergleich von Fusionsarchitekturen zur Klassifikation von Zeitreihen (CFT-Architektur als statische Referenzarchitektur zu DTs) [Dietrich, Schwenker, Palm 01]
- Multiple Decision Template (MDT) Algorithmus zur Klassifikation von Zeitreihen (Verbesserung der Klassifikationsleistung, falls Verwechslungen mit mehreren Klassen abwechselnd auftreten) [Dietrich, Palm, Schwenker 02]

Ergebnisse

Bioakustik

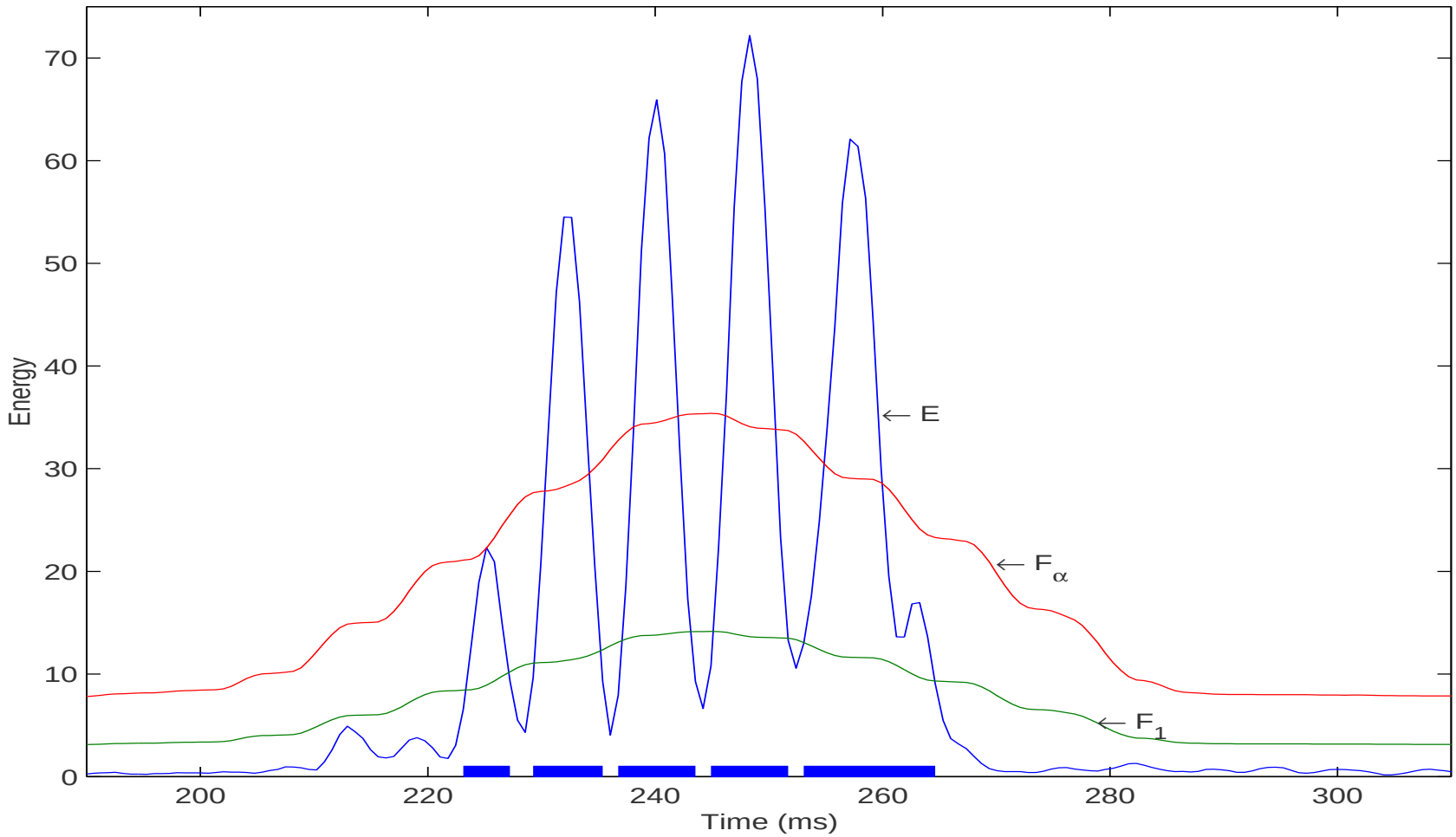
- Verfahren zur automatischen Klassifikation von Orthopteren, basierend auf der biologischen Systematik und der Fusion über mehrere Zeitskalen [Dietrich, Schwenker, Riede, Palm 01]
- Segmentierung und Extraktion biologisch relevanter Merkmale
- Prototyp eines Gesangerkennungssystems zur Klassifikation von Orthopteren in C/C++, basierend auf der Matlab C++-Library
- Bewertung akustischer Merkmale durch vollautomatische Merkmalsselektion

Zeitreihenklassifikation mit MCS

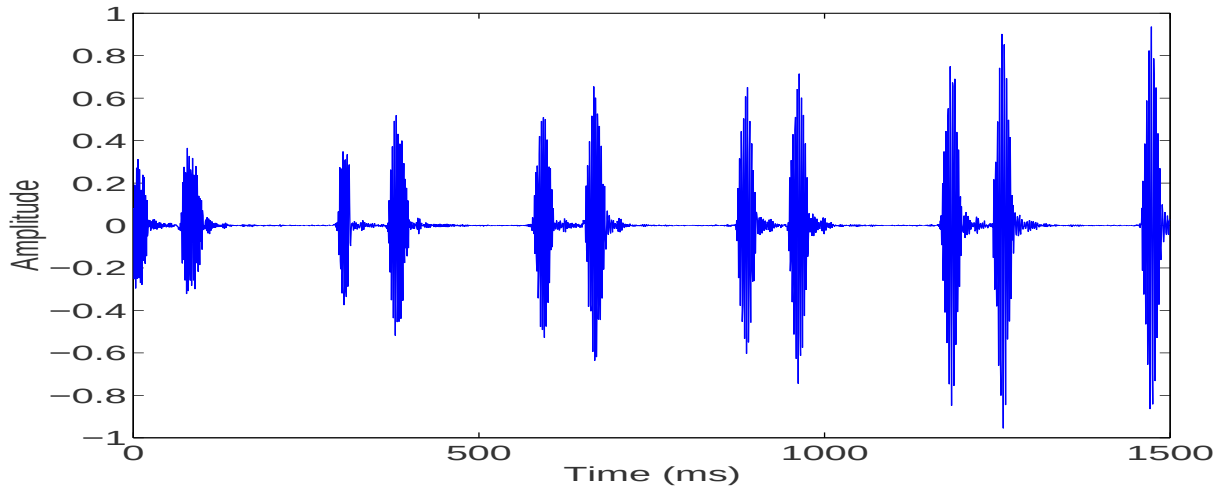
- Vergleich von Fusionsarchitekturen zur Klassifikation von Zeitreihen (CFT-Architektur als statische Referenzarchitektur zu DTs) [Dietrich, Schwenker, Palm 01]
- Multiple Decision Template (MDT) Algorithmus zur Klassifikation von Zeitreihen (Verbesserung der Klassifikationsleistung, falls Verwechslungen mit mehreren Klassen abwechselnd auftreten) [Dietrich, Palm, Schwenker 02]

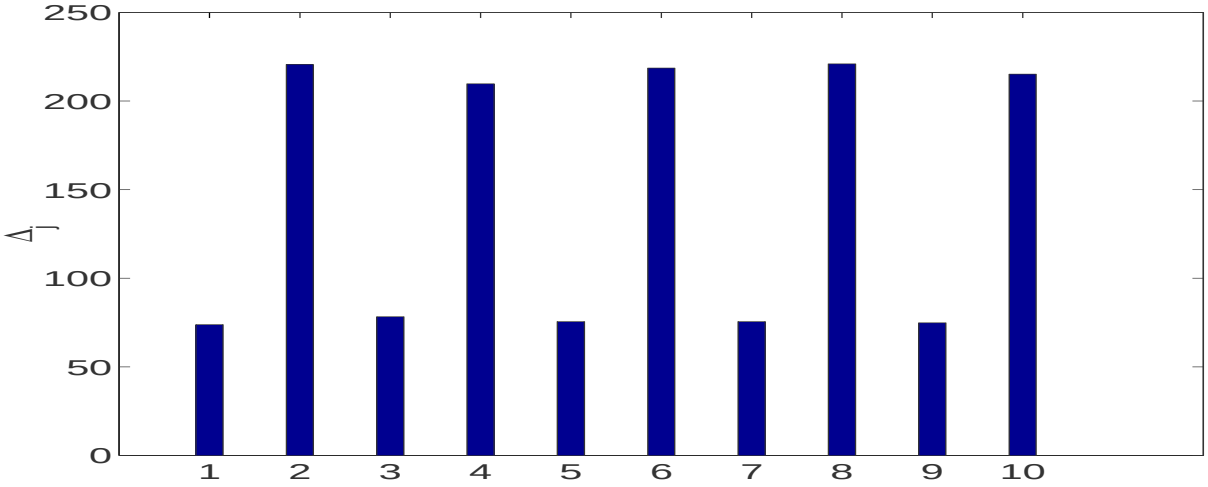
Signalsegmentierung - Pulsdetektion: *Zvenella transversa*

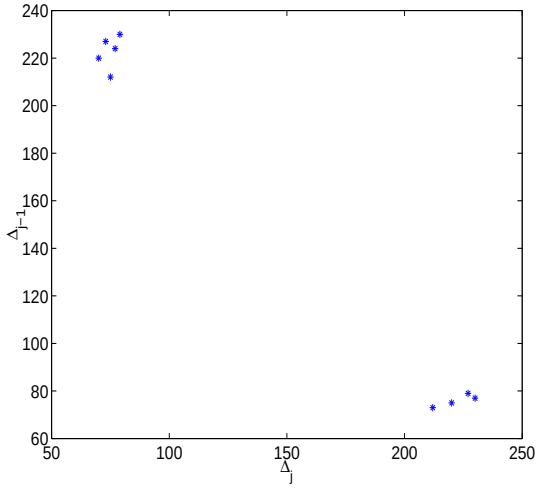
Idee: [Rabiner, Sambur 75]



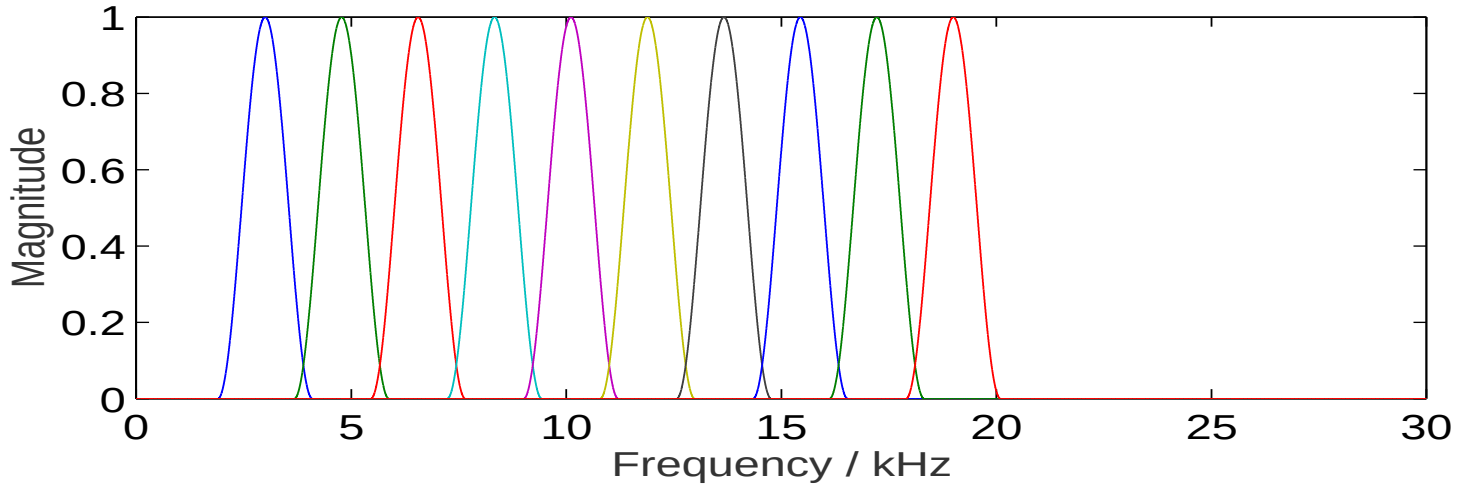
Zeitstruktur: *Noctitrella glabra*



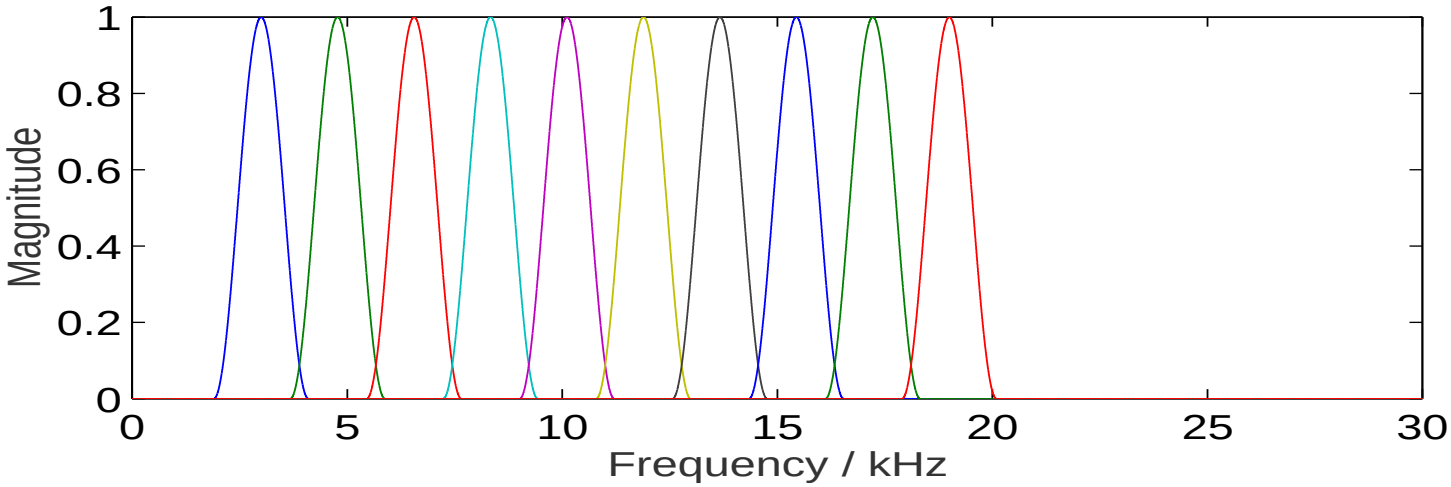


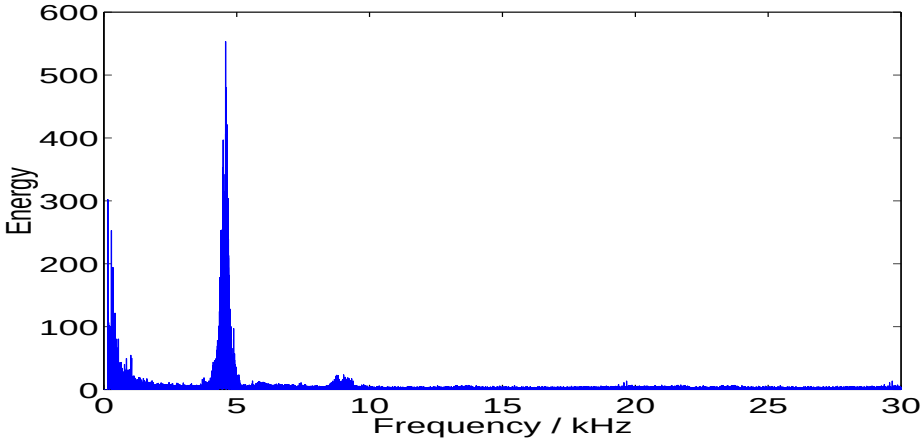


Filterbankenergien

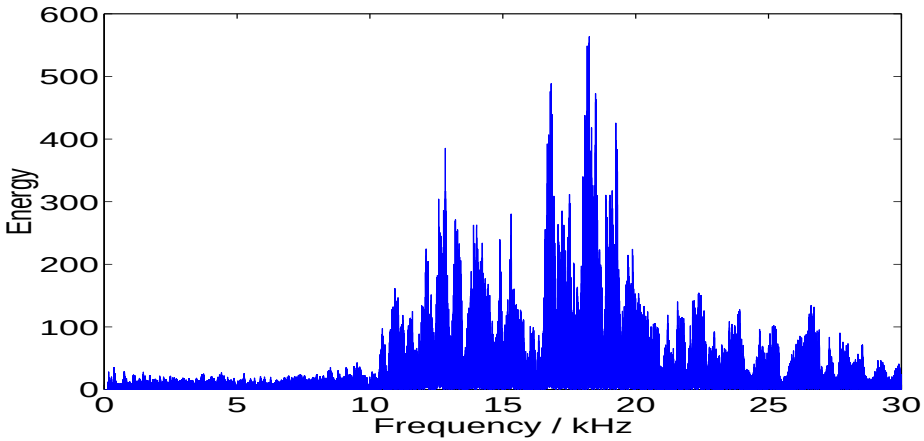


Filterbankenergien





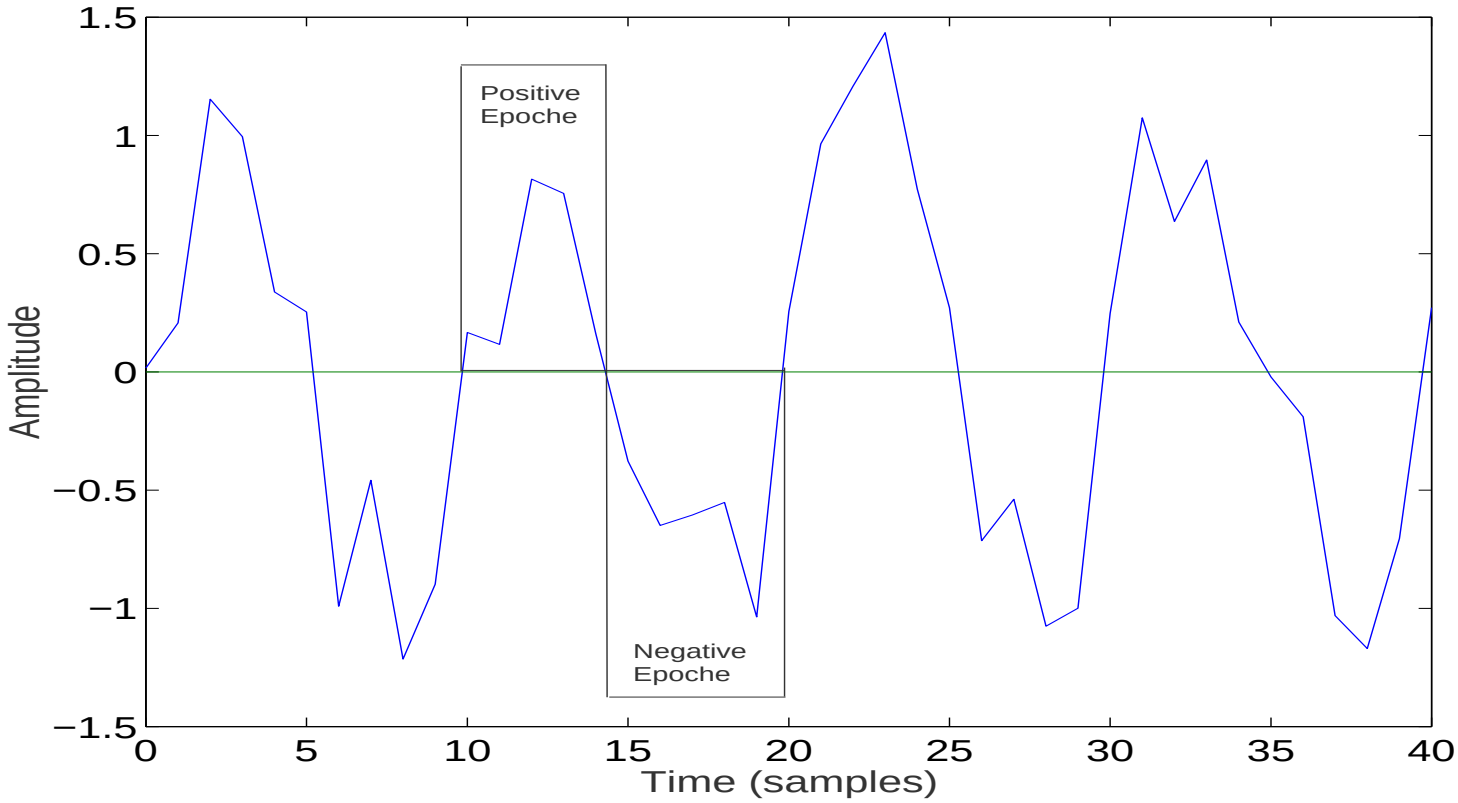
(a) *Homoeoxipha lycoides*



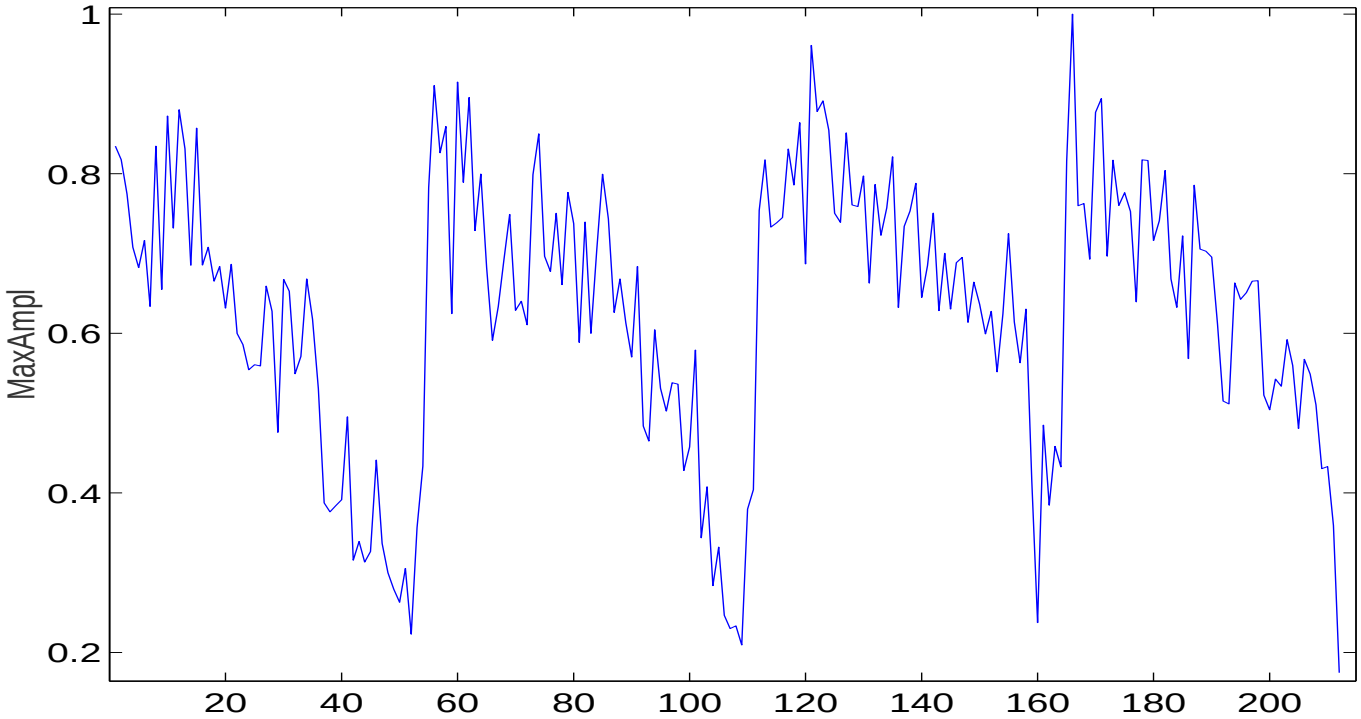
(b) *Ephippiger ephippiger*

Time Encoded Signals (TES)

[King, Gosling 78]

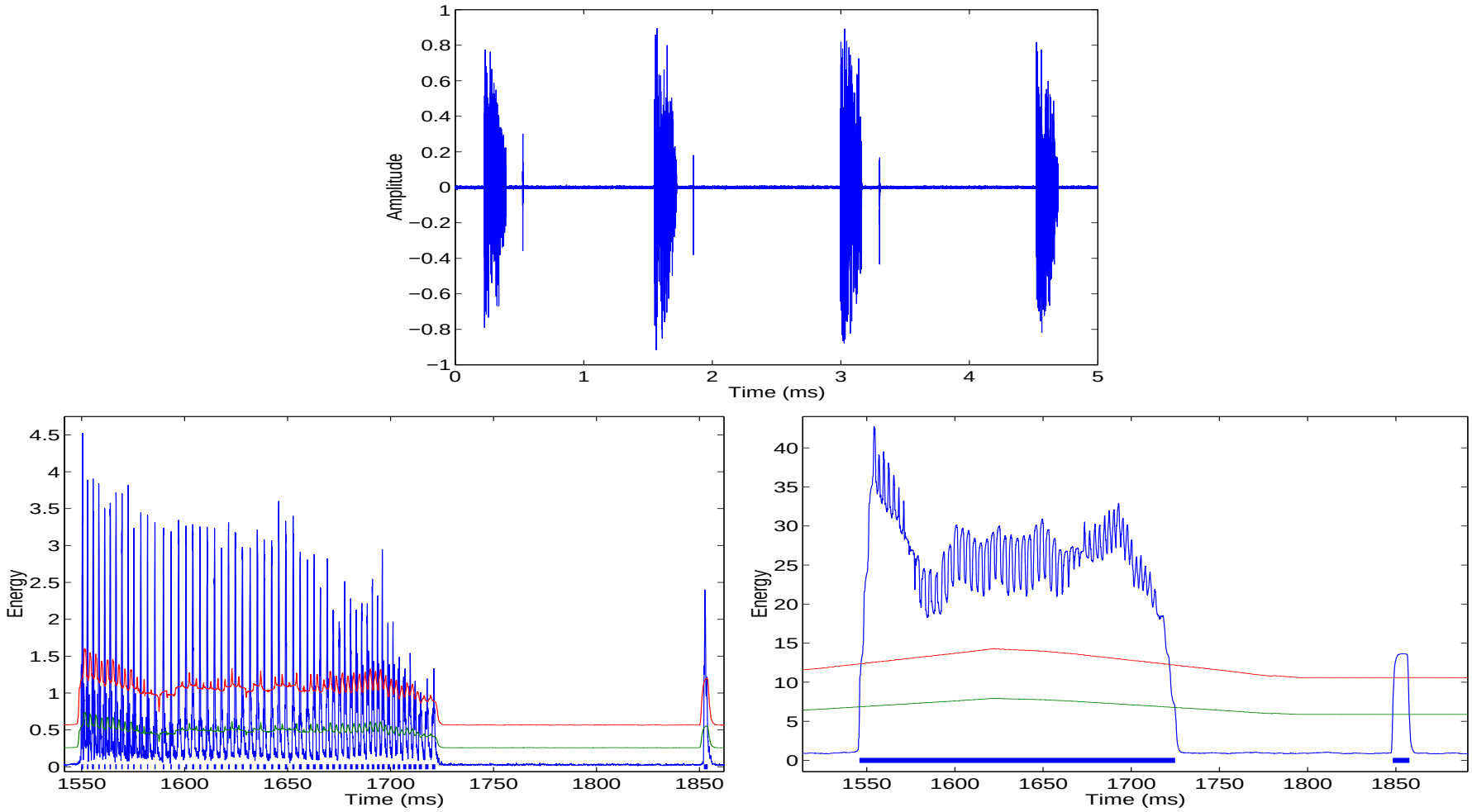


Amplitudenverlauf *Isophya modestior*



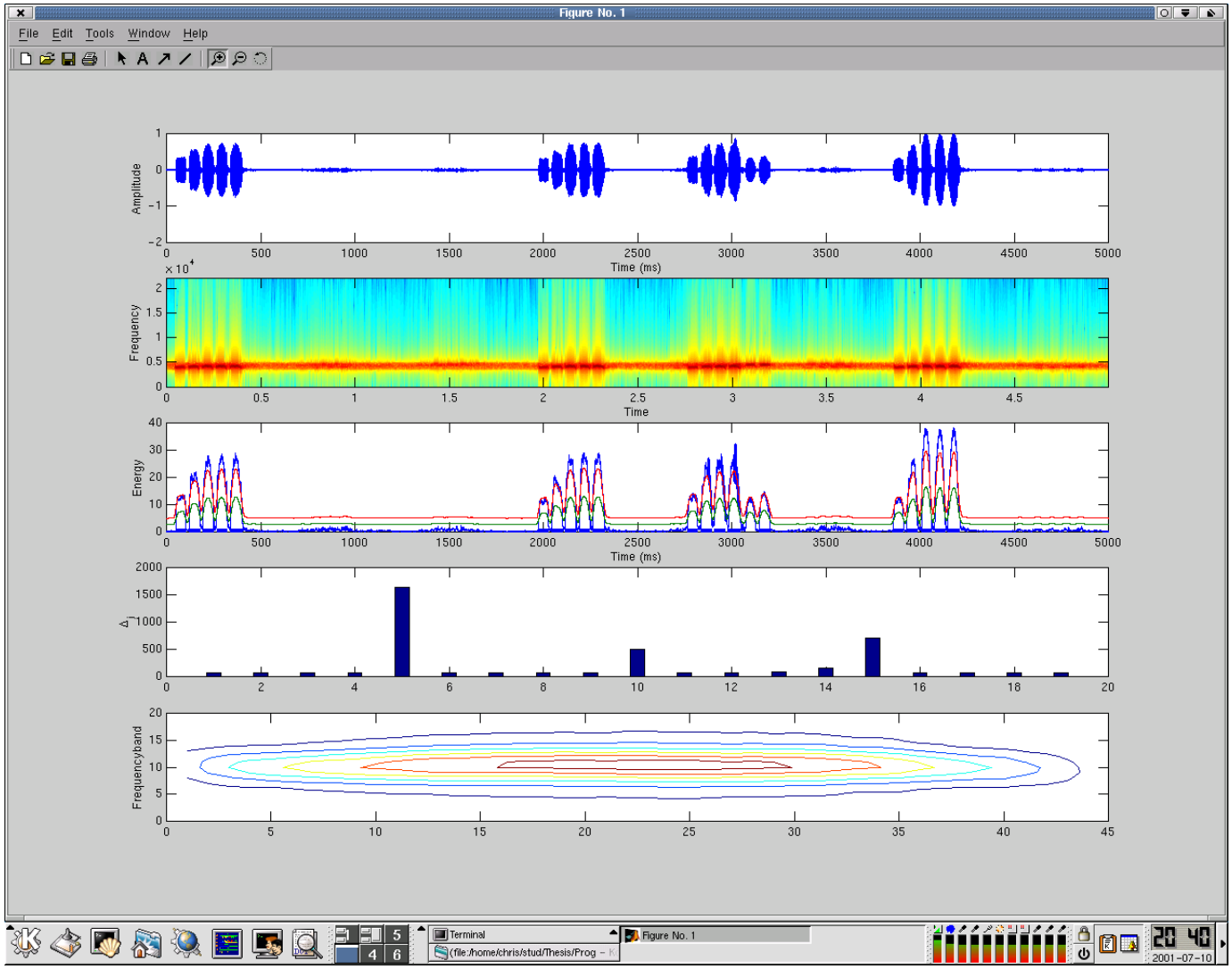
←

Extraktion von Merkmalen in verschiedenen Zeitskalen

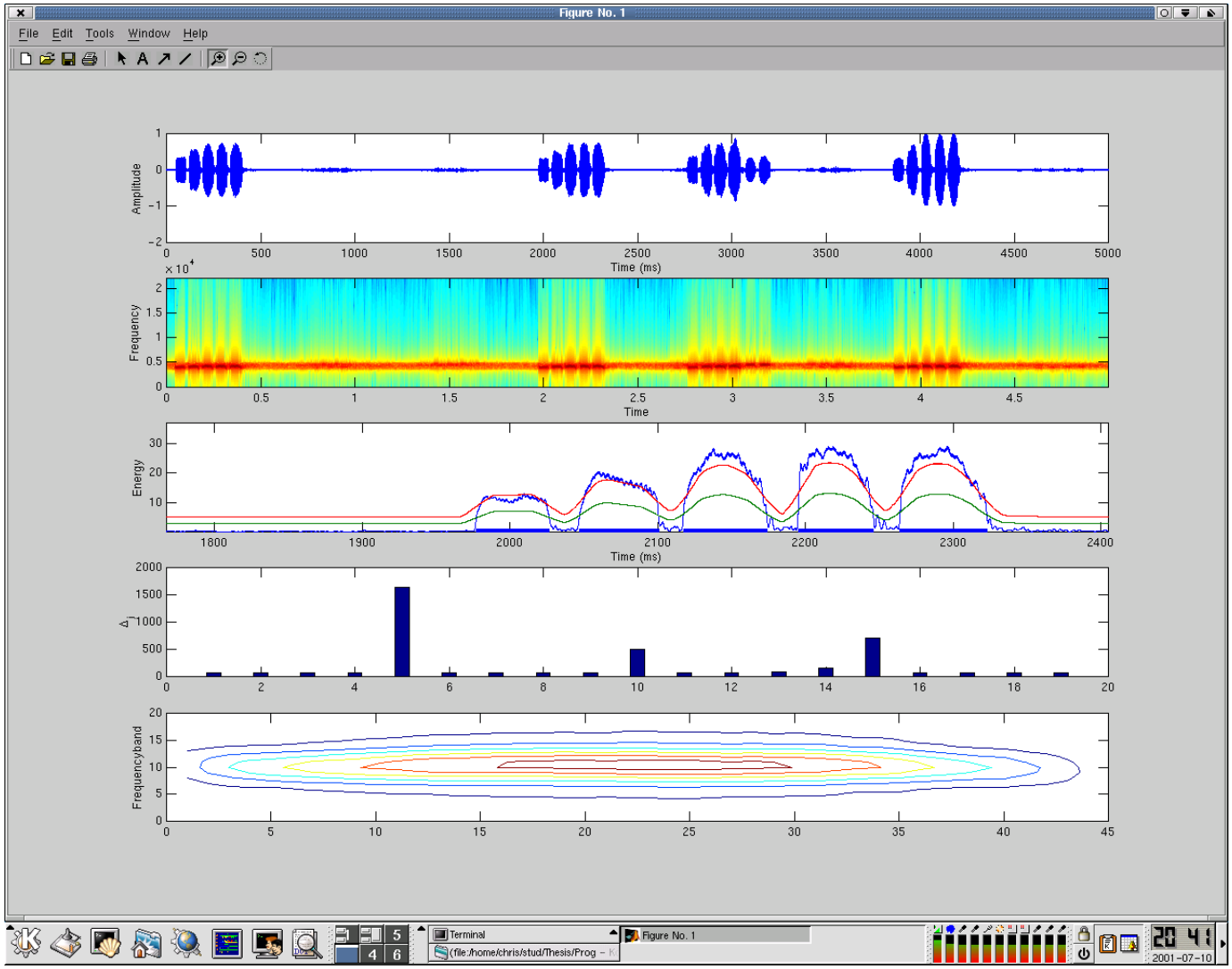


Isophya modestior (Signalausschnitt 5000 ms) ←

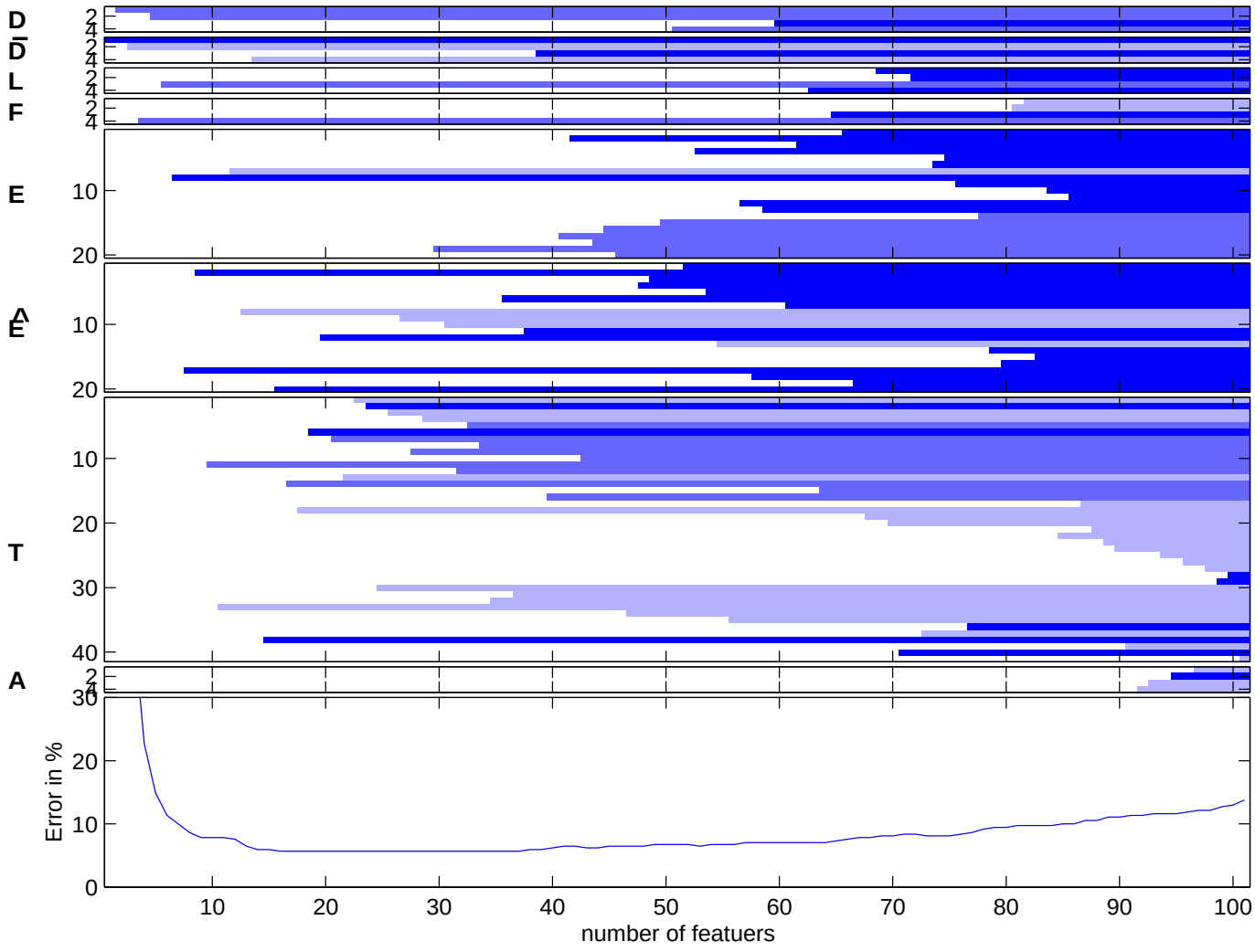
Visualisierung mit Matlab



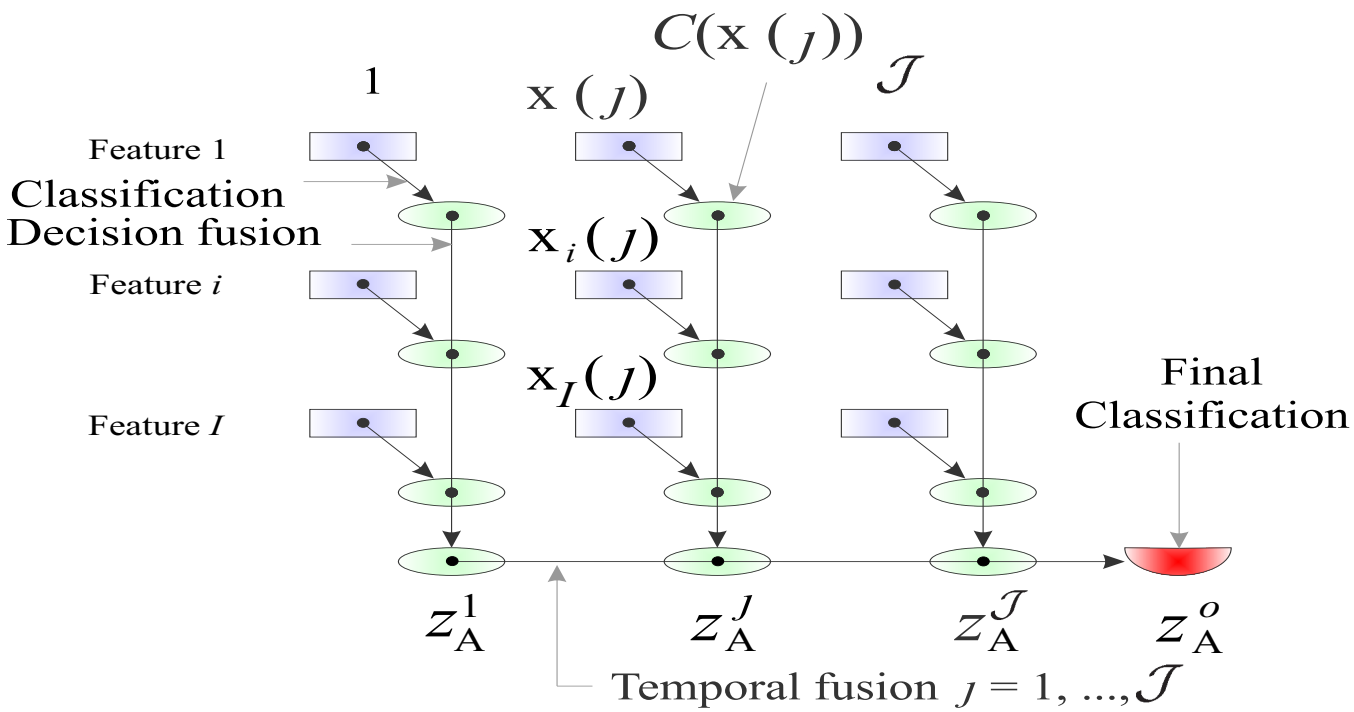
Visualisierung mit Matlab



Bewertung der Merkmale durch Vorwärtsselektion



CFT-Architektur



CFT:

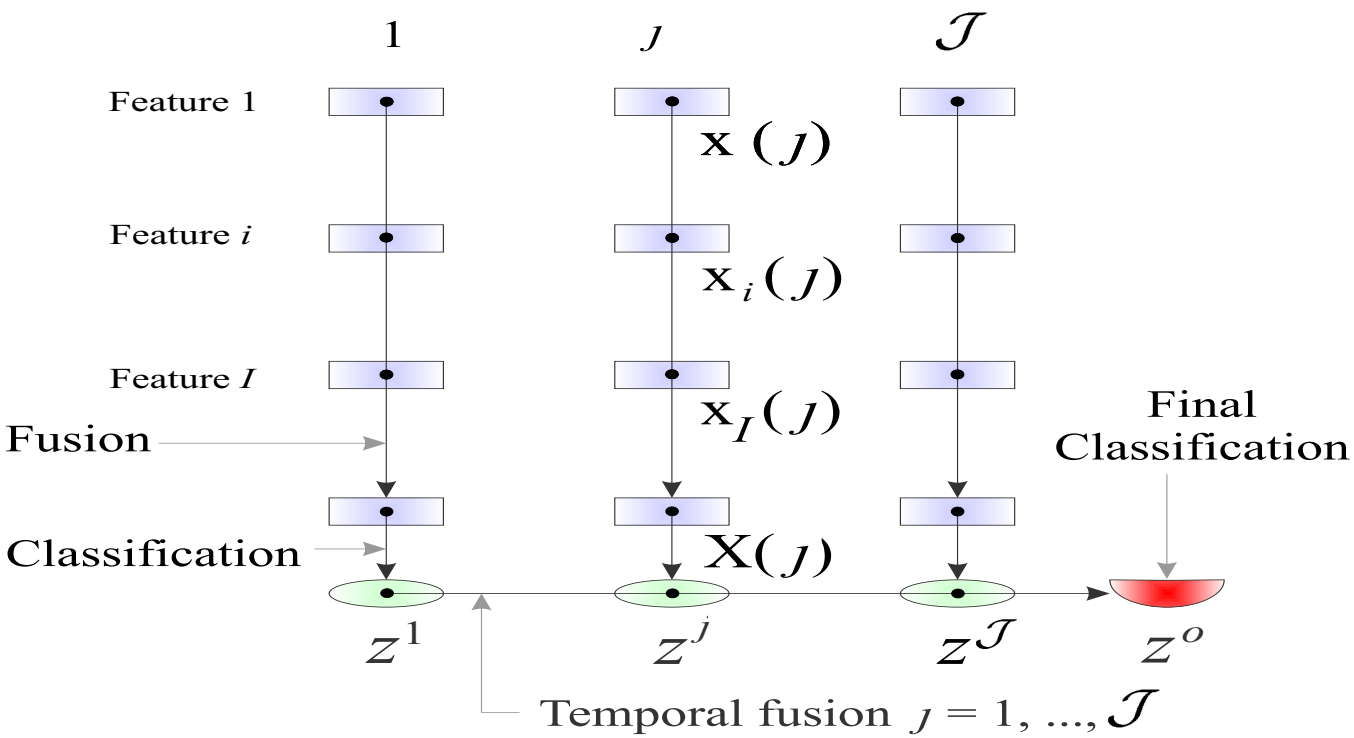
Classification: $\mathcal{C}^i : \mathbb{R}^{d_i} \rightarrow \Delta,\ i = 1 \ldots I$

$\mathcal{C}^1(\mathbf{x}_1(\jmath)) \ldots \mathcal{C}^I(\mathbf{x}_I(\jmath))$

Fusion: $\mathbf{z}_A^{\jmath} := \mathcal{F}(\mathcal{C}^1(\mathbf{x}_1(\jmath)) \ldots \mathcal{C}^I(\mathbf{x}_I(\jmath)))$

Temporal integration: $\mathbf{z}_A^{o} := \mathcal{F}(\mathbf{z}^1 \ldots \mathbf{z}^{\mathcal{J}})$

FCT-Architektur



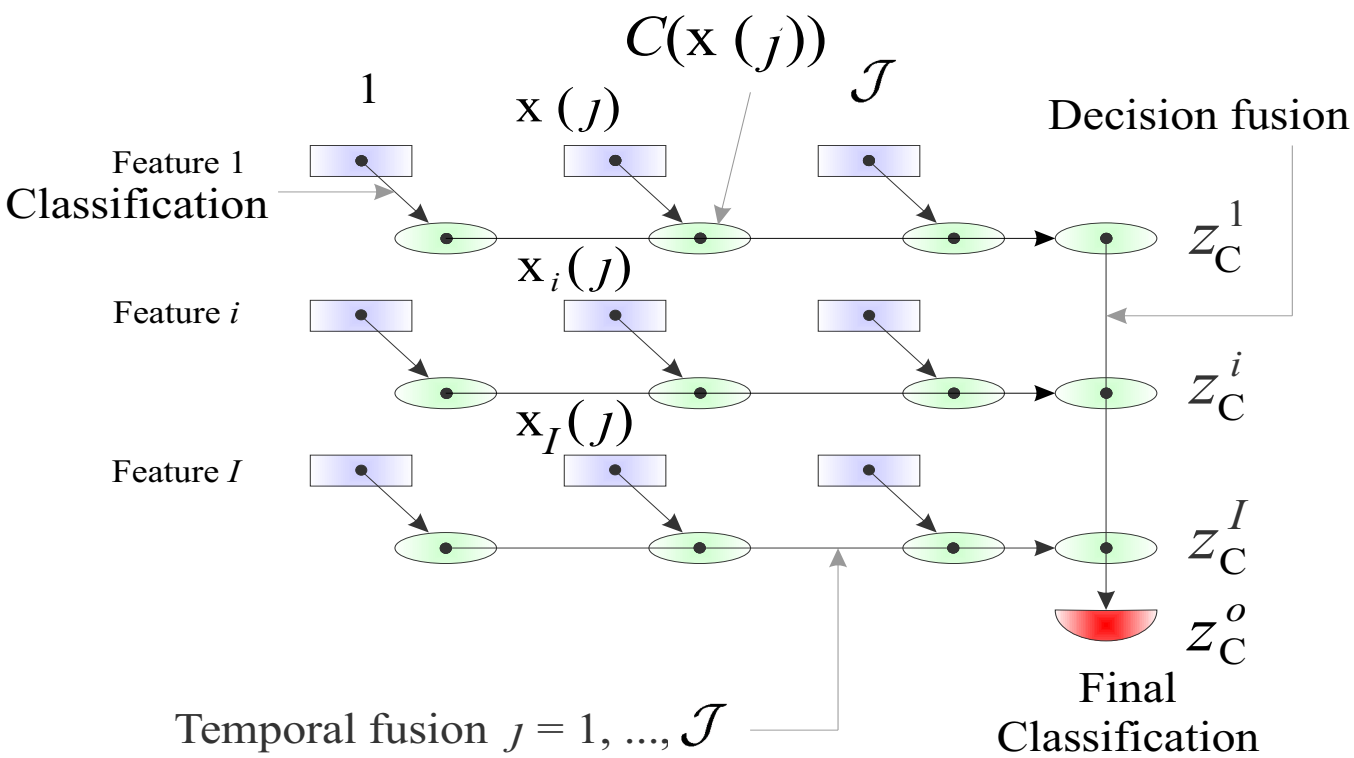
FCT:

Fusion: $X(\jmath) = (\mathbf{x}_1(\jmath) \ldots \mathbf{x}_I(\jmath)) \in \mathbb{R}^\Phi,\ \Phi = \sum_{i=1}^{I} d_i$

Classification: $\mathbf{z}_B^\jmath := \mathcal{C}^\jmath(X(\jmath))$

Temporal integration: $\mathbf{z}_B^o := \mathcal{F}(\mathbf{z}_B^1 \ldots \mathbf{z}_B^{\mathcal{J}})$

CTF-Architektur



CTF:

Classification: $\mathbf{x}_i(\jmath) \rightarrow \mathcal{C}^i(\mathbf{x}_i(\jmath)) \in \Delta$

Temporal integration: $\mathbf{z}_C^i := \mathcal{F}(\mathcal{C}^i(\mathbf{x}_i(1)) \ldots \mathcal{C}^i(\mathbf{x}_i(\mathcal{J}))) \in \Delta$

Fusion: $\mathbf{z}_C^o := \mathcal{F}(\mathbf{z}_C^1 \ldots \mathbf{z}_C^I)$

$\boxed{\leftarrow}$

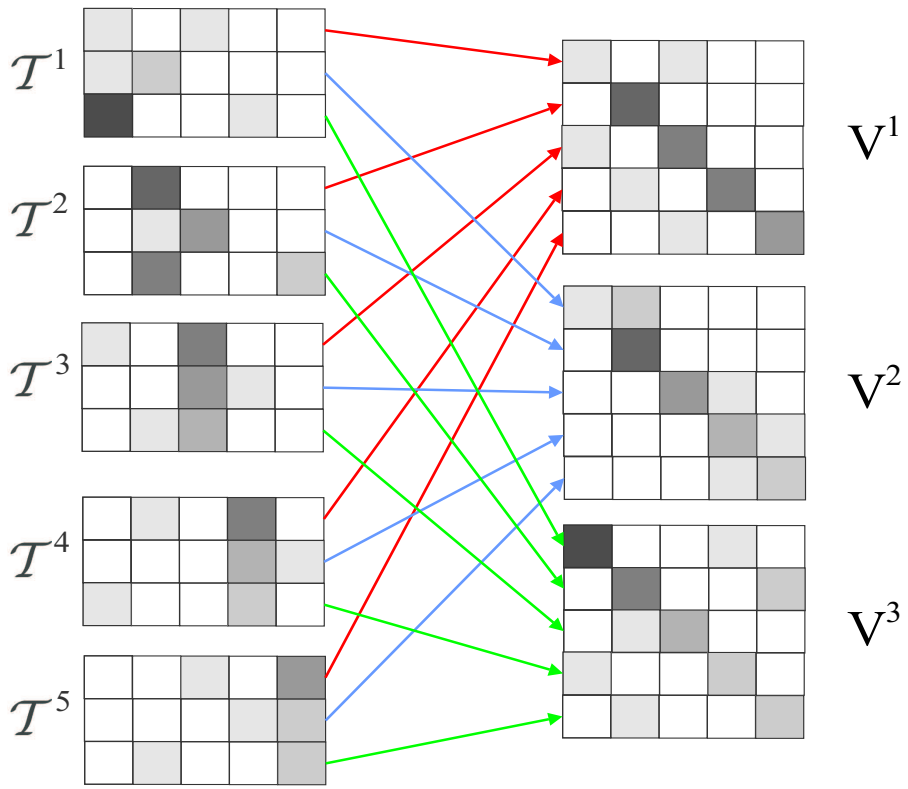
Einordnung von Decision Templates

Einordnung von Decision Templates

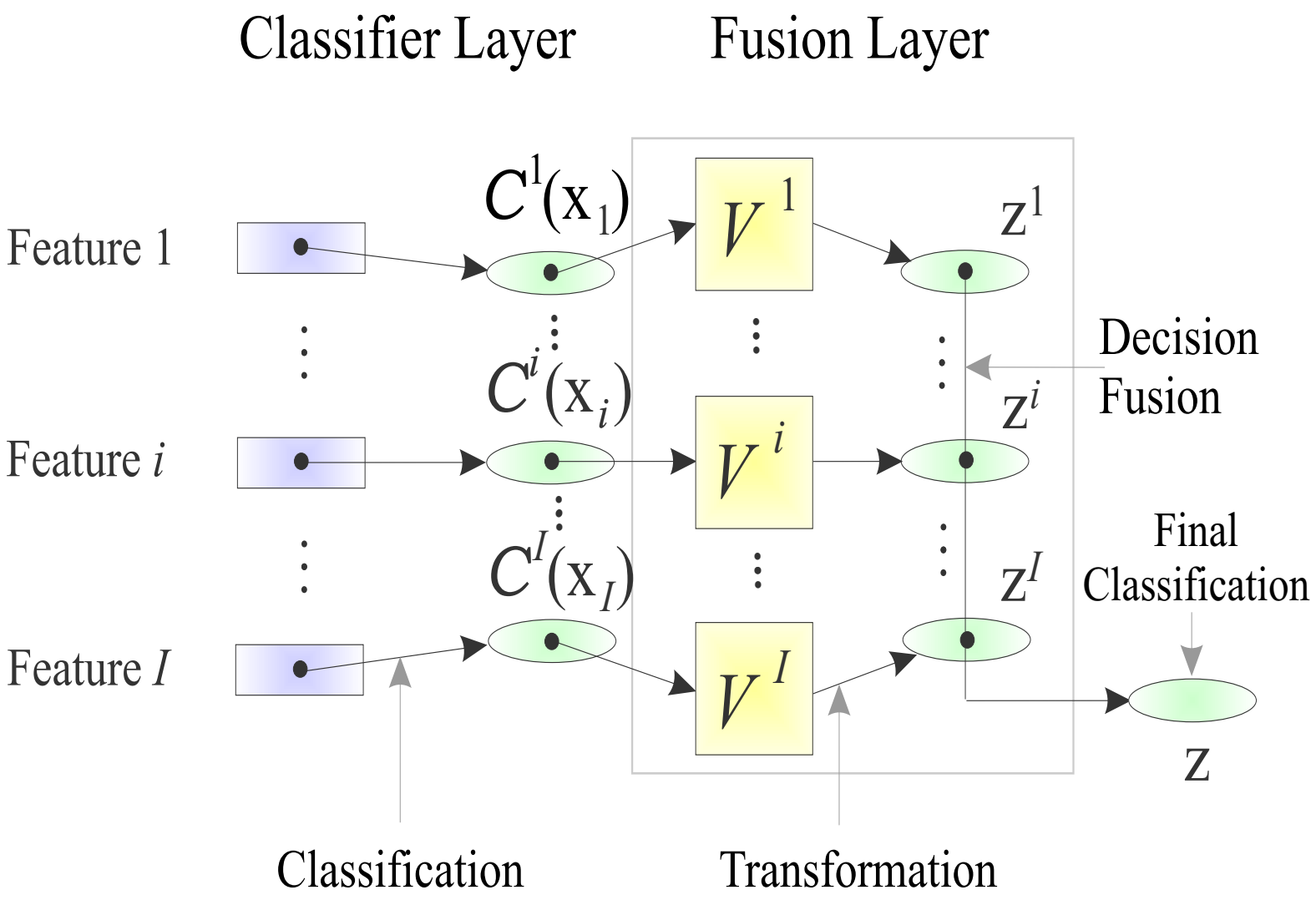
$$\mathcal{S}(\mathcal{P}, \mathcal{T}^\omega) = \frac{1}{I} \sum_{i=1}^{I} \langle \mathcal{P}_{i,\cdot}, \mathcal{T}^\omega_{i,\cdot} \rangle \qquad \mathcal{T}^\omega := \begin{bmatrix} \mathcal{T}^\omega_1 \\ \vdots \\ \mathcal{T}^\omega_I \end{bmatrix} \qquad V^i = \begin{bmatrix} \mathcal{T}^1_i \\ \vdots \\ \mathcal{T}^L_i \end{bmatrix}$$

Einordnung von Decision Templates

$$\mathcal{S}(\mathcal{P}, \mathcal{T}^\omega) = \frac{1}{I} \sum_{i=1}^{I} \langle \mathcal{P}_{i,\cdot}, \mathcal{T}^\omega_{i,\cdot} \rangle \qquad \mathcal{T}^\omega := \begin{bmatrix} \mathcal{T}^\omega_1 \\ \vdots \\ \mathcal{T}^\omega_I \end{bmatrix} \qquad V^i = \begin{bmatrix} \mathcal{T}^1_i \\ \vdots \\ \mathcal{T}^L_i \end{bmatrix}$$



Einordnung von Decision Templates



Fusionsmethoden mit linearen Matrixoperatoren

Klassifikatorausgaben: $(L \times M)$-Matrix $C_i = (C_i^\mu)_{\mu=1}^M$ mit $C_i^\mu \in \Delta^T$

Soll-Ausgabe: $(L \times M)$-Matrix Y mit $Y_{l,\mu} = \begin{cases} 1 & , l = \omega^\mu \\ 0 & , \text{otherwise} \end{cases}$

Fusionsmethoden mit linearen Matrixoperatoren

Klassifikatorausgaben: $(L \times M)$-Matrix $C_i = (C_i^\mu)_{\mu=1}^M$ mit $C_i^\mu \in \Delta^T$

Soll-Ausgabe: $(L \times M)$-Matrix Y mit $Y_{l,\mu} = \begin{cases} 1 & , l = \omega^\mu \\ 0 & , \text{otherwise} \end{cases}$

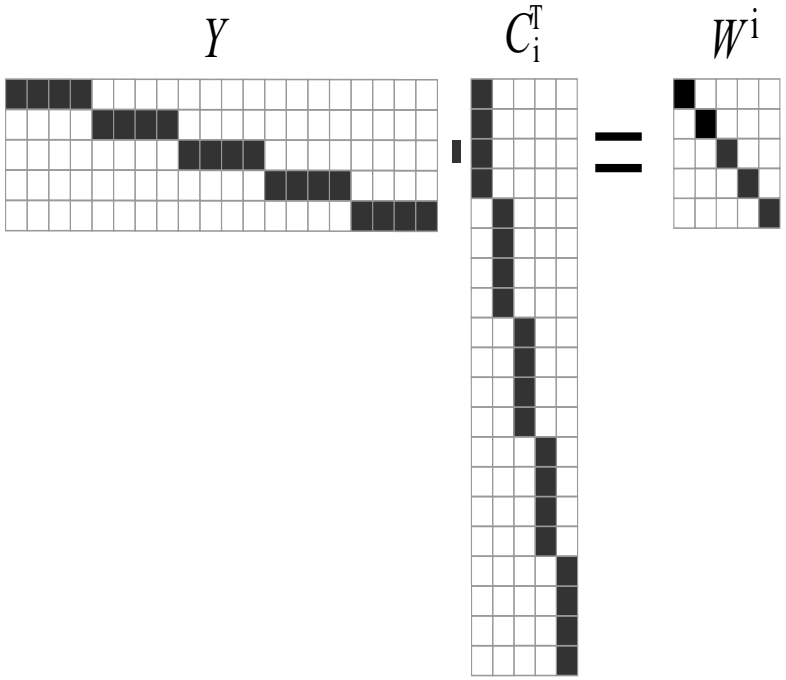
Verwechslungsmatrix: $W^i := Y C_i^T$

Fusionsmethoden mit linearen Matrixoperatoren

Klassifikatorausgaben: $(L \times M)$-Matrix $C_i = (C_i^\mu)_{\mu=1}^M$ mit $C_i^\mu \in \Delta^T$

Soll-Ausgabe: $(L \times M)$-Matrix Y mit $Y_{l,\mu} = \begin{cases} 1 & , l = \omega^\mu \\ 0 & , \text{otherwise} \end{cases}$

Verwechslungsmatrix: $W^i := Y C_i^T$

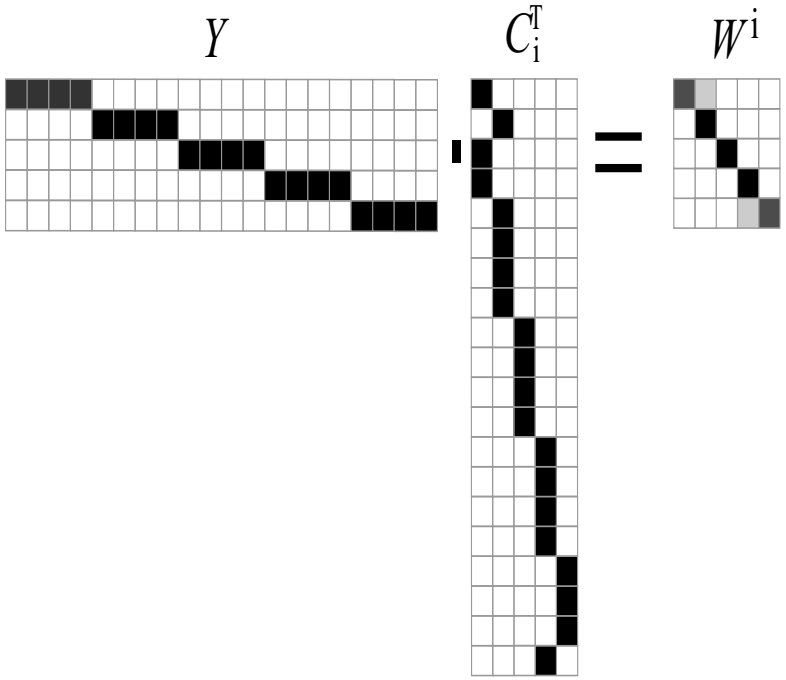


Fusionsmethoden mit linearen Matrixoperatoren

Klassifikatorausgaben: $(L \times M)$-Matrix $C_i = (C_i^\mu)_{\mu=1}^M$ mit $C_i^\mu \in \Delta^T$

Soll-Ausgabe: $(L \times M)$-Matrix Y mit $Y_{l,\mu} = \begin{cases} 1 & , l = \omega^\mu \\ 0 & , \text{otherwise} \end{cases}$

Verwechslungsmatrix: $W^i := Y C_i^T$



Fusionsmethoden mit linearen Matrixoperatoren

Klassifikatorausgaben: $(L \times M)$-Matrix $C_i = (C_i^\mu)_{\mu=1}^M$ mit $C_i^\mu \in \Delta^T$

Soll-Ausgabe: $(L \times M)$-Matrix Y mit $Y_{l,\mu} = \begin{cases} 1 & , l = \omega^\mu \\ 0 & , \text{otherwise} \end{cases}$

Verwechslungsmatrix: $W^i := Y C_i^T$

Korrelationsmatrix: [Kohonen 83] $V^i := W^i$

Fusionsmethoden mit linearen Matrixoperatoren

Klassifikatorausgaben: $(L \times M)$-Matrix $C_i = (C_i^\mu)_{\mu=1}^M$ mit $C_i^\mu \in \Delta^T$

Soll-Ausgabe: $(L \times M)$-Matrix Y mit $Y_{l,\mu} = \begin{cases} 1 & , l = \omega^\mu \\ 0 & , \text{otherwise} \end{cases}$

Verwechslungsmatrix: $W^i := Y C_i^T$

Korrelationsmatrix: [Kohonen 83] $V^i := W^i$

Decision Template: [Kuncheva 95] $V^i := (YY^T)^{-1} W^i$

Fusionsmethoden mit linearen Matrixoperatoren

Klassifikatorausgaben: $(L \times M)$-Matrix $C_i = (C_i^\mu)_{\mu=1}^M$ mit $C_i^\mu \in \Delta^T$

Soll-Ausgabe: $(L \times M)$-Matrix Y mit $Y_{l,\mu} = \begin{cases} 1 & , l = \omega^\mu \\ 0 & , \text{otherwise} \end{cases}$

Verwechslungsmatrix: $W^i := Y C_i^T$

Korrelationsmatrix: [Kohonen 83] $V^i := W^i$

Decision Template: [Kuncheva 95] $V^i := (YY^T)^{-1} W^i$

Naive Bayes Fusion: [Xu et al. 92] $V^i := W^i (C_i C_i^T)^{-1}$

Fusionsmethoden mit linearen Matrixoperatoren

Klassifikatorausgaben: $(L \times M)$-Matrix $C_i = (C_i^\mu)_{\mu=1}^M$ mit $C_i^\mu \in \Delta^T$

Soll-Ausgabe: $(L \times M)$-Matrix Y mit $Y_{l,\mu} = \begin{cases} 1 & , l = \omega^\mu \\ 0 & , \text{otherwise} \end{cases}$

Verwechslungsmatrix: $W^i := Y C_i^T$

Korrelationsmatrix: [Kohonen 83] $V^i := W^i$

Decision Template: [Kuncheva 95] $V^i := (YY^T)^{-1} W^i$

Naive Bayes Fusion: [Xu et al. 92] $V^i := W^i (C_i C_i^T)^{-1}$

Pseudoinverse Matrix: $V^i := W^i (C_i C_i^T)^+$

Fusionsmethoden mit linearen Matrixoperatoren

Klassifikatorausgaben: $(L \times M)$-Matrix $C_i = (C_i^\mu)_{\mu=1}^M$ mit $C_i^\mu \in \Delta^T$

Soll-Ausgabe: $(L \times M)$-Matrix Y mit $Y_{l,\mu} = \begin{cases} 1 & , l = \omega^\mu \\ 0 & , \text{otherwise} \end{cases}$

Verwechslungsmatrix: $W^i := Y C_i^T$

Korrelationsmatrix: [Kohonen 83] $V^i := W^i$

Decision Template: [Kuncheva 95] $V^i := (YY^T)^{-1} W^i$

Naive Bayes Fusion: [Xu et al. 92] $V^i := W^i (C_i C_i^T)^{-1}$

Pseudoinverse Matrix: $V^i := W^i (C_i C_i^T)^+$

$$\hat{Y}_i = V^i \hat{C}_i \qquad \hat{Y} = \frac{1}{I} \sum_{i=1}^{I} \hat{Y}_i$$

Fusionsmethoden mit linearen Matrixoperatoren

Klassifikatorausgaben: $(L \times M)$-Matrix $C_i = (C_i^\mu)_{\mu=1}^M$ mit $C_i^\mu \in \Delta^T$

Soll-Ausgabe: $(L \times M)$-Matrix Y mit $Y_{l,\mu} = \begin{cases} 1 & , l = \omega^\mu \\ 0 & , \text{otherwise} \end{cases}$

Verwechslungsmatrix: $W^i := Y C_i^T$

Korrelationsmatrix: [Kohonen 83] $V^i := W^i$

Decision Template: [Kuncheva 95] $V^i := (YY^T)^{-1} W^i$

Naive Bayes Fusion: [Xu et al. 92] $V^i := W^i (C_i C_i^T)^{-1}$

Pseudoinverse Matrix: $V^i := W^i (C_i C_i^T)^+$

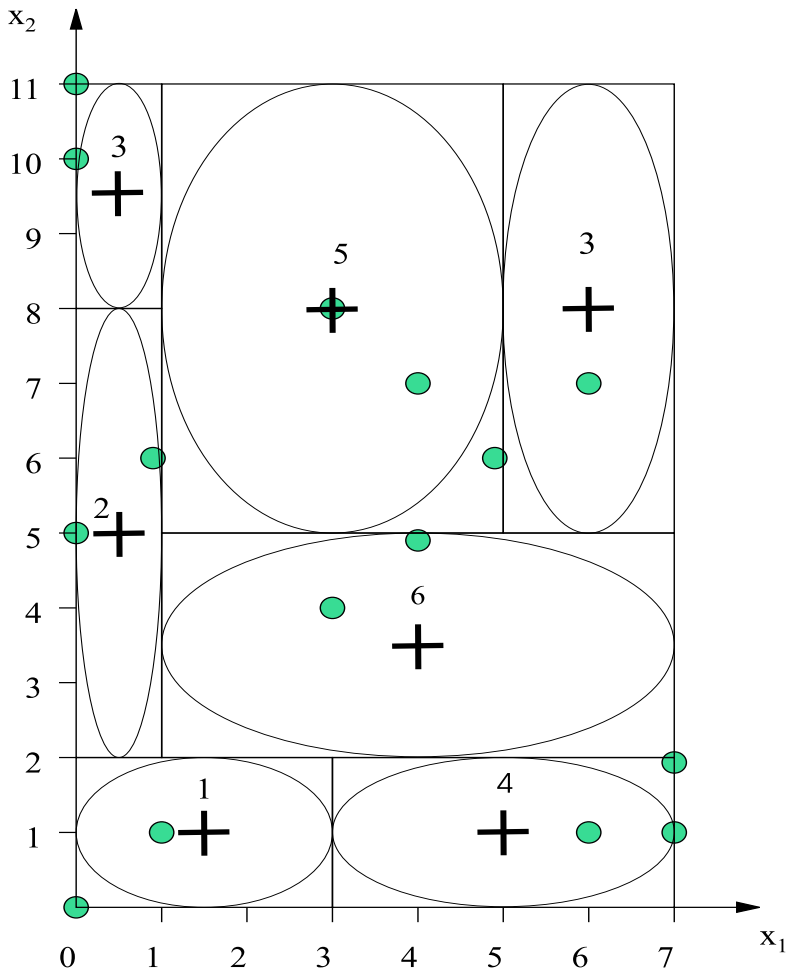
$$\hat{Y}_i = V^i \hat{C}_i \qquad \hat{Y} = \frac{1}{I} \sum_{i=1}^{I} \hat{Y}_i$$

$$\omega_\mu = \operatorname*{argmax}_l (\hat{Y}_{l,\mu}), \quad \mu = 1 \ldots Q$$

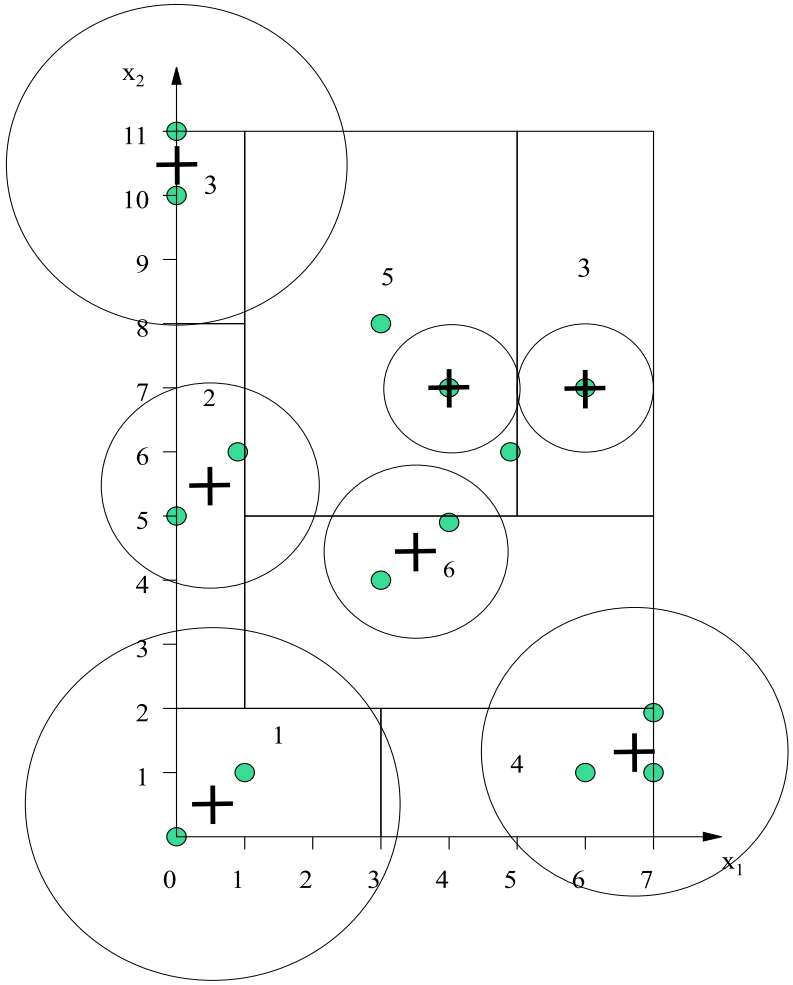
$\boxed{\leftarrow}$

Initialisierung von RBF Netzen mit Entscheidungsbäumen

[Kubat 98]



[Schwenker, Dietrich 00]



Multiple Decision Templates: MDT-Algorithmus

Algorithm $\omega^* = \mathrm{CDTbyMDT}((X(\jmath))_{\jmath=1}^{\mathcal{J}}, (\mathcal{T}^\omega)_{\omega=1}^{L})$

for $\jmath = 1, ..., \mathcal{J}$

(a) $\mathcal{P}^\jmath = [\mathcal{C}^1(\mathbf{x}_1(\jmath)), ..., \mathcal{C}^P(\mathbf{x}_P(\jmath))]^T$

foreach $\omega \in \Omega$

(b) $\mathcal{N}_K(\mathcal{P}^\jmath, \mathcal{T}^\omega) = (\tau_1, ..., \tau_K)$

(c) $\hat{\mathcal{T}}^\omega = \frac{1}{K}\sum_{i=1}^{K} \mathcal{T}^{\omega,\tau_i}$

(d) $C_\omega^\jmath = \mathcal{S}(\mathcal{P}^\jmath, \hat{\mathcal{T}}^\omega)$

end

$C^\jmath = C^\jmath / \sum_{\omega=1}^{L} C_\omega^\jmath$

end

(e) $C^o = \mathcal{F}\{C^1, ..., C^{\mathcal{J}}\}$

$\omega^* = \underset{\omega}{\operatorname{argmax}}(C_\omega^o)$

Algorithm CDTbyMDT: Klassifikation mit Multiple Decision Templates