Research Group
Learning with Neural Methods on Structured Data

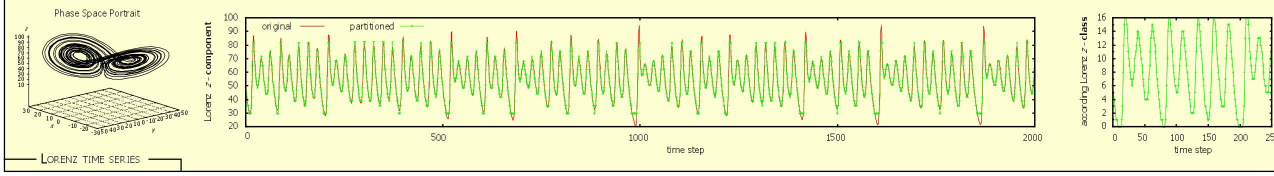
Dept. of Mathematics and Computer Science
University of Osnabrück
Albrechtstraße 28
D - 49069 Osnabrück

UNIVERSITÄT OSNABRÜCK

Modeling Temporal Structure with Neural Methods

Marc Strickert, Thorsten Bojer

http://www.inf.uos.de/lnm/



LORENZ TIME SERIES

GRLVQ generalized relevance learning vector quantization

CLUSTERING TASK

Training Set

Mapping of vectors x^i to their classes y^i:

$$X = \left\{ (x^i, y^i) \in \mathbb{R}^n \times \{1, \dots, C\} \mid i = 1, \dots, m \right\} .$$

Prototype vectors represent the association of input vectors with its class.

prototypes: $W = \left\{ (W^j, c^j) \in \mathbb{R}^n \times \{1, \dots, C\} \mid j = 1, \dots, k \leq m \right\} .$

Prototype Adaptation

Closest *correct* prototype $\vec{W}^1$:

$$\vec{W}^1 := \vec{W}^1 + \epsilon f' \cdot \frac{d_2}{(d_1 + d_2)^2} (x - \vec{W}^1) . \qquad (1)$$

Closest *wrong* prototype $\vec{W}^2$:

$$\vec{W}^2 := \vec{W}^2 - \epsilon f' \cdot \frac{d_1}{(d_1 + d_2)^2} (x - \vec{W}^2) . \qquad (2)$$

$\epsilon \in (0, 1)$ [learning rate]
$f' = \mathrm{sgd}'((d_1 - d_2)/(d_1 + d_2))$ [derivative of the sigmoid]
$d_i = d^2(x, \vec{W}^i, \lambda)$ [*weighted* squared Euclidean distances] :

$$d^2(x, y, \lambda) := \sum_{i=1}^{n} \lambda_i (x_i - y_i)^2 , \quad \lambda_i \geq 0 . \qquad (3)$$

Weight Adaptation

$$\lambda_l := \lambda_l - \epsilon_1 f' \cdot \left(\frac{d_2}{(d_1 + d_2)^2} (x_l^i - W_l^1)^2 - \frac{d_1}{(d_1 + d_2)^2} (x_l^i - W_l^2)^2 \right) . \qquad (4)$$

$\epsilon_1 \in (0, 1)$ [weight adaptation rate]

Weight normalization: $\lambda_l := \lambda_l / |\lambda|$.

Gradient Descent

Error function: $E = \sum_{i} f((d_1 - d_2)/(d_1 + d_2))$.

Pruning Scheme

Set the smallest weight $\lambda_{min} \neq 0$ from a sufficiently trained classification task to constant zero.

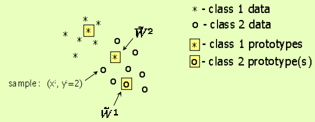
ALGORITHM

initialize prototypes $(W^j, c^j) \in \mathbb{R}^n \times \{1, \dots, C\}$.
initialize $\lambda_1 = \dots = \lambda_n = 1/n$.

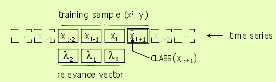
repeat 'training':
  choose training example (x^i, y^i).
  compute $\vec{W}^1$ *and* $\vec{W}^2$ *using metric (3),* $d(W^j, x^i, \lambda)$, $\forall j : c^j = y^i$.
  adapt both prototypes according to (1) and (2) respectively,
  adapt each $\lambda_l \neq 0$ *using (4),*
  set $\lambda_k := 0$ *if* $0 < \lambda_k < EPS$,
  normalize λ
until classification error acceptable.

repeat 'pruning':
  apply pruning scheme,
  redo training
until classification error degrades significantly.

ILLUSTRATION



DATA PROCESSING



BASIC IDEA

Taking into consideration the overall *significance of components for one-step prediction*, we formulate an alternative to the delay embedding strategy: input weighting obtained from a recent *neural* algorithm is used to construct vectors with the time series components of most important relevance.

We apply an adaptation of *Kohonen's* vector quantization scheme (LVQ) with built-in detection of significant input vector components, the **generalized relevance learning vector quantization** (GRLVQ), in order to achieve a relevance based embedding of scalar time series.

Using the neural classification method on time series data we obtain:

(1) *analysis* - temporal properties of a window on the past,
(2) *modeling* - short-term and long-term prediction.

We use artificial data from a chaotic Lorenz System and real-life runoff data of a long-term time series from a large lysimeter.

REFERENCES

1. G. Bontempi, M. Birattari: *A Multi-Step-Ahead Prediction Method Based on Local Dynamic Properties*, in M. Verleysen ed., ESANN'2000, pp. 311-316, D-Facto, 2000.
2. M.C. Casdagli: *Recurrence Plots Revisited*, Physica D 108, pp. 12-44, 1997.
3. G. Dorffner: *Neural Networks for Time Series Processing*, Neural Network World, 6(4), pp. 447-468, 1996.
4. A.M. Fraser, H.L. Swinney: *Independent Coordinates for Strange Attractors from Mutual Information*, Physical Review A 33 (1986), pp. 1134-1140, 1986.
5. B. Hammer, T. Villmann: *Estimating relevant input dimensions for self-organizing algorithms*, to appear in proc. of WSOM 2001.
6. S. Hochreiter, J. Schmidhuber: *Long short-term memory*, Neural Computation, 9(8), pp. 1735-1780, 1997.
7. H. Kantz, T. Schreiber: *Nonlinear time series analysis*, Cambridge University Press, 1997.
8. T. Kohonen: *Self-Organizing Maps*, Springer, 1997.
9. E.N. Lorenz: *Deterministic Nonperiodic Flow*, Journ. of Atmospheric Sc. 20, p. 130f., 1963.
10. A. S. Sato, K. Yamada: *Generalized learning vector quantization*, in G. Tesauro, D. Touretzky, and T. Leen, eds., Advances in Neural Information Processing Systems, Vol. 7, pp. 423-429, MIT Press, 1995.
11. M. Strickert, T. Bojer, B. Hammer: *Generalized Relevance LVQ for Time Series*, to appear in proc. of ICANN 2001.
12. F. Takens: *Detecting Strange Attractors in Turbulence*, Lecture Notes in Mathematics Vol. 898,

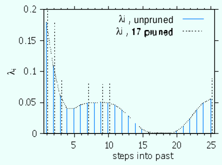
RESULTS

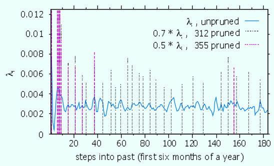
ANALYSIS

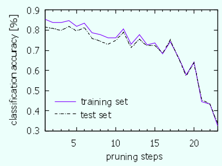
LORENZ

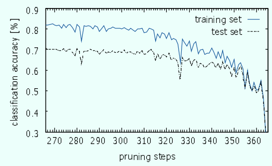
LYSIMETER

Relevance of Past

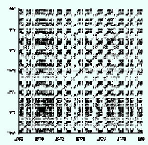


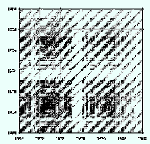






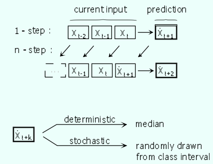
Embedding / Recurrence

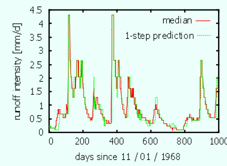


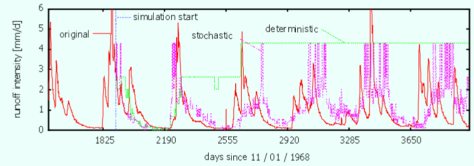


MODELING

Prediction







CONCLUSIONS

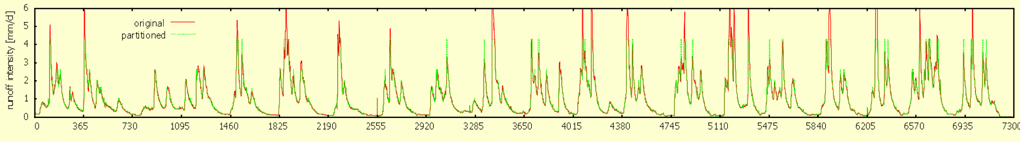
ANALYSIS

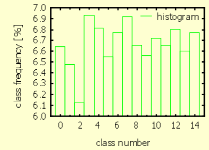
1. Embedding vectors are reduced to reflecting data intrinsic dimensionality -> compact description.
2. Components of the embedding vectors need not be temporally equidistant -> less aliasing than delay embedding.
3. Approach is free of parameters in terms of assumptions concerning functional regression, **but** Intital size of temporal window unclear, making iterated variation necessary.

MODELING

No a-priori modeling assumptions -> very simple models, **but**

(a) Clustering always produces discrete output, thus requiring a-posteriori smoothing.
(b) Training on global data sets disregards instationarities.





LYSIMETER RUNOFF SERIES

05 / 22 / 2001

We gratefully acknowledge the MWK, Lower Saxony, for financial support, and Michael Brandenburg, Dept. of Ecology, Münster, for supplying the runoff data.