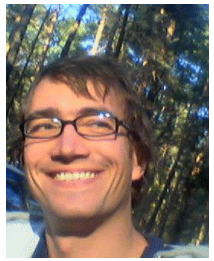
max planck institut
informatik

BIOLOGISCHE KYBERNETIK

Transductive Support Vector Machines for Structured Variables

Alexander Zien Ulf Brefeld Tobias Scheffer



Label Sequence Learning

- Protein secondary structure prediction:

  x = "XSITKTELDG ILPLVARGKV…" → y = " SS TT SS EEEE SS… "

- Named entity recognition (NER):

  x = "Tom comes from London." → y = "Person,-,-,Location"

  x = "S of PTH and CT…" → y = "-,-,Gene,-,Gene,…"

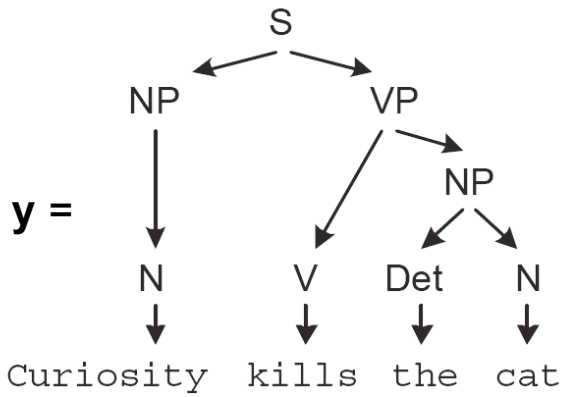
- Part-of-speech (POS) tagging:

  x = "Curiosity kills the cat." → y = "noun, verb, det, noun"

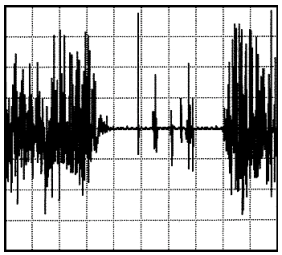
Alexander Zien, Ulf Brefeld, Tobias Scheffer

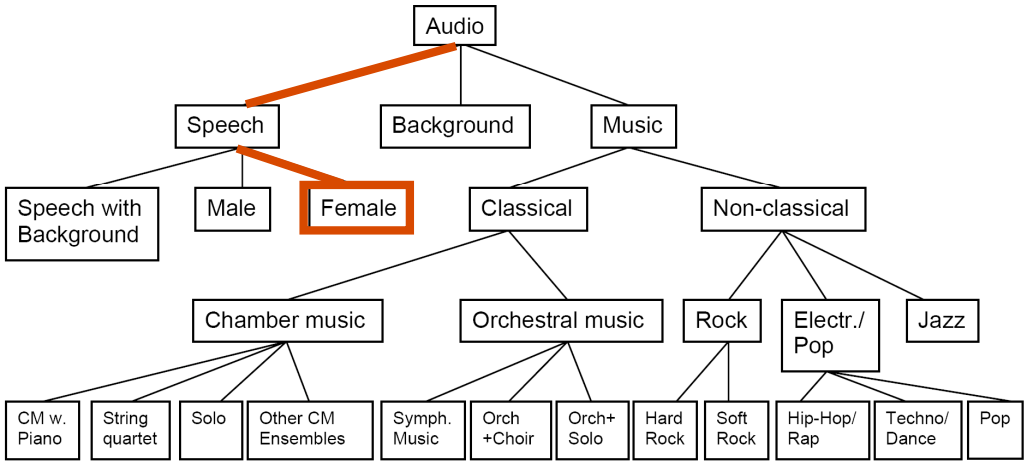
Natural Language Parsing

x = "Curiosity kills the cat" → y =



Classification with Taxonomies

x =  → y =



Alexander Zien, Ulf Brefeld, Tobias Scheffer

Semi-supervised Discriminative Learning

- Labeled training data is scarce and expensive.
  - E.g., experiments in computational biology.
  - Need for expert knowledge.
  - Tedious and time consuming.

- Unclassified instances are abundant and cheap.
  - Extract sentences from www (POS-tagging, NER).
  - Assess primary structure of proteins from DNA/RNA.
  - …

There is a need for semi-supervised techniques in structural learning!

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Overview

- Recap: SVM with structured output variables.
  - Convex optimization problem.
- Transductive SVM with structured variables.
  - Uses information from unlabeled data.
  - Combinatorial, non-convex optimization problem.
- Efficient optimization:
  - Transform, remove discrete parameters.
  - Differentiable, continuous optimization problem.
- Empirical results.

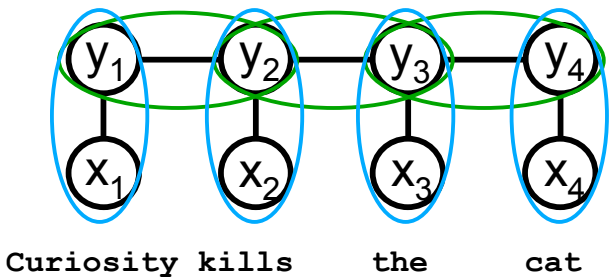
Structural Learning

- Given:
  - Labeled pairs $(\mathbf{x}_1,\mathbf{y}_1), \ldots, (\mathbf{x}_n,\mathbf{y}_n)$, drawn from $P_{\mathcal{X}\mathcal{Y}}$.

- Learn decision function: $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ with low risk.
- Learn generalized linear model:
  - $f(\mathbf{x}, \mathbf{y}) = \langle \mathbf{w}, \Phi(\mathbf{x}, \mathbf{y}) \rangle$
  - with low regularized empirical risk:
  - $\hat{R}(f) = \sum_{i=1}^{n} \ell(\mathbf{x}_i, \mathbf{y}_i, f) + \|f\|^2$
- Inference, decoding:
  - Compute output $\hat{\mathbf{y}} = \operatorname*{argmax}_{\mathbf{y} \in \mathcal{Y}} f(\mathbf{x}, \mathbf{y})$

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Label Sequence Learning

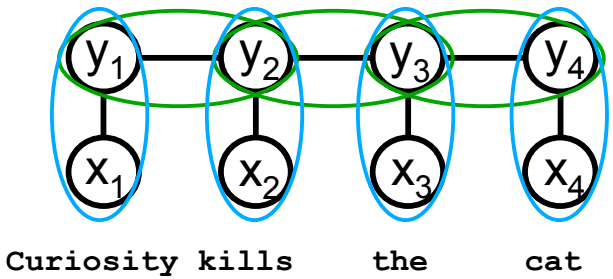


- Attribute for each pair of values of adjacent labels $\mathbf{y}_t$ und $\mathbf{y}_{t+1}$.
  - $\varphi_{123}(\mathbf{y}_t,\mathbf{y}_{t+1})$ = [[$\mathbf{y}_t$="Noun" $\wedge$ $\mathbf{y}_{t+1}$="Verb"]]
- Attribute for each pair of values for input $\mathbf{x}_t$ and output $\mathbf{x}_t$.

  $\bar{\varphi}_{234}(\mathbf{x}_t,\mathbf{y}_t)$ = [[$\mathbf{y}_t$="Noun" $\wedge$ $\mathbf{x}_t$ = "cat"]]

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Label Sequence Learning



- Attribute for each pair of values of adjacent labels $\mathbf{y}_t$ und $\mathbf{y}_{t+1}$.
  - $\varphi_{123}(\mathbf{y}_t,\mathbf{y}_{t+1})$ = [[$\mathbf{y}_t$="Noun" $\wedge$ $\mathbf{y}_{t+1}$="Verb"]]
- Attribute for each pair of values for input $\mathbf{x}_t$ and output $\mathbf{x}_t$.

  $\bar{\varphi}_{234}(\mathbf{x}_t,\mathbf{y}_t)$ = [[$\mathbf{y}_t$="Noun" $\wedge$ $\mathbf{x}_t$ = "cat"]]

- Label-label: $\sum_t \varphi_i(\mathbf{y}_t,\mathbf{y}_{t+1})$.
- Label-observation $\sum_t \bar{\varphi}_i(\mathbf{x}_t,\mathbf{y}_t)$.

- Joint feature representation $\mathbf{\Phi}(\mathbf{x},\mathbf{y})=\sum_t(\ldots,\varphi_{123}(\mathbf{y}_t,\mathbf{y}_{t+1}),\ldots,\bar{\varphi}_{234}(\mathbf{x}_t,\mathbf{y}_t),\ldots)^T$
- Weight vector $\mathbf{w}=(\ldots,w_{123},\ldots,w_{234},\ldots)^T$

- Use Viterbi decoder to find argmax.

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Structural Learning

- Given:
  - Labeled pairs $(\mathbf{x}_1,\mathbf{y}_1), \ldots, (\mathbf{x}_n,\mathbf{y}_n)$, drawn from $P_{\mathcal{X}\mathcal{Y}}$.

- Learn decision function: $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ with low risk.
- Learn generalized linear model:
  - $f(\mathbf{x}, \mathbf{y}) = \langle \mathbf{w}, \Phi(\mathbf{x}, \mathbf{y}) \rangle$
  - with low regularized empirical risk:
  - $\hat{R}(f) = \sum_{i=1}^{n} \ell(\mathbf{x}_i, \mathbf{y}_i, f) + \|f\|^2$
- Inference, decoding:
  - Compute output $\hat{\mathbf{y}} = \operatorname*{argmax}_{\mathbf{y} \in \mathcal{Y}} f(\mathbf{x}, \mathbf{y})$

Alexander Zien, Ulf Brefeld, Tobias Scheffer

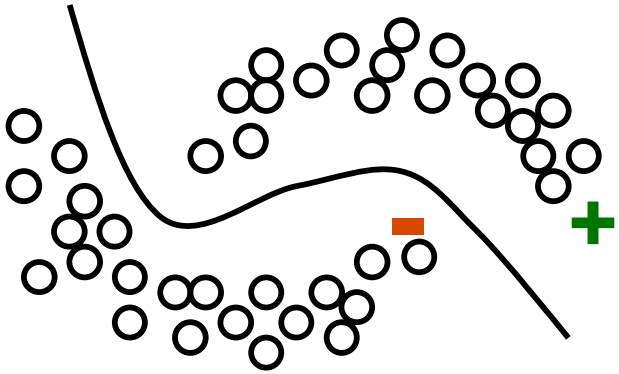
Structural Learning

- Given:
  - Labeled pairs $(\mathbf{x}_1,\mathbf{y}_1), \ldots, (\mathbf{x}_n,\mathbf{y}_n)$, drawn from $P_{\mathcal{X}\mathcal{Y}}$.
  - Unlabeled instances $\mathbf{x}_{n+1}, \ldots, \mathbf{x}_{n+m}$.
- Learn decision function: $f : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ with low risk.
- Learn generalized linear model:
  - $f(\mathbf{x}, \mathbf{y}) = \langle \mathbf{w}, \Phi(\mathbf{x}, \mathbf{y}) \rangle$
  - with low regularized empirical risk:
  - $\hat{R}(f) = \sum_{i=1}^{n} \ell(\mathbf{x}_i, \mathbf{y}_i, f) + \|f\|^2 + \mathrm{r}(\mathbf{x}_{n+1}, \ldots, \mathbf{x}_{n+m})$
  - Add data-dependent regularizer.
  - Dependent on unlabeled pairs, linked to low risk.

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Data-Dependent Regularizer

- Separate unlabeled instances by large margin.
- Decision boundary not to cross high-density regions.



- Add slack terms for unlabeled instances to optimization problem.
- Treat class labels as additional discrete parameters.

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Inductive Optimization Criterion

- Over all $\mathbf{w}$:

$$\min_{\mathbf{w},\ \xi_k} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \xi_i$$

$$s.t. \quad \forall_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} : \mathbf{w}^\top \left[\Phi(\mathbf{x}_i, \mathbf{y}_i) - \Phi(\mathbf{x}_i, \bar{\mathbf{y}}_i)\right] \geq 1 - \xi_i, \quad \xi_i \geq 0$$

- Labeled examples $(\mathbf{x}_i, \mathbf{y}_i)$.

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Inductive Optimization Criterion

- Over all $\mathbf{w}$:

$$\min_{\mathbf{w},\ \xi_k} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \xi_i$$

$$s.t. \quad \forall_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} : \mathbf{w}^\top \left[\Phi(\mathbf{x}_i, \mathbf{y}_i) - \Phi(\mathbf{x}_i, \bar{\mathbf{y}}_i)\right] \geq 1 - \xi_i, \quad \xi_i \geq 0$$

- Iterative working-set algorithm.
  - For each $\mathbf{x}_i$, find highest-scoring offending $\bar{\mathbf{y}} \neq \mathbf{y}_i$.
  - If margin constraint is violated, add to working set.
  - Repeat until no more margin violators.

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Inductive Optimization Criterion

- Over all $\mathbf{w}$:

$$\min_{\mathbf{w},\ \xi_k} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \xi_i$$

$$s.t. \quad \forall_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} : \mathbf{w}^\top \left[\Phi(\mathbf{x}_i, \mathbf{y}_i) - \Phi(\mathbf{x}_i, \bar{\mathbf{y}}_i)\right] \geq 1 - \xi_i, \quad \xi_i \geq 0$$

- Labeled examples $(\mathbf{x}_i, \mathbf{y}_i)$.

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Transductive Optimization Criterion

- Over all $\mathbf{w}$, $\mathbf{y}_j$:

$$\min_{\mathbf{w},\mathbf{y}_j,\xi_k} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \xi_i + C^*\sum_j \xi_j$$

$$s.t. \quad \begin{array}{ll} \forall_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} : \mathbf{w}^\top\left[\Phi(\mathbf{x}_i,\mathbf{y}_i) - \Phi(\mathbf{x}_i,\bar{\mathbf{y}}_i)\right] \geq 1 - \xi_i, & \xi_i \geq 0 \\ \forall_{\bar{\mathbf{y}}_j \neq \mathbf{y}_j} : \mathbf{w}^\top\left[\Phi(\mathbf{x}_j,\mathbf{y}_j) - \Phi(\mathbf{x}_j,\bar{\mathbf{y}}_j)\right] \geq 1 - \xi_j, & \xi_j \geq 0 \end{array}$$

- Labeled examples $(\mathbf{x}_i, \mathbf{y}_i)$.
- Unlabeled examples $\mathbf{x}_j$.

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Transductive Optimization Criterion

- Over all $\mathbf{w}$, $\mathbf{y}_j$:

$$\min_{\mathbf{w},\mathbf{y}_j,\xi_k} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \xi_i + C^*\sum_j \xi_j$$

$$s.t. \quad \begin{array}{ll} \forall_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} : \mathbf{w}^\top\left[\Phi(\mathbf{x}_i,\mathbf{y}_i) - \Phi(\mathbf{x}_i,\bar{\mathbf{y}}_i)\right] \geq 1 - \xi_i, & \xi_i \geq 0 \\ \forall_{\bar{\mathbf{y}}_j \neq \mathbf{y}_j} : \mathbf{w}^\top\left[\Phi(\mathbf{x}_j,\mathbf{y}_j) - \Phi(\mathbf{x}_j,\bar{\mathbf{y}}_j)\right] \geq 1 - \xi_j, & \xi_j \geq 0 \end{array}$$

- Non-convex problem.
- Variables $\mathbf{y}_j$ are discrete.
- Integer program; NP-complete!

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Transductive Optimization Criterion

- Over all $\mathbf{w}$, $\mathbf{y}_j$:

$$\min_{\mathbf{w},\mathbf{y}_j,\xi_k} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \xi_i + C^*\sum_j \xi_j$$

$$s.t. \quad \begin{array}{ll} \forall_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} : \mathbf{w}^\top\left[\Phi(\mathbf{x}_i,\mathbf{y}_i) - \Phi(\mathbf{x}_i,\bar{\mathbf{y}}_i)\right] \geq 1 - \xi_i, & \xi_i \geq 0 \\ \forall_{\bar{\mathbf{y}}_j \neq \mathbf{y}_j} : \mathbf{w}^\top\left[\Phi(\mathbf{x}_j,\mathbf{y}_j) - \Phi(\mathbf{x}_j,\bar{\mathbf{y}}_j)\right] \geq 1 - \xi_j, & \xi_j \geq 0 \end{array}$$

- Non-convex problem.
- Variables $\mathbf{y}_j$ are discrete
- Integer program; NP-complete!

- $\rightarrow$ Unconstrained, differentiable optimization problem.

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Continuous Optimization

Solve for slack terms, closed expression of loss.

$$\min_{\mathbf{w},\mathbf{y}_j,\xi_k} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \xi_i + C^*\sum_j \xi_j$$

$$s.t. \quad \begin{array}{l} \forall_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} : \mathbf{w}^\top\left[\Phi(\mathbf{x}_i,\mathbf{y}_i) - \Phi(\mathbf{x}_i,\bar{\mathbf{y}}_i)\right] \geq 1 - \xi_i, \quad \xi_i \geq 0 \\ \forall_{\bar{\mathbf{y}}_j \neq \mathbf{y}_j} : \mathbf{w}^\top\left[\Phi(\mathbf{x}_j,\mathbf{y}_j) - \Phi(\mathbf{x}_j,\bar{\mathbf{y}}_j)\right] \geq 1 - \xi_j, \quad \xi_j \geq 0 \end{array}$$

$$\Leftrightarrow$$

$$\xi_i = \max_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} \max\left\{1 - \mathbf{w}^\top\left[\Phi(\mathbf{x}_i,\mathbf{y}_i) - \Phi(\mathbf{x}_i,\bar{\mathbf{y}}_i)\right], 0\right\}$$

$$\xi_j = \min_{\mathbf{y}_j} \max_{\bar{\mathbf{y}}_j \neq \mathbf{y}_j} \max\left\{1 - \mathbf{w}^\top\left[\Phi(\mathbf{x}_j,\mathbf{y}_j) - \Phi(\mathbf{x}_j,\bar{\mathbf{y}}_j)\right], 0\right\}$$

Continuous Optimization

- Solve for slack terms, closed expression of loss.

$$\min_{\mathbf{w},\mathbf{y}_j,\xi_k} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \xi_i + C^*\sum_j \xi_j$$

$$s.t. \quad \begin{aligned} &\forall_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} : \mathbf{w}^\top\left[\Phi(\mathbf{x}_i,\mathbf{y}_i) - \Phi(\mathbf{x}_i,\bar{\mathbf{y}}_i)\right] \geq 1 - \xi_i, \quad \xi_i \geq 0 \\ &\forall_{\bar{\mathbf{y}}_j \neq \mathbf{y}_j} : \mathbf{w}^\top\left[\Phi(\mathbf{x}_j,\mathbf{y}_j) - \Phi(\mathbf{x}_j,\bar{\mathbf{y}}_j)\right] \geq 1 - \xi_j, \quad \xi_j \geq 0 \end{aligned}$$

$$\Leftrightarrow$$

$$\xi_i = \max_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} \max\left\{1 - \mathbf{w}^\top\left[\Phi(\mathbf{x}_i,\mathbf{y}_i) - \Phi(\mathbf{x}_i,\bar{\mathbf{y}}_i)\right], 0\right\}$$

$$\xi_j = \min_{\mathbf{y}_j} \max_{\bar{\mathbf{y}}_j \neq \mathbf{y}_j} \max\left\{1 - \mathbf{w}^\top\left[\Phi(\mathbf{x}_j,\mathbf{y}_j) - \Phi(\mathbf{x}_j,\bar{\mathbf{y}}_j)\right], 0\right\}$$

Now written as closed, unconstrained optimization problem.

Alexander Zien, Ulf Brefeld, Tobias Scheffer

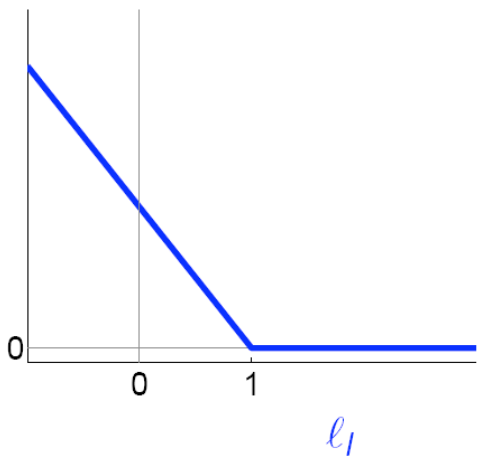
Continuous Optimization

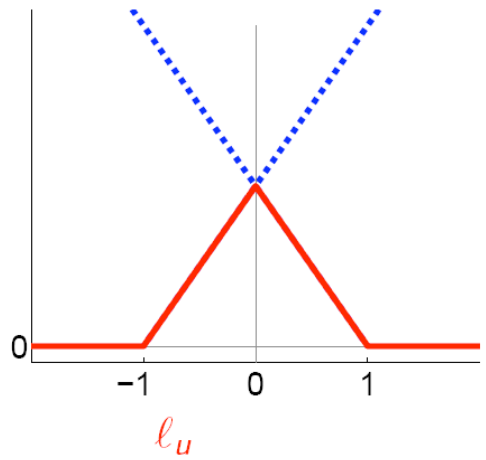
- Make differentiable.

$$\min_{\mathbf{w},\mathbf{y}_j,\xi_k} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \xi_i + C^*\sum_j \xi_j$$

$$s.t. \quad \xi_i = \max_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} \max\left\{1 - \mathbf{w}^\top\left[\Phi(\mathbf{x}_i,\mathbf{y}_i) - \Phi(\mathbf{x}_i,\bar{\mathbf{y}}_i)\right], 0\right\}$$

$$\xi_j = \min_{\mathbf{y}_j} \max_{\bar{\mathbf{y}}_j \neq \mathbf{y}_j} \max\left\{1 - \mathbf{w}^\top\left[\Phi(\mathbf{x}_j,\mathbf{y}_j) - \Phi(\mathbf{x}_j,\bar{\mathbf{y}}_j)\right], 0\right\}$$





Alexander Zien, Ulf Brefeld, Tobias Scheffer

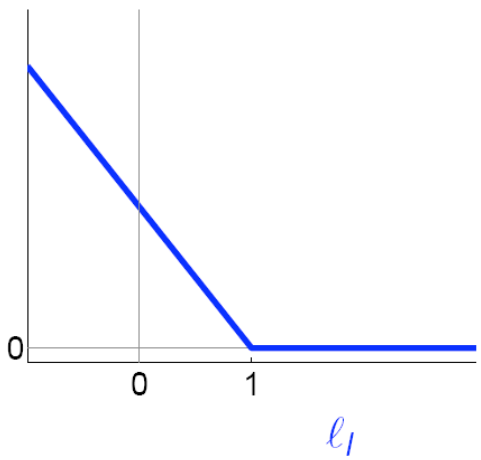
Continuous Optimization

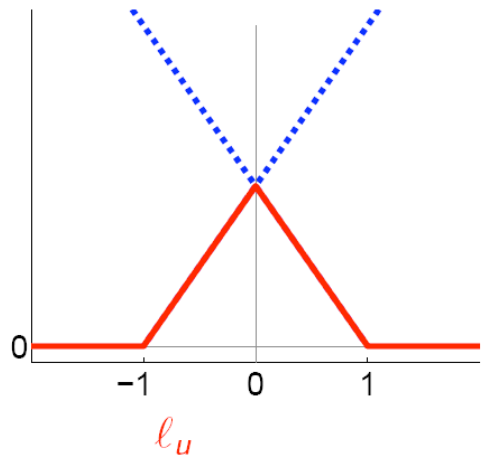
- Make differentiable.

$$\min_{\mathbf{w},\mathbf{y}_j,\xi_k} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \xi_i + C^*\sum_j \xi_j$$

$$s.t. \quad \xi_i = \max_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} \ell_l\left(\mathbf{w}^\top\Phi(\mathbf{x}_i,\mathbf{y}_i) - \mathbf{w}^\top\Phi(\mathbf{x}_i,\bar{\mathbf{y}}_i)\right)$$

$$\xi_j = \min_{\mathbf{y}_j}\max_{\bar{\mathbf{y}}_j \neq \mathbf{y}_j} \ell_u\left(\mathbf{w}^\top\Phi(\mathbf{x}_j,\mathbf{y}_j) - \mathbf{w}^\top\Phi(\mathbf{x}_j,\bar{\mathbf{y}}_j)\right)$$





Alexander Zien, Ulf Brefeld, Tobias Scheffer

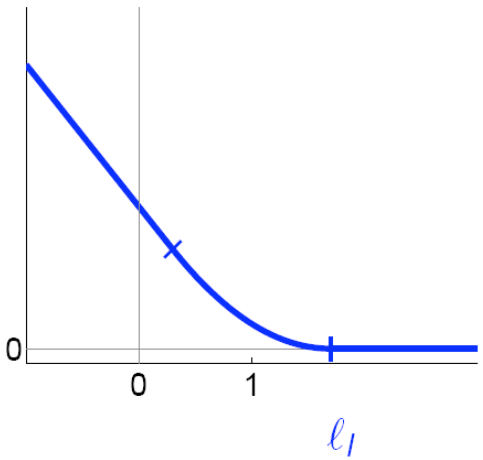
Continuous Optimization

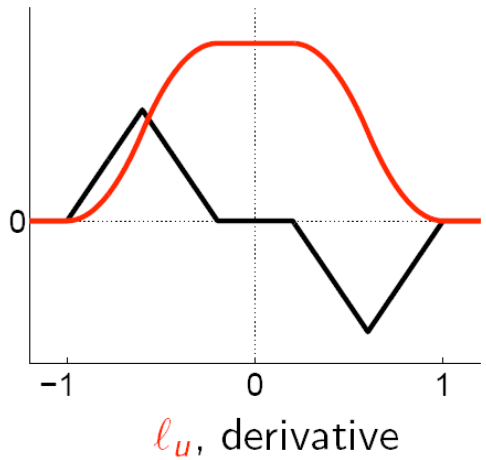
- Make differentiable.

$$\min_{\mathbf{w},\mathbf{y}_j,\xi_k} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \xi_i + C^*\sum_j \xi_j$$

$$s.t. \quad \xi_i = \max_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} \ell_l\left(\mathbf{w}^\top\Phi(\mathbf{x}_i,\mathbf{y}_i) - \mathbf{w}^\top\Phi(\mathbf{x}_i,\bar{\mathbf{y}}_i)\right)$$

$$\xi_j = \min_{\mathbf{y}_j}\max_{\bar{\mathbf{y}}_j \neq \mathbf{y}_j} \ell_u\left(\mathbf{w}^\top\Phi(\mathbf{x}_j,\mathbf{y}_j) - \mathbf{w}^\top\Phi(\mathbf{x}_j,\bar{\mathbf{y}}_j)\right)$$





Alexander Zien, Ulf Brefeld, Tobias Scheffer

Continuous Optimization

- Make differentiable.

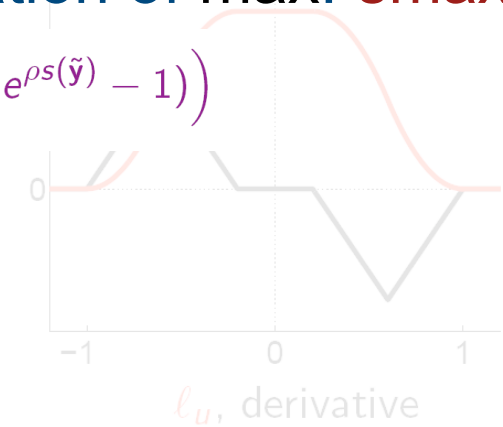
$$\min_{\mathbf{w},\mathbf{y}_j,\xi_k} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \xi_i + C^*\sum_j \xi_j$$

$$s.t. \quad \xi_i = \operatorname*{smax}_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} \ell_l\left(\mathbf{w}^\top\Phi(\mathbf{x}_i,\mathbf{y}_i) - \mathbf{w}^\top\Phi(\mathbf{x}_i,\bar{\mathbf{y}}_i)\right)$$

$$\xi_j = \min_{\mathbf{y}_j} \operatorname*{smax}_{\bar{\mathbf{y}}_j \neq \mathbf{y}_j} \ell_u\left(\mathbf{w}^\top\Phi(\mathbf{x}_j,\mathbf{y}_j) - \mathbf{w}^\top\Phi(\mathbf{x}_j,\bar{\mathbf{y}}_j)\right)$$

- Differentiable approximation of max: smax.

$$\operatorname*{smax}_{\tilde{\mathbf{y}} \neq \mathbf{y}_k}(s(\tilde{\mathbf{y}})) = \frac{1}{\rho}\log\left(1 + \sum\nolimits_{\tilde{\mathbf{y}} \neq \mathbf{y}_k}\left(e^{\rho s(\tilde{\mathbf{y}})} - 1\right)\right)$$



Alexander Zien, Ulf Brefeld, Tobias Scheffer

Continuous Optimization

- Unconstraint, differentiable optimization criterion.

$$
\begin{aligned}
\min_{\mathbf{w},\xi_k} \quad & \frac{1}{2}\mathbf{w}^\top\mathbf{w} \\
& + C \sum_i \operatorname*{smax}_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} \ell_l \left(\mathbf{w}^\top \Phi(\mathbf{x}_i, \mathbf{y}_i) - \mathbf{w}^\top \Phi(\mathbf{x}_i, \bar{\mathbf{y}}_i) \right) \\
& + C^* \sum_j \min_{\mathbf{y}_j} \operatorname*{smax}_{\bar{\mathbf{y}}_j \neq \mathbf{y}_j} \ell_u \left(\mathbf{w}^\top \Phi(\mathbf{x}_j, \mathbf{y}_j) - \mathbf{w}^\top \Phi(\mathbf{x}_j, \bar{\mathbf{y}}_j) \right)
\end{aligned}
$$

- Iterative working-set algorithm for labeled and unlabeled data, until no more margin violators.

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Continuous Optimization

- Unconstraint, differentiable optimization criterion.

$$\min_{\mathbf{w},\xi_k} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \operatorname*{smax}_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} \ell_l\left(\mathbf{w}^\top\Phi(\mathbf{x}_i,\mathbf{y}_i) - \mathbf{w}^\top\Phi(\mathbf{x}_i,\bar{\mathbf{y}}_i)\right) + C^*\sum_j \min_{\mathbf{y}_j} \operatorname*{smax}_{\bar{\mathbf{y}}_j \neq \mathbf{y}_j} \ell_u\left(\mathbf{w}^\top\Phi(\mathbf{x}_j,\mathbf{y}_j) - \mathbf{w}^\top\Phi(\mathbf{x}_j,\bar{\mathbf{y}}_j)\right)$$

- Gradient descent over primal parameters:
  - Follow $\frac{\partial obj}{\partial \mathbf{w}}$.

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Continuous Optimization

- Unconstraint, differentiable optimization criterion.

$$\begin{aligned}\min_{\mathbf{w},\xi_k} \quad & \frac{1}{2}\mathbf{w}^\top\mathbf{w} \\ & + C\sum_i \operatorname*{smax}_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} \ell_l\left(\mathbf{w}^\top\Phi(\mathbf{x}_i,\mathbf{y}_i) - \mathbf{w}^\top\Phi(\mathbf{x}_i,\bar{\mathbf{y}}_i)\right) \\ & + C^*\sum_j \min_{\mathbf{y}_j} \operatorname*{smax}_{\bar{\mathbf{y}}_j \neq \mathbf{y}_j} \ell_u\left(\mathbf{w}^\top\Phi(\mathbf{x}_j,\mathbf{y}_j) - \mathbf{w}^\top\Phi(\mathbf{x}_j,\bar{\mathbf{y}}_j)\right)\end{aligned}$$

- Gradient descent over parameters.
- Invoke Representer Theorem:

$$\mathbf{w} = \sum_{k=1}^{n+m}\sum_{\mathbf{y}\in\mathcal{Y}(\mathbf{x}_k)} \alpha_{k,\mathbf{y}}\Phi(\mathbf{x}_k,\mathbf{y})$$

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Continuous Optimization

- Unconstraint, differentiable optimization criterion.

$$\min_{\mathbf{w},\xi_k} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \operatorname*{smax}_{\bar{\mathbf{y}}_i \neq \mathbf{y}_i} \ell_l\left(\mathbf{w}^\top\Phi(\mathbf{x}_i,\mathbf{y}_i) - \mathbf{w}^\top\Phi(\mathbf{x}_i,\bar{\mathbf{y}}_i)\right) + C^*\sum_j \min_{\mathbf{y}_j} \operatorname*{smax}_{\bar{\mathbf{y}}_j \neq \mathbf{y}_j} \ell_u\left(\mathbf{w}^\top\Phi(\mathbf{x}_j,\mathbf{y}_j) - \mathbf{w}^\top\Phi(\mathbf{x}_j,\bar{\mathbf{y}}_j)\right)$$

- Chain rule gives dual gradient.
  - Calculate $\dfrac{\partial obj}{\partial \alpha_{k,\mathbf{y}}} = \dfrac{\partial obj}{\partial \mathbf{w}} \cdot \dfrac{\partial \mathbf{w}}{\partial \alpha_{k,\mathbf{y}}}$
  - With $\mathbf{w}^\top\Phi(\mathbf{x}_i,\mathbf{y}_i) = \sum_k \sum_{\mathbf{y}} \alpha_{k,\mathbf{y}} \underbrace{\Phi(\mathbf{x}_k,\mathbf{y})^\top\Phi(\mathbf{x}_i,\mathbf{y}_i)}_{k((\mathbf{x}_k,\mathbf{y}),(\mathbf{x}_i,\mathbf{y}_i))}$

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Continuous Optimization

- Continuous TSVM for structured variables.

Input: labeled points $\{(\mathbf{x}_i, \mathbf{y}_i)\}$, unlabeled points $\{\mathbf{x}_j\}$.
Output: working set $\mathcal{W}$ and associated $\alpha_{k,\mathbf{y}}$.

Initialize $\mathcal{W} \leftarrow \{(\mathbf{x}_i, \mathbf{y}_i)\}$.

Alternate until convergence:

1. Augment working set $\mathcal{W}$
   - add $\{(\mathbf{x}_i, \bar{\mathbf{y}}_i^*)\}$ to $\mathcal{W}$ (worst margin violators)
   - find $\{\mathbf{y}_j{}^*\}$ (highest scoring labels)
   - add $\{(\mathbf{x}_j, \bar{\mathbf{y}}_j^*)\}$ to $\mathcal{W}$ (2nd highest scoring labels)
2. Optimize α by preconditioned Conjugate Gradient.

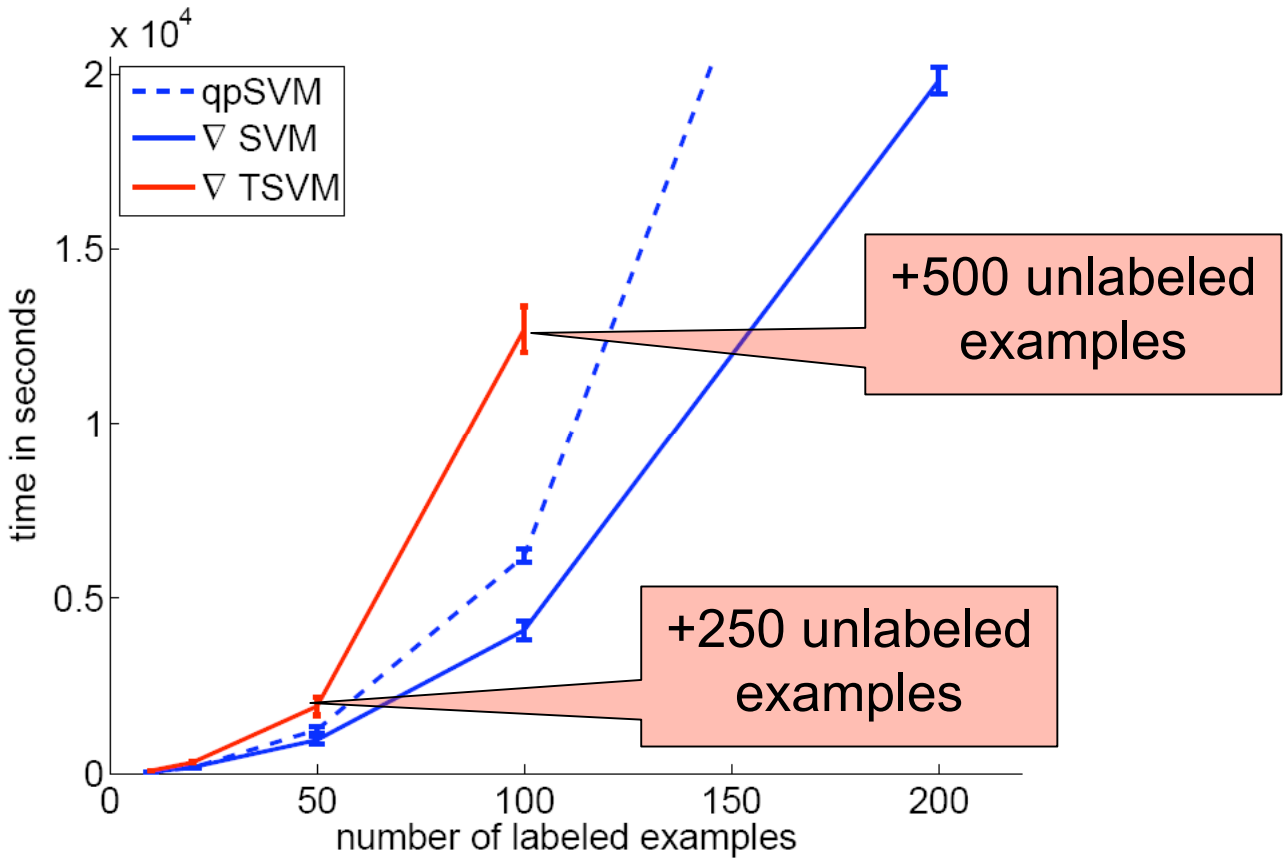
Alexander Zien, Ulf Brefeld, Tobias Scheffer

Experiments

- Execution time QP versus continuous optimization.
- Accuracy supervised vs. semi-supervised learning.
  - Multi-class classification: text classification.
  - Label-sequence learning: NER.
- Comparison / combination with Laplacian kernel.

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Execution Time

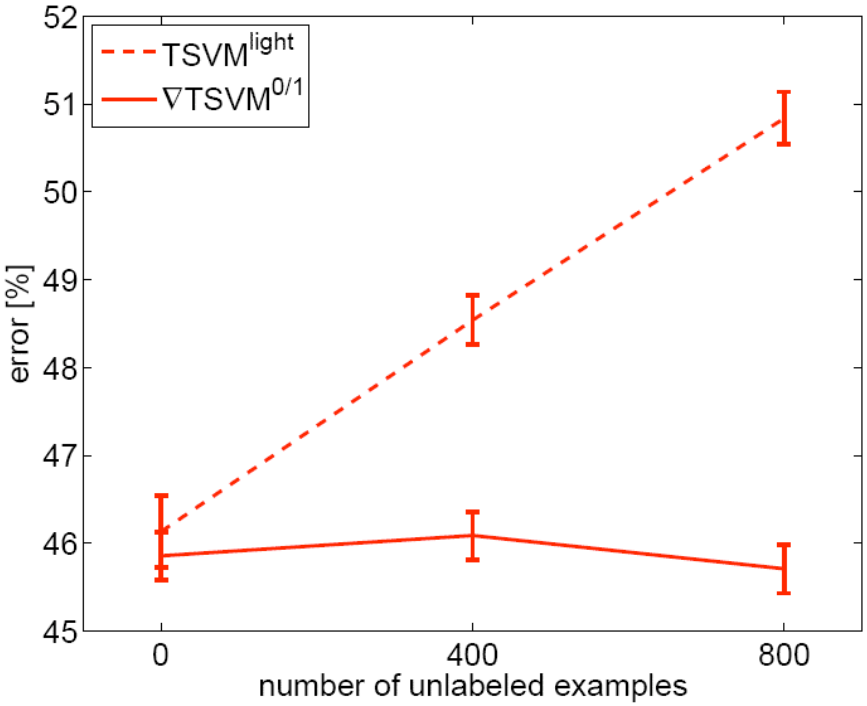


- Continuous optimization many times faster!

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Accuracy: Cora Text Classification

- Multiclass, 8 classes, 200 unlabeled examples.

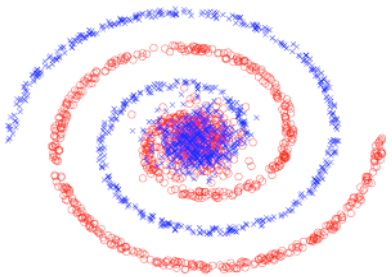


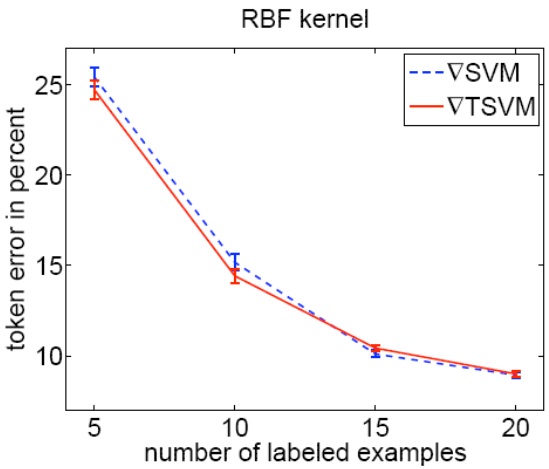
- Combinatorial optimization: much higher error!
- Continuous optimization: higher or same error!

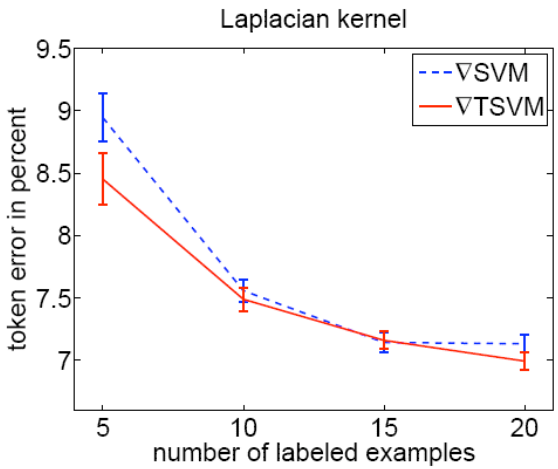
Alexander Zien, Ulf Brefeld, Tobias Scheffer

Accuracy: Galaxies Problem

- Artificial data set [Lafferty et al., IMCL 2004].
- Laplacian kernel: huge benefit.
- TSVM: marginal benefit.
- Laplacian+TSVM: slight additional benefit.



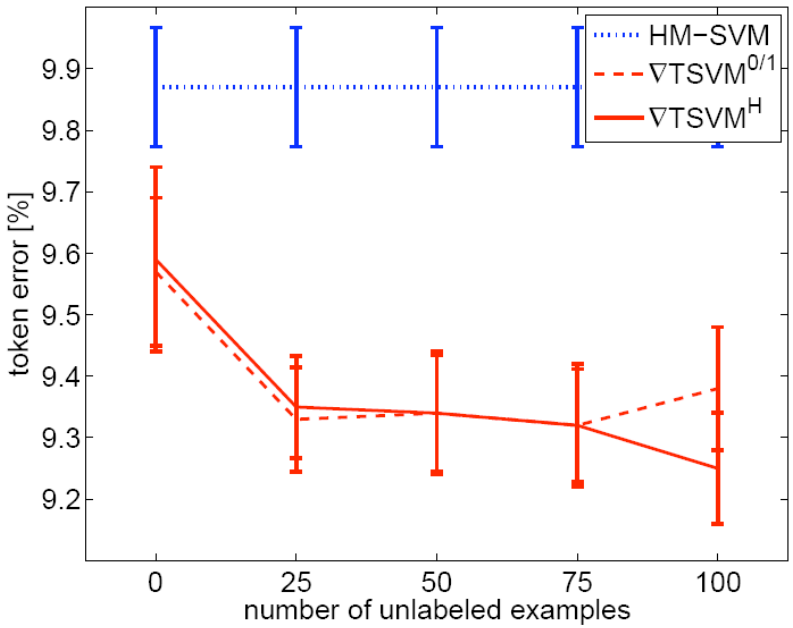




Alexander Zien, Ulf Brefeld, Tobias Scheffer

Accuracy: Spanish News Wire NER

- Sequence learning, 9 labels.



- Gradient TSVM clearly outperforms SVM.
- Adding unlabeled data further decreases error.

Alexander Zien, Ulf Brefeld, Tobias Scheffer

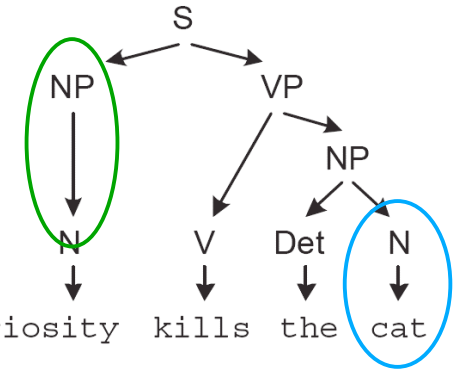
Conclusions

- TSVM for structured outputs.
  - Use information from unlabeled (test) examples.
  - Unconstrained, differentiable optimization criterion.
  - Efficient conjugate gradient descent algorithm.
- Empirically:
  - Sometines no improvement.
  - Sometimes, unlabeled data increase accuracy significantly.
- Because:
  - SVM criterion is convex;
  - TSVM criterion has many local minima.

Alexander Zien, Ulf Brefeld, Tobias Scheffer

Natural Language Parsing

y =



x = Curiosity kills the cat

$$\varphi_{NP\to N}(\mathbf{y}) = w_{NP\to N} \sum_{p\in \mathbf{y}} [[\, \mathbf{y}_p = NP\to N \,]]$$

$$\varphi_{N\to\text{"cat"}}(\mathbf{x},\mathbf{y}) = w_{N\to\text{"cat"}} \sum_{p\in \mathbf{y}} [[\, \mathbf{y}_p = N\to\text{"cat"} \,]]$$

$$\boldsymbol{\varphi}(\mathbf{x},\mathbf{y}) = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \begin{matrix} S \to NP,VP \\ NP \to N \\ VP \to V \\ VP \to V,NP \\ \\ N \to ate \\ N \to cat \end{matrix}$$

- Joint feature representation: $\boldsymbol{\Phi}(\mathbf{x},\mathbf{y}) = (\ldots,\varphi_{NP\to N}(y_p),\ldots,\varphi_{N\to\text{"cat"}}(\mathbf{x},y_p),\ldots)^T$
- Weight vector $\mathbf{w} = (\ldots,w_{NP\to N},\ldots,w_{N\to\text{"cat"}},\ldots)^T$.
- Use chart parser as decoder to find argmax.

Alexander Zien, Ulf Brefeld, Tobias Scheffer