An Introduction to Machine Learning

L6: Structured Estimation

Alexander J. Smola

Statistical Machine Learning Program
Canberra, ACT 0200 Australia
Alex.Smola@nicta.com.au

Tata Institute, Pune, January 2007

Overview

L1: Machine learning and probability theory
Introduction to pattern recognition, classification, regression, novelty detection, probability theory, Bayes rule, inference

L2: Density estimation and Parzen windows
Nearest Neighbor, Kernels density estimation, Silverman's rule, Watson Nadaraya estimator, crossvalidation

L3: Perceptron and Kernels
Hebb's rule, perceptron algorithm, convergence, kernels

L4: Support Vector estimation
Geometrical view, dual problem, convex optimization, kernels

L5: Support Vector estimation
Regression, Novelty detection

L6: Structured Estimation
Sequence annotation, web page ranking, path planning, implementation and optimization

L6 Structured Estimation

Multiclass Estimation

- Margin Definition
- Optimization Problem
- Dual Problem

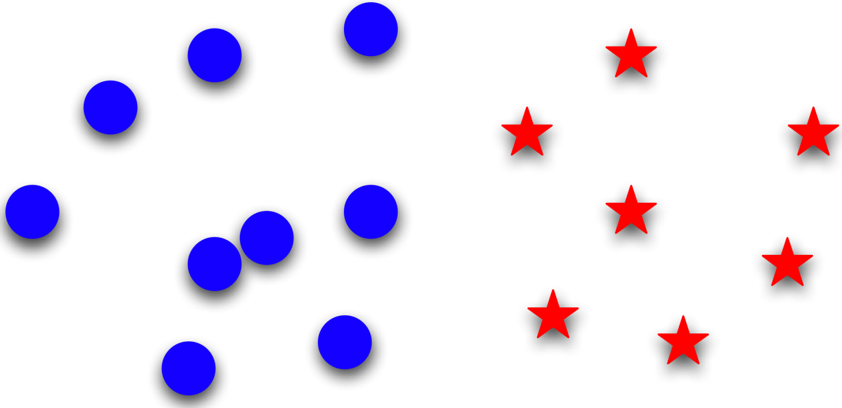
Max-Margin-Markov Networks

- Feature map
- Column generation and SVMStruct
- Application to sequence annotation

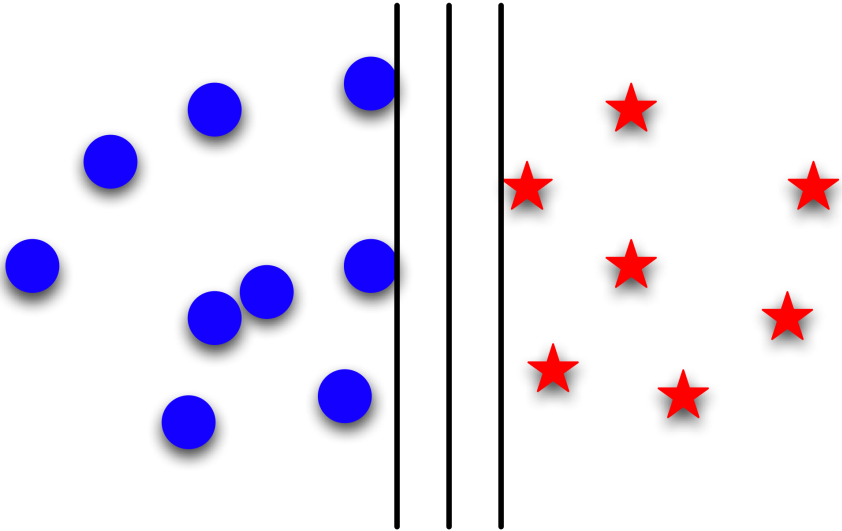
Web Page Ranking

- Ranking Measures
- Linear assignment problems
- Examples

Binary Classification



Binary Classification



Multiclass Classification

Goal

Given x_i and $y_i \in \{1, \ldots, N\}$, define a margin.

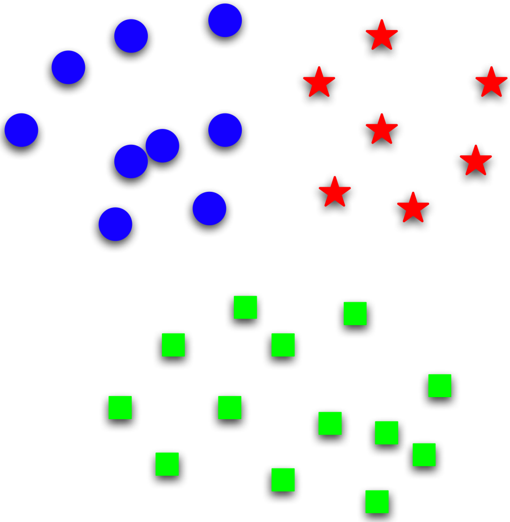
Binary Classification

$$
\begin{aligned}
\text{for } y_i = 1 \ \ \langle x_i, w \rangle &\geq 1 + \langle x_i, -w \rangle \\
\text{for } y_i = -1 \ \ \langle x_i, -w \rangle &\geq 1 + \langle x_i, w \rangle
\end{aligned}
$$

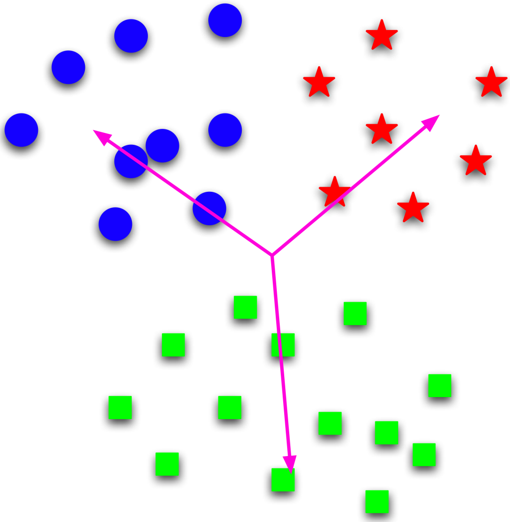
Multiclass Classification

$$\langle x_i, w_y \rangle \geq 1 + \langle x_i, w_{y'} \rangle \ \text{ for all } y' \neq y.$$

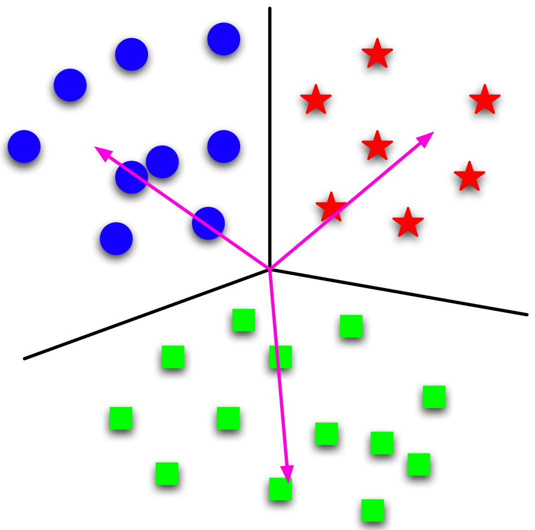
Multiclass Classification



Multiclass Classification



Multiclass Classification



Structured Estimation

Key Idea

Combine x and y into **one** feature vector $\phi(x, y)$.

Large Margin Condition and Slack

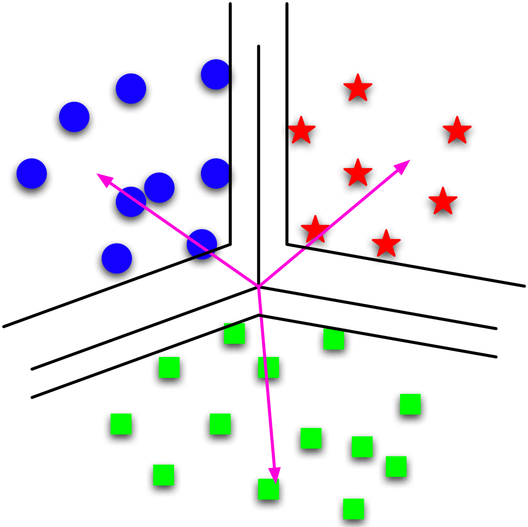
$$\langle \Phi(x, y), w \rangle \geq \Delta(y, y') + \langle \Phi(x, y'), w \rangle - \xi \text{ for all } y' \neq y.$$

- $\Delta(y, y')$ is the cost of misclassifying y for y'.
- $\xi \geq 0$ is as a slack variable.

$$\underset{w,\xi}{\text{minimize}} \ \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{m} \xi_i$$

subject to $\langle \Phi(x_i, y_i) - \Phi(x_i, y'), w \rangle \geq \Delta(y_i, y') - \xi_i$ for all $y' \neq y_i$.

Multiclass Margin



Dual Problem

Quadratic Program

$$\underset{\alpha}{\text{minimize}} \ \frac{1}{2} \sum_{i,j,y,y'} \alpha_{iy} \alpha_{jy'} K_{iy,jy'} - \sum_{i,y} \alpha_{iy} \Delta(y_i, y)$$

$$\text{subject to} \ \sum_{y} \alpha_{iy} \leq C \text{ and } \alpha_{iy} \geq 0.$$

Here $K_{iy,jy'} = \langle \phi(x_i, y_i) - \phi(x_i, y), \phi(x_j, y_j) - \phi(x_j, y') \rangle$.

$$w = \sum_{i,y} \alpha_{iy} \left(\phi(x_i, y_i) - \phi(x_i, y) \right).$$

Solving It

- Use SVMStruct (by Thorsten Joachims)
- Column generation (subset optimization). At optimality:

$$\alpha_{iy} \left[\langle \phi(x_i, y_i) - \phi(x_i, y), w \rangle - \Delta(y_i, y) \right] = 0$$

Pick (i, y) pairs for which this doesn't hold.

Implementing It

Start

Use an existing structured SVM solver, e.g. SVMStruct.

Loss Function

Define a loss function $\Delta(y, y')$ for your problem.

Feature Map

Define a suitable feature map $\phi(x, y)$. More examples later.

Column Generator

Implement algorithm which maximizes

$$\langle \phi(x_i, y), w \rangle + \Delta(y_i, y)$$

Mini Summary

Multiclass Margin

- Joint Feature Map
- Relative margin using misclassification error
- Binary classification a special case

Optimization Problem

- Convex Problem
- Can be solved using existing packages
- Column generation
- Joint feature map

Named Entity Tagging

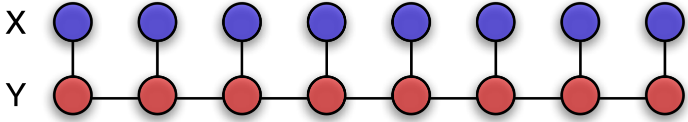
Goal

Given a document, i.e. a sequence of words, find those words which correspond to named entities.

Interaction

Adjacent labels will influence which words get tagged.

`President Bush was hiding behind the bush.`



Joint Feature Map

$$\phi(x, y) = \left[\sum_{i=1}^{l} y_i \phi(x_i), \sum_{i=1}^{l} y_i y_{i+1} \right]$$

Estimation and Column Generation

Loss Function

Count how many of the labels are wrong, i.e. $\Delta(y, y') = \|y - y'\|_1$.

Estimation

Find sequence y maximizing $\langle \phi(x, y), w \rangle$, that is

$$\sum_{i=1}^{l} y_i \langle \phi(x_i), w_1 \rangle + y_i y_{i+1} w_2$$

For column generation additional term $\sum_{i=1}^{l} |y_i - y_i'|$.

Dynamic Programming

We are maximizing a function $\sum_{i=1}^{l} f(y_i, y_{i+1})$.

Dynamic Programming

Background

Generalized distributive law, Viterbi, Shortest path

Key Insight

To maximize $\sum_{i=1}^{l} f(y_i, y_{i+1})$, once we've picked $y_j = 1$ the problems on either side become independent. In equations

$$\begin{aligned}
&\operatorname*{maximize}_{y} \sum_{i=1}^{l} f(y_i, y_{i+1}) \\
=&\operatorname*{maximize}_{y_2,\ldots,y_l} \left[\sum_{i=2}^{l} f(y_i, y_{i+1}) + \underbrace{\operatorname*{maximize}_{y_1} f(y_1, y_2)}_{:=g_2(y_2)}\right] \\
=&\operatorname*{maximize}_{y_3,\ldots,y_l} \left[\sum_{i=3}^{l} f(y_i, y_{i+1}) + \underbrace{\operatorname*{maximize}_{y_2} f(y_2, y_3) + g_2(y_2)}_{:=g_3(y_3)}\right]
\end{aligned}$$

Implementing It

Forward Pass

- Compute recursion

$$g_{i+1}(y_{i+1}) := \underset{y_i}{\text{maximize}}\, f(y_i, y_{i+1}) + g_i(y_i)$$

- Store best answers

$$y_i(y_{i+1}) := \underset{y_i}{\text{argmax}}\, f(y_i, y_{i+1}) + g_i(y_i)$$

Backward Pass

After computing the last term y_l, solve recursion $y_i(y_{i+1})$.

Cost

- Linear time for forward and backward pass
- Linear storage

Extensions

Fancy Feature Maps
Can use more complicated interactions between words and labels.

Fancy Labels
More sophisticated than binary labels. E.g. tag for place, person, organization, etc.

Fancy Structures
Rather than linear structure, have a 2D structure. Annotate images.

Mini Summary

Named Entity Tagging

- Sequence of words, find named entities
- Can be written as a structured estimation problem
- Feature map decomposes into separate terms

Dynamic Programming

- Objective function a sum of adjacent terms
- Same as Viterbi algorithm
- Linear time and space

Web Page Ranking

Goal

Given a set of documents d_i and a query q, find ranking of documents such that most relevant documents come first.

Data

At training time, we have ratings of pages $y_i \in \{0, 5\}$.

Scoring Function

Discounted cumulative gain. That is, we gain more if we rank relevant pages highly, namely

$$\mathrm{DCG}(\pi, y) = \sum_{i,j} \pi_{ij} \frac{2^{y_i} + 1}{\log(j+1)}.$$

π is a permutation matrix (exactly one entry per row / column is 1, rest is 0).

From Scores to Losses

Goal

We need a loss function, not a performance score.

Idea

Use performance relative to the best as loss score.

Practical Implementation

Instead of $\mathrm{DCG}(\pi, y)$ use $\Delta(\mathbf{1}, \pi) = \mathrm{DCG}(1, y) - \mathrm{DCG}(\pi, y)$.

Feature map ...

Goal

Find w such that $\langle w, \phi(d_i, q)\rangle$ gives us a score (like PageRank, but we want to learn it from data).

Joint feature map

- Need to map $q, \{d_1, \ldots, d_l\}$ and π into feature space.
- Want to get sort operation at test time from $\langle \phi(q, D, \pi), w\rangle$.

Solution

$$\phi(q, D, \pi) = \sum_{i,j} \pi_{ij} c_i \phi(q, d_j) \text{ where } c_i \text{ is decreasing.}$$

Consequence

$\sum_{i,j} \pi_{ij} c_i \langle \phi(q, d_j), w\rangle$ is maximized by sorting documents along c_i, i.e. in descending order.

Sorting

Unsorted: score is 57

c_i	1	2	3	4	5
Page ranks	3	2	3	9	1

Sorted: score is 71

c_i	1	2	3	4	5
Page ranks	1	2	3	3	9

This is also known as the Polya-Littlewood-Hardy inequality

Column Generation

Goal

Efficiently find permutation which maximizes

$$\langle \phi(q, D, \pi), w \rangle + \Delta(\mathbf{1}, \pi)$$

Optimization Problem

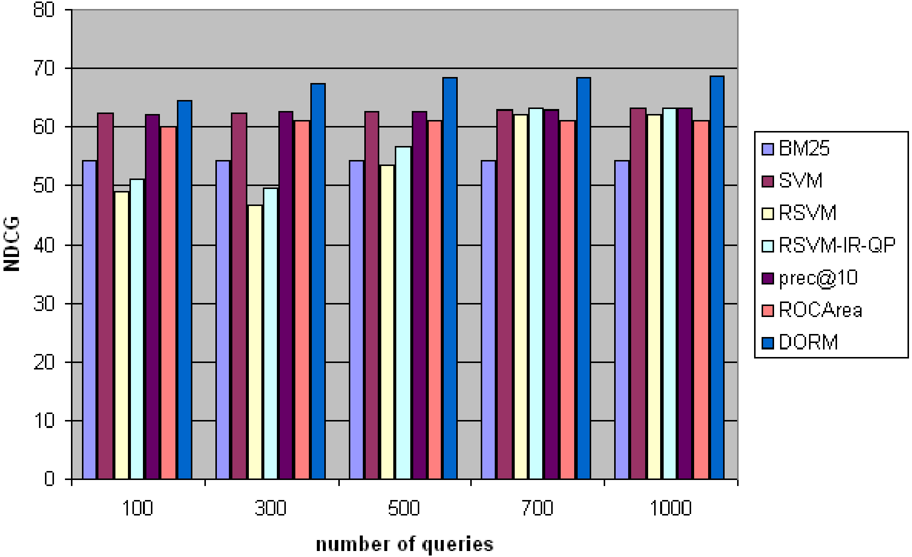
$$\underset{\pi}{\text{maximize}} \sum_{i,j} \pi_{ij} \left[c_i \langle \phi(d_j, q), w \rangle + \frac{2^{y_i} + 1}{\log(j+1)} \right]$$

This is a linear assignment problem. Efficient codes exist (Hungarian marriage algorithm) to solve this in $O(l^3)$ time.

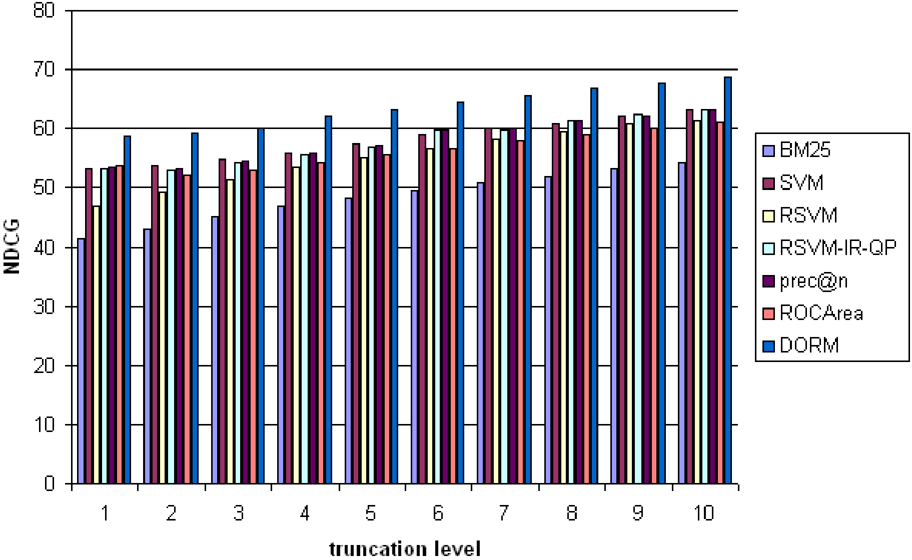
Putting everything together

- Use existing SVM solver (e.g. SVMStruct)
- Implement column generator for training
- Design sorting kernel

NDCG Optimization



NDCG Optimization



Mini Summary

Ranking Problem

- Web page ranking (documents with relevance score)
- Multivariate performance score
- Hard to optimize directly

Feature Map

- Maps permutations and data jointly into feature space
- Simple sort operation at test time

Column Generation

- Linear assignment problem
- Integrate in structured SVM solver

Summary

Structured Estimation

- Basic idea
- Optimization problem

Named Entity Tagging

- Annotation of a sequence
- Joint featuremap
- Dynamic programming

Ranking

- Multivariate performance score
- Linear assignment problem