Max-Margin Methods for NLP:
Estimation, Structure, and Applications

Dan Klein and Ben Taskar
EECS Department
UC Berkeley

Introduction

- Much of NLP can be seen as making decisions
  - About structured analyses (sequences, trees, graphs)
  - On the basis of multiple information sources, or *features* (words, word classes, tree configurations, etc.)
- Widespread adoption of discriminative methods
  - Use of arbitrary features
  - Various formulations: maxent, SVM, perceptron
  - Common use: local discriminative decisions, possibly chained
  - Relatively new: global methods which exploit model structure (CRFs, max-margin networks)
- This tutorial will cover:
  - Part I: Flat max-margin methods (SVMs)
  - Part II: Structured max-margin methods (sequences, trees, matchings)

Outline

- Part I: Flat Classification
  - Linear classifiers and loss functions
  - Primal and dual SVM formulations
  - Training SVMs
- Part II: Structured Classification
  - Structured linear classifiers
  - Factored learning formulations
  - Experimental results

Example: Text Classification

- We want to classify documents into categories

DOCUMENT	CATEGORY
… win the election …	*POLITICS*
… win the game …	*SPORTS*
… see a movie …	*OTHER*

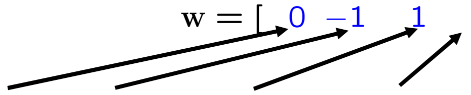
- Classically, do this on the basis of words in the document, but other information sources are potentially relevant:
  - Document length
  - Average word length
  - Document's source
  - Burstiness of new words in document

Some Definitions

INPUTS	$\mathbf{x}^i$	*… win the election …*
TRUE OUTPUTS	$\mathbf{y}^i$	*POLITICS*
OUTPUT SPACE	$\mathcal{Y}$	*SPORTS*, *POLITICS*, *OTHER*
ANY OUTPUTS	$\mathbf{y}$	*SPORTS*, *POLITICS*, *OTHER*

Binary Linear Models

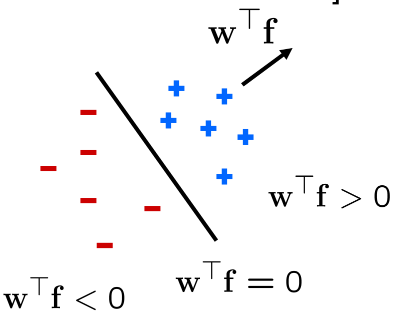
- Two Classes *POLITICS* = +, *SPORTS* = -
- Features $\mathbf{f}(\text{...win the election...}) = [\ 1\quad 0\quad 1\quad 0]$
- Weights $\mathbf{w} = [\ 0\quad -1\quad 1\quad 0]$



[*"win"* *"game"* *"election"* *"movie"*]

- Prediction rule

$$\text{prediction}(\mathbf{x}, \mathbf{w}) = \begin{cases} + & \text{if } \mathbf{w}^\top \mathbf{f}(\mathbf{x}) \geq 0 \\ - & \text{if } \mathbf{w}^\top \mathbf{f}(\mathbf{x}) < 0 \end{cases}$$

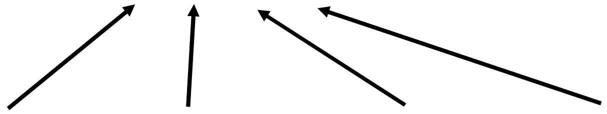


$\mathbf{w}^\top\mathbf{f}$
$\mathbf{w}^\top\mathbf{f} > 0$
$\mathbf{w}^\top\mathbf{f} = 0$
$\mathbf{w}^\top\mathbf{f} < 0$

Multiclass Linear Models

- Multiple Classes *SPORTS*, *POLITICS*, *OTHER*

$$\begin{array}{rl}
\mathbf{f}_i(\mathit{POLITICS}) = & [\ 0\ \ 0\ \ 0\ \ 0\ \ 1\ \ 0\ \ 1\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0] \\
\mathbf{f}_i(\mathit{SPORTS}) = & [\ 1\ \ 0\ \ 1\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0] \\
\mathbf{f}_i(\mathit{OTHER}) = & [\ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 1\ \ 0\ \ 1\ \ 0] \\
\mathbf{w} = & [\ 1\ \ 1\ {-1}\ {-2}\ \ 1\ {-1}\ \ 1\ {-2}\ {-2}\ {-1}\ {-1}\ \ 1]
\end{array}$$



[*"win"*∧SPORTS *"game"*∧SPORTS *"election"*∧SPORTS *"movie"*∧SPORTS]

$$\begin{array}{rcl}
\mathbf{f}(\mathbf{x},\mathbf{y}) & = & [\ 0\ \ 0\ \ \cdots\ \ \mathbf{f}(\mathbf{x})\ \ \cdots\ \ 0\ ] \\
\mathbf{w} & = & [\ \mathbf{w}_0\ \ \mathbf{w}_1\ \ \cdots\ \ \mathbf{w}_\mathbf{y}\ \ \cdots\ \ \mathbf{w}_k\ ]
\end{array}$$

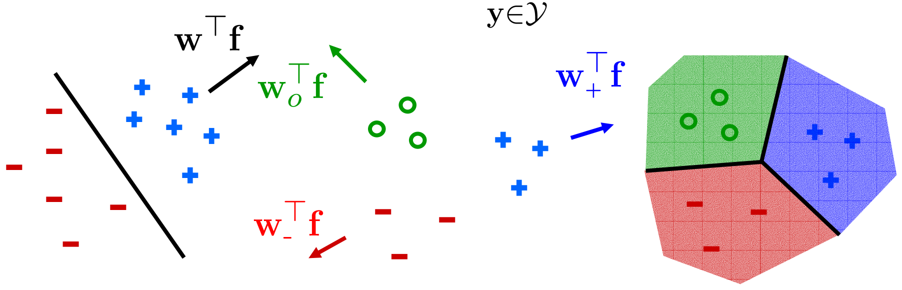
Multiclass Linear Models

$$\begin{array}{rcl}
\mathbf{f}(\mathbf{x},\mathbf{y}) & = & [\ 0\ \ 0\ \ \cdots\ \ \mathbf{f}(\mathbf{x})\ \ \cdots\ \ 0\ ] \\
\mathbf{w} & = & [\ \mathbf{w}_0\ \ \mathbf{w}_1\ \ \cdots\ \ \mathbf{w}_\mathbf{y}\ \ \cdots\ \ \mathbf{w}_k\ ]
\end{array}$$

- Scores and Predictions

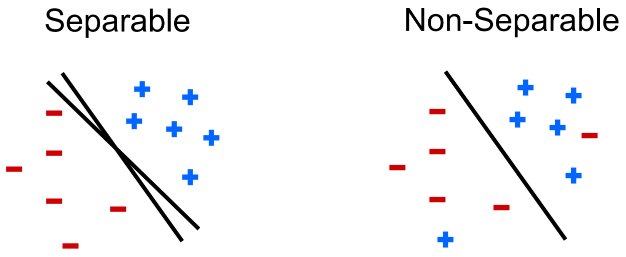
$$\mathit{score}(\mathbf{x}^i, \mathbf{y}, \mathbf{w}) = \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) = \mathbf{w_y}^\top \mathbf{f}(x^i)$$

$$\mathit{prediction}(\mathbf{x}^i, \mathbf{w}) = \operatorname*{arg\,max}_{\mathbf{y} \in \mathcal{Y}} \mathbf{w}^\top \mathbf{f}_i(\mathbf{y})$$


$\mathbf{w}^\top\mathbf{f}$
$\mathbf{w}_o^\top\mathbf{f}$
$\mathbf{w}_+^\top\mathbf{f}$
$\mathbf{w}_-^\top\mathbf{f}$

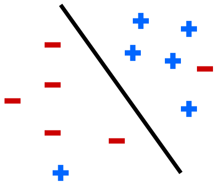
Separability

- A data set is (linearly) *separable* in a feature space if some linear classifier classifies all points correctly.

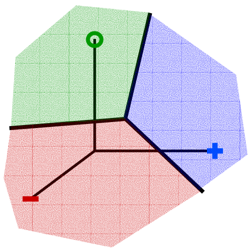

Separable
Non-Separable

- If a data set is separable, there are usually multiple separating hypotheses.

Caution about Diagrams



- A diagram you'll often see:
  - Two-class classification
  - Fractional feature values
  - Mixed regions → non-separable
  - Sample complexity



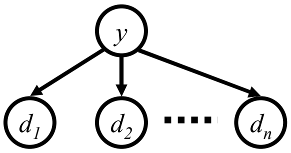
- Common NLP case:
  - Multi-class classification
    - Each input corresponds to |Y| points $f_i(y)$ (one per class)
  - (Mostly) 0/1 features
    - Data on the "corners"
  - Everything's separable
  - Coupon collection

Linear Models: Naïve-Bayes

- (Multinomial) Naïve-Bayes: $\mathbf{x}^i = d_1, d_2, \cdots d_n$

$$\begin{array}{rcllllllll} \mathbf{f}_i(\mathbf{y}) & = & [& 0 & 1, & \#v_1, & \#v_2, & \cdots & \#v_{|V|} & 0 \quad] \\ \mathbf{w} & = & [& \cdots & \log \mathsf{P}(y), & \log \mathsf{P}(v_1|y), & \log \mathsf{P}(v_2|y), & \cdots & \log \mathsf{P}(v_n|y) & \cdots \quad] \end{array}$$

$$\begin{aligned} \text{score}(\mathbf{x}_i, \mathbf{y}, \mathbf{w}) &= \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) \\ &= \log \mathsf{P}(\mathbf{y}) + \sum_k \#v_k \log \mathsf{P}(v_k|\mathbf{y}) \\ &= \log \left(\mathsf{P}(\mathbf{y}) \prod_k \mathsf{P}(v_k|\mathbf{y})^{\#v_k} \right) \\ &= \log \left(\mathsf{P}(\mathbf{y}) \prod_{d \in \mathbf{x}^i} \mathsf{P}(d|\mathbf{y}) \right) \\ &= \log \mathsf{P}(\mathbf{x}^i, \mathbf{y}) \end{aligned}$$



Bad Model Assumptions

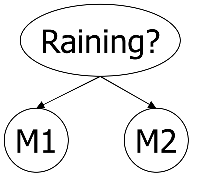
Reality

Raining

Sunny

P(+,+,r) = 3/8 P(−,−,r) = 1/8 P(+,+,s) = 1/8 P(−,−,s) = 3/8

NB Model



NB FACTORS:
- P(s) = 1/2
- P(+|s) = 1/4
- P(+|r) = 3/4

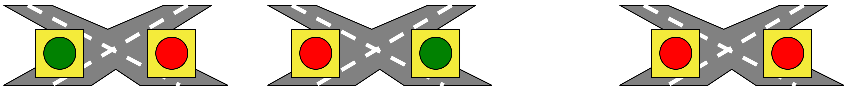
PREDICTIONS:
- P(r,+,+) = (½)(¾)(¾)
- P(s,+,+) = (½)(¼)(¼)
- P(r|+,+) = 9/10
- P(s|+,+) = 1/10

Worse Model Assumptions

Reality

Lights Working

Lights Broken

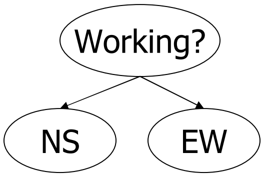


P(g,r,w) = 3/7

P(r,g,w) = 3/7

P(r,r,b) = 1/7

NB Model



NB FACTORS:

- P(w) = 6/7
- P(r|w) = 1/2
- P(g|w) = 1/2

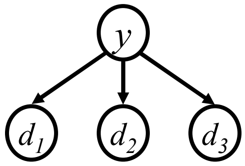
- P(b) = 1/7
- P(r|b) = 1
- P(g|b) = 0

Details: Stoplights

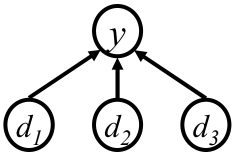
- What does the model say when both lights are red?
  - P(b,r,r) = (1/7)(1)(1) = 1/7 = 4/28
  - P(w,r,r) = (6/7)(1/2)(1/2) = 6/28 = 6/28
  - P(w|r,r) = 6/10!
- We'll guess that (r,r) indicates lights are working!
- Imagine if P(b) were boosted higher, to 1/2:
  - P(b,r,r) = (1/2)(1)(1) = 1/2 = 4/8
  - P(w,r,r) = (1/2)(1/2)(1/2) = 1/8 = 1/8
  - P(w|r,r) = 1/5!
- Changing the parameters bought accuracy at the expense of data likelihood
- Discriminative models can partially compensate for wrong models

Generative vs Discriminative

- Generative Models
  - Joint density over P(X,Y)
  - E.g. Naïve-Bayes, HMMs, PCFGs
  - Model assumptions allow decomposition into small factors which can be estimated independently
  - Do not set weights to account for feature interactions



- Discriminative Models
  - Predict Y given X, not always distributions
  - E.g. maximum entropy, SVMs, perceptrons
  - Set weights to account for feature interactions
  - Require *inference on training set* to evaluate hypotheses

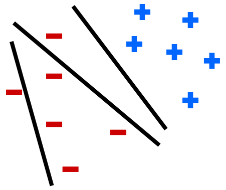


Linear Models: Perceptron

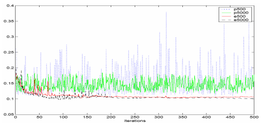
- Simple discriminative method for intuition

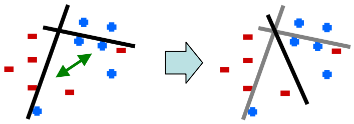
$$\mathbf{y}' = \arg\max_{\mathbf{y}} \mathbf{w}^\top \mathbf{f}_i(\mathbf{y})$$

$$\mathbf{w} \leftarrow \mathbf{w} + \underbrace{\eta\left(\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y}')\right)}_{\Delta_i(\mathbf{y}')}$$



- This is a procedure, not an optimization problem!
  - May not converge if non-separable
  - Noisy





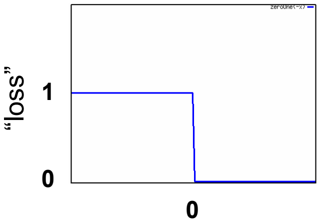
- Voted / averaged perceptron [Freund & Schapire 99, Collins 02]
  - Regularize / reduce variance by aggregating over iterations

Objective Functions

- Reminder: $score(\mathbf{x}^i, \mathbf{y}, \mathbf{w}) = \mathbf{w}^\top \mathbf{f}_i(\mathbf{y})$
- What do we want from weights?
  - Depends!
  - Minimize (training) errors?

$$\sum_i step\left(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \max_{\mathbf{y}\neq\mathbf{y}^i} \mathbf{w}^\top \mathbf{f}_i(\mathbf{y})\right)$$

  - This is the “zero-one loss”
    - Discontinuous, minimizing is NP-complete
    - Not really what we want anyway
  - Maxents and SVMs have losses related to the zero-one loss



$$\overbrace{\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \max_{\mathbf{y}\neq\mathbf{y}^i} \mathbf{w}^\top \mathbf{f}_i(\mathbf{y})}^{\text{“margin”}}$$

Linear Models: Maximum Entropy

- Maximum entropy (logistic regression)
  - Use the activations as probabilities:

$$\mathrm{P}(\mathbf{y}|\mathbf{x},\mathbf{w}) = \frac{\exp(\mathbf{w}^\top \mathbf{f}(\mathbf{x},\mathbf{y}))}{\sum_{\mathbf{y}'} \exp(\mathbf{w}^\top \mathbf{f}(\mathbf{x},\mathbf{y}'))} \begin{array}{l} \longleftarrow \text{Make positive} \\ \longleftarrow \text{Normalize} \end{array}$$

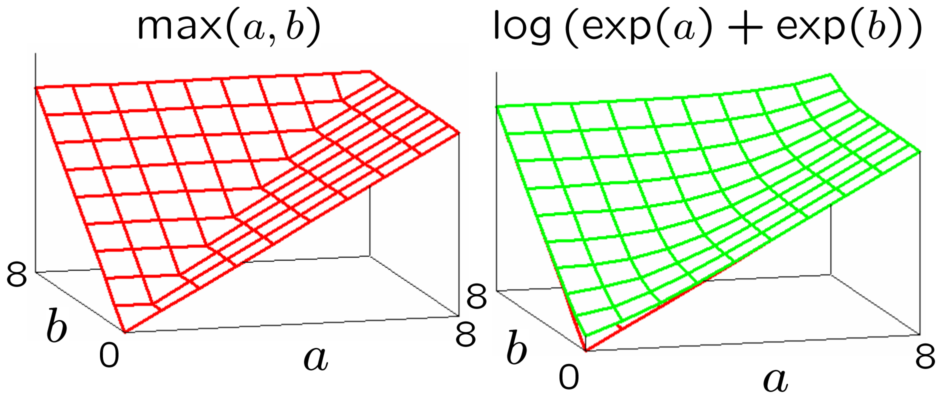
  - Maximize the (log) conditional likelihood of training data

$$\max_{\mathbf{w}} \ \log \prod_i \mathrm{P}(\mathbf{y}^i|\mathbf{x}^i,\mathbf{w}) = \sum_i \log\left(\frac{\exp(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i))}{\sum_{\mathbf{y}} \exp(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}))}\right)$$

$$\max_{\mathbf{w}} \ \sum_i \left(\underbrace{\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \log \sum_{\mathbf{y}} \exp(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}))}_{\text{“soft margin”}}\right)$$

“Soft-Max”

$$\max(a,b) \approx \log\left(\exp(a) + \exp(b)\right)$$



$\max(a,b)$

$\log\left(\exp(a) + \exp(b)\right)$

Maximum Entropy II

- Also: regularization (smoothing)

$$\max_{\mathbf{w}} \sum_i \left(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \log \sum_{\mathbf{y}} \exp(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y})) \right) - k||w||^2$$

- Maximize likelihood = Minimize “log-loss”

$$\min_{\mathbf{w}} \; k||w||^2 - \sum_i \left(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \log \sum_{\mathbf{y}} \exp(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y})) \right)$$

- Motivation
  - Connection to maximum entropy principle
  - Might want to do a good job of being uncertain on noisy cases…
  - … in practice, though, posteriors are pretty peaked

Log-Loss

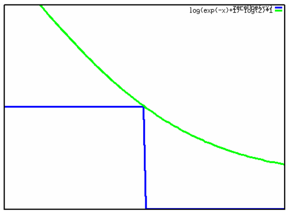
- If we view maxent as a minimization problem:

$$\min_{\mathbf{w}} \; k||w||^2 - \sum_i \left(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \log \sum_{\mathbf{y}} \exp(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y})) \right)$$

- This minimizes the “log-loss” on each example

$$-\left[\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \log \sum_{\mathbf{y}} \exp(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y})) \right]$$

$$-\log \left(\frac{\exp(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i))}{\sum_{\mathbf{y}} \exp(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}))} \right) = -\log \mathrm{P}(\mathbf{y}^i|\mathbf{x}^i, \mathbf{w})$$



$$\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \max_{\mathbf{y} \neq \mathbf{y}^i} \mathbf{w}^\top \mathbf{f}_i(\mathbf{y})$$

- Log-loss bounds zero-one loss

SVMs

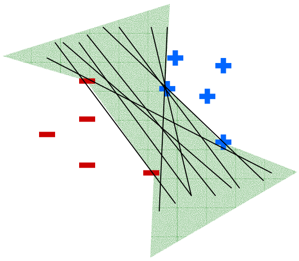
- SVM Try 1: Separate the training data

$$\forall i, \forall \mathbf{y} \neq \mathbf{y}^i \quad \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) \geq \mathbf{w}^\top \mathbf{f}_i(\mathbf{y})$$

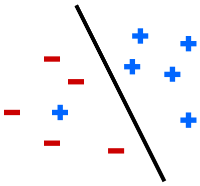
$$\mathbf{w}^\top \mathbf{f}(\ldots\text{win election}\ldots, \mathit{POLITICS}) \geq \mathbf{w}^\top \mathbf{f}(\ldots\text{win election}\ldots, \mathit{SPORTS})$$

$$\mathbf{w}^\top \mathbf{f}(\ldots\text{win election}\ldots, \mathit{POLITICS}) \geq \mathbf{w}^\top \mathbf{f}(\ldots\text{win election}\ldots, \mathit{OTHER})$$

1. This is an entire feasible space; need an objective function!



2. Training data may not even be separable



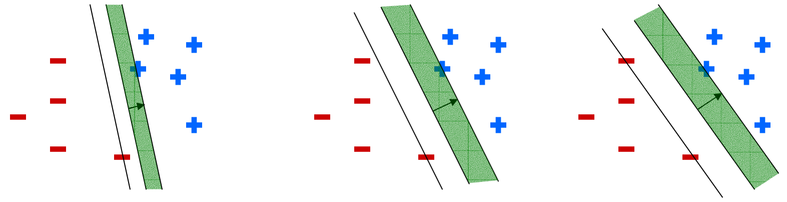
Maximum Margin

- SVM Try 2: find the maximum margin separator

$$\max_{||\mathbf{w}||\leq 1} \quad \gamma \qquad\qquad \ell_i(\mathbf{y}) = \begin{cases} 0 & \text{if } \mathbf{y} = \mathbf{y}^i \\ 1 & \text{if } \mathbf{y} \neq \mathbf{y}^i \end{cases}$$

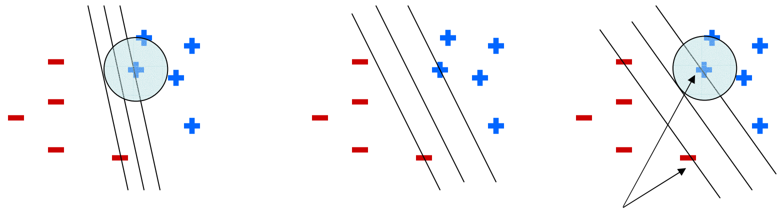
$$\text{s.t.} \quad \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) \geq \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \gamma \ell_i(\mathbf{y}) \quad \forall i, \forall \mathbf{y}$$

$$\begin{array}{ll}
\mathbf{w}^\top \mathbf{f}(\text{win election}, \mathit{POLITICS}) \geq \mathbf{w}^\top \mathbf{f}(\text{win election}, \mathit{SPORTS}) + \gamma & 1 \\
\mathbf{w}^\top \mathbf{f}(\text{win election}, \mathit{POLITICS}) \geq \mathbf{w}^\top \mathbf{f}(\text{win election}, \mathit{OTHER}) + \gamma & 1 \\
\mathbf{w}^\top \mathbf{f}(\text{win election}, \mathit{POLITICS}) \geq \mathbf{w}^\top \mathbf{f}(\text{win election}, \mathit{POLITICS}) & 0 \\
 & \ell
\end{array}$$



Why Max Margin?

- Why do this? Various arguments:
  - Decisions on training points are maximally robust to “feature jitter”
  - As we’ll see, solution depends only on the boundary cases, or *support vectors* (but remember how this diagram is broken!)
  - Sparse solutions (features not in support vectors get zero weight)
  - Generalization bound arguments



Support vectors

Max Margin / Small Norm

- SVM Try 3: find the smallest w which separates data

Remember this condition? →

$$\begin{aligned} \max_{||\mathbf{w}|| \leq 1} \quad & \gamma \\ \text{s.t.} \quad & \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) \geq \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \gamma \ell_i(\mathbf{y}) \quad \forall i, \forall \mathbf{y} \end{aligned}$$

- Instead of fixing the scale of w, we can fix $\gamma = 1$

$$\begin{aligned} \min_{\mathbf{w}} \quad & \frac{1}{2}||\mathbf{w}||^2 \\ \text{s.t.} \quad & \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^*) \geq \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + 1\ell_i(\mathbf{y}) \quad \forall i, \mathbf{y} \end{aligned}$$

Max Gamma to Min W

$$\begin{aligned} \max_{||\mathbf{w}|| \leq 1} \quad & \gamma \\ \text{s.t.} \quad & \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) \geq \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \gamma \ell_i(\mathbf{y}) \quad \forall i, \mathbf{y} \end{aligned}$$

$$\mathbf{w} = \gamma u$$

$$\gamma = 1/||u||$$

$$\begin{aligned} \max_{||\gamma u|| \leq 1} \quad & 1/||u||^2 \\ \text{s.t.} \quad & \gamma u^\top \mathbf{f}_i(\mathbf{y}^i) \geq \gamma u^\top \mathbf{f}_i(\mathbf{y}) + \gamma \ell_i(\mathbf{y}) \quad \forall i, \mathbf{y} \end{aligned}$$

$$\begin{aligned} \max_{||\gamma u|| \leq 1} \quad & 1/||u||^2 \\ \text{s.t.} \quad & u^\top \mathbf{f}_i(\mathbf{y}^i) \geq u^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y}) \quad \forall i, \mathbf{y} \end{aligned}$$

$$\begin{aligned} \min_{||\gamma u|| \geq 1} \quad & ||u||^2 \\ \text{s.t.} \quad & u^\top \mathbf{f}_i(\mathbf{y}^i) \geq u^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y}) \quad \forall i, \mathbf{y} \end{aligned}$$

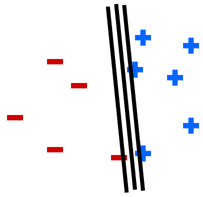
$$\begin{aligned} \min_{u} \quad & ||u||^2 \\ \text{s.t.} \quad & u^\top \mathbf{f}_i(\mathbf{y}^i) \geq u^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y}) \quad \forall i, \mathbf{y} \end{aligned}$$

$$\begin{aligned} \min_{u} \quad & \frac{1}{2}||u||^2 \\ \text{s.t.} \quad & u^\top \mathbf{f}_i(\mathbf{y}^i) \geq u^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y}) \quad \forall i, \mathbf{y} \end{aligned}$$

$$\begin{aligned} \min_{\mathbf{w}} \quad & \frac{1}{2}||\mathbf{w}||^2 \\ \text{s.t.} \quad & \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) \geq \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y}) \quad \forall i, \mathbf{y} \end{aligned}$$

Maximum Margin

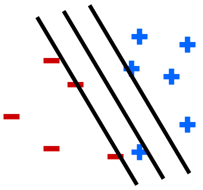
- SVM Try 4: allow for non-separability
  - Add slack to the constraints
  - Make objective pay (linearly) for slack:



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

$$\text{s.t.} \quad \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) + \xi_i \geq \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y}) \quad \forall i, \mathbf{y}$$

  - C is called the *capacity* of the SVM – the smoothing knob (more on this later)



- Learning:
  - Can stick this into Matlab if you want
  - Constrained optimization is hard; better methods!

Min-Max Formulation

- We have a constrained minimization

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

$$\text{s.t.} \quad \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) + \xi_i \geq \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y}) \quad \forall i, \mathbf{y}$$

- …but we can solve for ξ_i

$$\forall i, \mathbf{y}, \quad \xi_i \geq \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y}) - \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i)$$

$$\forall i, \quad \xi_i = \max_{\mathbf{y}} \left[\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y})\right] - \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i)$$

- Giving

$$\min_{\mathbf{w}} \frac{1}{2}||\mathbf{w}||^2 - C\sum_i \left(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \max_{\mathbf{y}} \left[\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y})\right]\right)$$

Max vs "Soft-Max" Margin

- SVMs:

$$\min_{\mathbf{w}} k||\mathbf{w}||^2 - \sum_i \left(\underbrace{\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \max_{\mathbf{y}} \left(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(y) \right)}_{\text{Hard (Penalized) Margin}} \right)$$

- Maxent:

$$\min_{\mathbf{w}} \quad k||w||^2 - \sum_i \left(\underbrace{\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \log \sum_{\mathbf{y}} \exp \left(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) \right)}_{\text{Soft Margin}} \right)$$

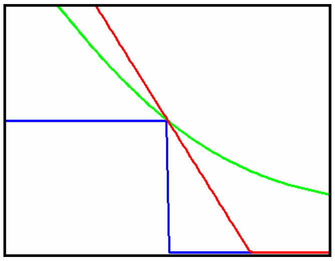
- Very similar! Both try to make the true score better than a function of the other scores.
  - The SVM tries to beat the augmented runner-up
  - The maxent classifier tries to beat the "soft-max"

Hinge Loss

- Consider the per-instance SVM objective:

$$\min_{\mathbf{w}} k||\mathbf{w}||^2 - \sum_i \left(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \max_{\mathbf{y}} \left[\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(y) \right] \right)$$

- This is called the "hinge loss"
  - Upper bounds zero-one loss
  - Unlike maxent / log loss, you stop gaining objective once the true label wins by enough
  - You can start from here and derive the SVM objective



$$\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \max_{\mathbf{y} \neq \mathbf{y}^i} \mathbf{w}^\top \mathbf{f}_i(\mathbf{y})$$

Loss Functions: I

- Zero-One Loss

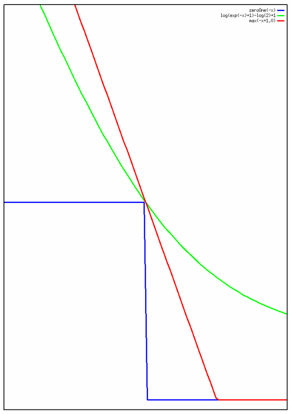
$$\sum_i step\left(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \max_{\mathbf{y} \neq \mathbf{y}^i} \mathbf{w}^\top \mathbf{f}_i(\mathbf{y})\right)$$

- Hinge

$$\sum_i \left(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \max_{\mathbf{y}} \left[\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(y)\right]\right)$$

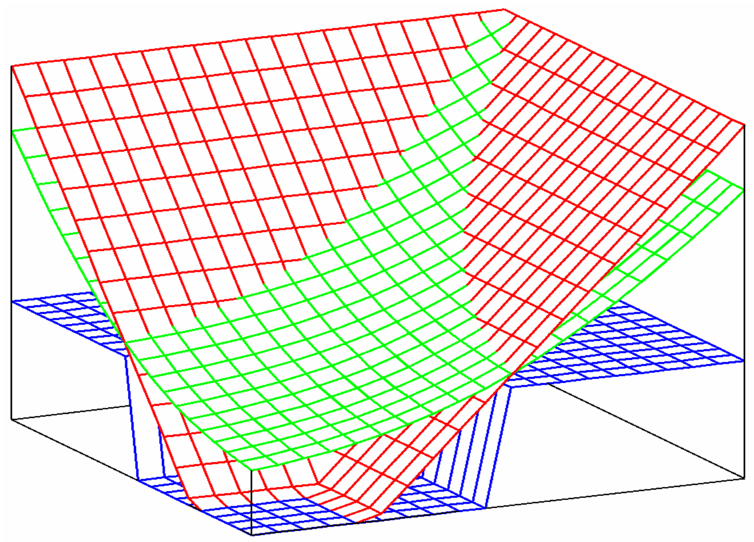
- Log

$$\sum_i \left(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \log \sum_{\mathbf{y}} \exp\left(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y})\right)\right)$$

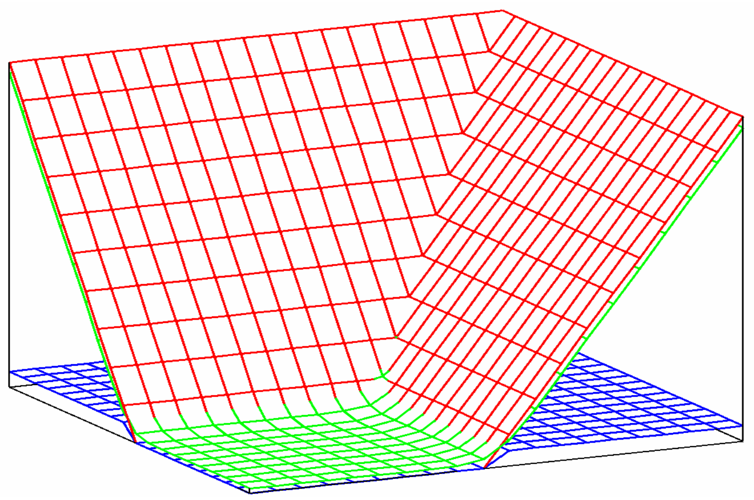


$$\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \max_{\mathbf{y} \neq \mathbf{y}^i} \mathbf{w}^\top \mathbf{f}_i(\mathbf{y})$$

Loss Functions: II



Loss Functions: III



Outline

- Part I: Flat Classification
  - Linear classifiers and loss functions
  - Primal and dual SVM formulations
  - Training SVMs
- Part II: Structured Classification
  - Structured linear classifiers
  - Factored learning formulations
  - Experimental results

Dual Formulation

- We want to optimize:

$$\begin{aligned} \min_{\mathbf{w},\xi} \quad & \frac{1}{2}||\mathbf{w}||^2 + C\sum_i \xi_i \\ \forall i, y \quad & \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) + \xi_i \geq \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y}^i) \end{aligned}$$

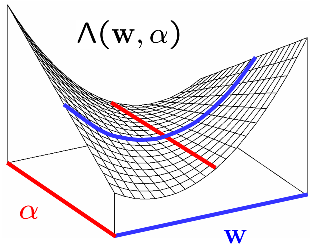
- This is hard because of the constraints.

- Solution: method of Lagrange multipliers

Lagrange Duality

- We start out with a constrained optimization problem:

$$\begin{aligned} f(\mathbf{w}^*) &= \min_{\mathbf{w}} \; f(\mathbf{w}) \\ & \quad\; g(\mathbf{w}) \geq 0 \end{aligned}$$



- We form the *Lagrangian*:

$$\Lambda(\mathbf{w}, \alpha) = f(\mathbf{w}) - \alpha\, g(\mathbf{w})$$

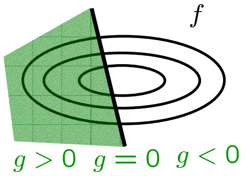
- This is useful because the constrained solution is a saddle point of Λ (we'll show this):

$$f(\mathbf{w}^*) = \underbrace{\min_{\mathbf{w}} \max_{\alpha \geq 0} \Lambda(\mathbf{w}, \alpha)}_{\textit{Primal problem in } \mathrm{w}} = \underbrace{\max_{\alpha \geq 0} \min_{\mathbf{w}} \Lambda(\mathbf{w}, \alpha)}_{\textit{Dual problem in } \alpha}$$

Primal Game

- Original: $f(\mathbf{w}^*) \;=\; \min_{\mathbf{w}} \; f(\mathbf{w}) \quad s.t. \;\; g(\mathbf{w}) \geq 0$
- Lagrangian: $\Lambda(\mathbf{w}, \alpha) = f(\mathbf{w}) - \alpha\, g(\mathbf{w})$

  $\Lambda(\mathbf{w}) = \max_{\alpha \geq 0} \, [f(\mathbf{w}) - \alpha\, g(\mathbf{w})]$



- Claim: primal game solves the original constrained problem:

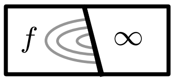
$$\min_{\mathbf{w}} \max_{\alpha \geq 0} \Lambda(\mathbf{w}, \alpha) = \min_{\mathbf{w}} \Lambda(\mathbf{w}) \;=\; f(\mathbf{w}^*)$$

- Proof: consider the value of

$$\Lambda(\mathbf{w}) = \max_{\alpha \geq 0} \, [f(\mathbf{w}) - \alpha\, g(\mathbf{w})]$$

$$g(\mathbf{w}) = 0 \Rightarrow f(\mathbf{w})$$
$$g(\mathbf{w}) > 0 \Rightarrow f(\mathbf{w})$$
$$g(\mathbf{w}) < 0 \Rightarrow \infty$$

$\Lambda(\mathbf{w})$



$\Rightarrow \quad \min_{\mathbf{w}} \Lambda(\mathbf{w}) = \min_{\mathbf{w}: g \geq 0} f(\mathbf{w}) = f(\mathbf{w}^*)$

Dual Game

- Original: $f(\mathbf{w}^*) \;=\; \min_{\mathbf{w}} \; f(\mathbf{w}) \quad s.t. \;\; g(\mathbf{w}) \geq 0$
- Lagrangian: $\Lambda(\mathbf{w}, \alpha) = f(\mathbf{w}) - \alpha\, g(\mathbf{w})$

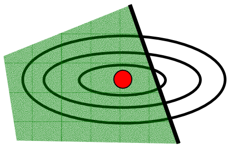
  $\Lambda(\alpha) = \min_{\mathbf{w}} \, [f(\mathbf{w}) - \alpha\, g(\mathbf{w})]$

- Claim: dual game also solves the original problem:

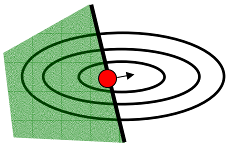
$$\max_{\alpha \geq 0} \min_{\mathbf{w}} \Lambda(\mathbf{w}, \alpha) = \max_{\alpha \geq 0} \Lambda(\alpha) \;=\; f(\mathbf{w}^*)$$

- Proof:

Case I: Constraint Inactive



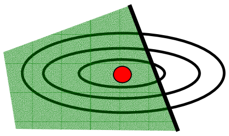
Case II: Constraint Active



Dual Game IIa

- Lagrangian: $\Lambda(\alpha) = \min_{\mathbf{w}} [f(\mathbf{w}) - \alpha\, g(\mathbf{w})]$
- Claim: $\max_{\alpha \geq 0} \min_{\mathbf{w}} \Lambda(\mathbf{w}, \alpha) = \max_{\alpha \geq 0} \Lambda(\alpha) \;=\; f(\mathbf{w}^*)$

Case I: Constraint Inactive



At $\mathbf{w}^*$, $g > 0$, so if $\alpha > 0$,

$f(\mathbf{w}^*) - \alpha\, g(\mathbf{w}^*) < f(\mathbf{w}^*)$,

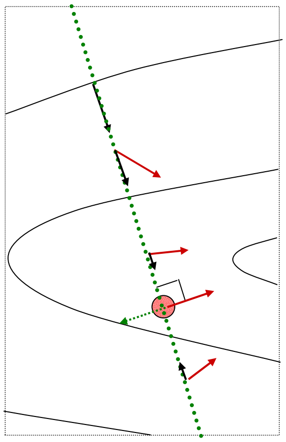
$\Lambda(\alpha) < f(\mathbf{w}^*)$

But $\Lambda(0) = f(\mathbf{w}^*)$

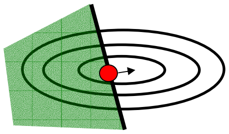
So $\max_{\alpha \geq 0} \Lambda(\alpha) = f(\mathbf{w}^*)$

Dual Game IIb

- Lagrangian: $\Lambda(\alpha) = \min_{\mathbf{w}} [f(\mathbf{w}) - \alpha\, g(\mathbf{w})]$
- Claim: $\max_{\alpha \geq 0} \min_{\mathbf{w}} \Lambda(\mathbf{w}, \alpha) = \max_{\alpha \geq 0} \Lambda(\alpha) \;=\; f(\mathbf{w}^*)$



Case II: Constraint Active



At $\mathbf{w}^*$, $g = 0$, so $\forall\alpha$,

$\Lambda(\mathbf{w}^*, \alpha) = f(\mathbf{w}^*) - \alpha\, g(\mathbf{w}^*) = f(\mathbf{w}^*)$,

so $\forall\alpha$, $\Lambda(\alpha) < f(\mathbf{w}^*)$

At $\mathbf{w}^*$, $\nabla f \neq 0$, but

$\exists\alpha^*$ s.t. $\nabla f(\mathbf{w}^*) = \alpha^* \nabla g(\mathbf{w}^*)$

At α^*, $\nabla\Lambda(\alpha^*, \mathbf{w}^*) = \nabla f - \alpha^* \nabla g = 0$

so $\Lambda(\alpha^*) = f(\mathbf{w}^*)$

Lagrangian for SVMs

- Primal constrained problem:

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

$$\forall i, y \quad \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) + \xi_i \geq \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y}^i)$$

- Lagrangian:

$$\min_{\mathbf{w},\xi} \max_{\alpha \geq 0} \quad \frac{1}{2}||\mathbf{w}||^2 + C\sum_i \xi_i - \sum_{i,y} \alpha_i(y) \left(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) - \ell_i(\mathbf{y}) + \xi_i\right)$$

Dual Formulation II

- Duality tells us that

$$\min_{\mathbf{w},\xi} \max_{\alpha \geq 0} \quad \frac{1}{2}||\mathbf{w}||^2 + C\sum_i \xi_i - \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y}) \left(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) - \ell_i(\mathbf{y}) + \xi_i\right)$$

has the same value as

$$\max_{\alpha \geq 0} \min_{\mathbf{w},\xi} \quad \frac{1}{2}||\mathbf{w}||^2 + C\sum_i \xi_i - \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y}) \left(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) - \ell_i(\mathbf{y}) + \xi_i\right)$$

- This is useful because if we think of the α's as constants, we have an unconstrained min in w and ξ that we can solve analytically.
- Then we end up with an optimization over α instead of w (easier).

Dual Formulation III

- Minimize the Lagrangian for fixed α's:

$$\Lambda(\mathbf{w},\xi,\alpha) \;=\; \frac{1}{2}||\mathbf{w}||^2 + C\sum_i \xi_i - \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\left(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) - \ell_i(\mathbf{y}) + \xi_i\right)$$

$$\frac{\partial \Lambda(\mathbf{w},\xi,\alpha)}{\partial \mathbf{w}} \;=\; \mathbf{w} - \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\left(\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y})\right)$$

$$\frac{\partial \Lambda(\mathbf{w},\xi,\alpha)}{\partial \mathbf{w}} = 0 \quad \Rightarrow \quad \mathbf{w} = \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\left(\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y})\right)$$

$$\frac{\partial \Lambda(\mathbf{w},\xi,\alpha)}{\partial \xi_i} \;=\; C - \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})$$

$$\frac{\partial \Lambda(\mathbf{w},\xi,\alpha)}{\partial \xi_i} = 0 \quad \Rightarrow \quad \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y}) = C$$

Dual Formulation IV

- We now know that for fixed α, the minimum of

$$\Lambda(\mathbf{w},\xi,\alpha) \;=\; \frac{1}{2}||\mathbf{w}||^2 + C\sum_i \xi_i - \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\left(\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) - \ell_i(\mathbf{y}) + \xi_i\right)$$

obeys $\sum_{i,\mathbf{y}} \alpha_i(\mathbf{y}) = C$ and $\mathbf{w} = \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\left(\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y})\right)$

- Plugging these back into Λ:

$$\min_{\mathbf{w},\xi} \Lambda(\mathbf{w},\xi,\alpha) \;=\; -\frac{1}{2}\left|\left|\sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\left(\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y})\right)\right|\right|^2 + \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\ell_i(\mathbf{y})$$

Dual Formulation V

- This doesn't reference the primal weights w at all, so we can now worry about the outer max problem:

$$\max_{\alpha \geq 0} \quad \Lambda(\alpha) = -\frac{1}{2}\left|\left|\sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\left(\mathbf{f}_i(\mathbf{y}^*) - \mathbf{f}_i(\mathbf{y})\right)\right|\right|^2 + \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\ell_i(\mathbf{y})$$

$$\text{s.t.} \quad \sum_{\mathbf{y}} \alpha_i(\mathbf{y}) = C \quad \forall i$$

- And this solves the original constrained primal:

$$\max_{\alpha \geq 0} \Lambda(\alpha) = \max_{\alpha \geq 0} \min_{\mathbf{w},\xi} \Lambda(\mathbf{w}, \xi, \alpha) \quad = \quad f(\mathbf{w}^*)$$

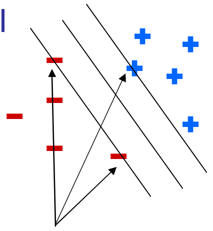
$$\mathbf{w} = \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y}) \left(\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y})\right)$$

What are the Alphas?

- Each example (and label) gave to a primal constraint

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

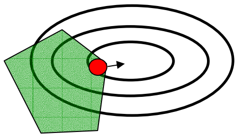
$$\text{s.t.} \quad \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) + \xi_i \geq \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y}) \quad \forall i, \mathbf{y}$$



Support vectors

- In the solution, an $\alpha_i(\mathrm{y})$ will be:
  - Zero if that constraint is inactive
  - Positive if that constrain is active
  - i.e. positive on the support vectors
- Support vectors form the weights:

$$\mathbf{w} = \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y}) \left(\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y})\right)$$



Outline

- Part I: Flat Classification
  - Linear classifiers and loss functions
  - Primal and dual SVM formulations
  - Training SVMs
- Part II: Structured Classification
  - Structured linear classifiers
  - Factored learning formulations
  - Experimental results

Back to Learning SVMs

- We want to find α which maximize

$$\max_{\alpha \geq 0} \quad \Lambda(\alpha) = -\frac{1}{2}\left\|\sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\left(\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y})\right)\right\|^2 + \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\ell_i(\mathbf{y})$$

$$\text{s.t.} \quad \sum_{\mathbf{y}} \alpha_i(\mathbf{y}) = C \quad \forall i$$

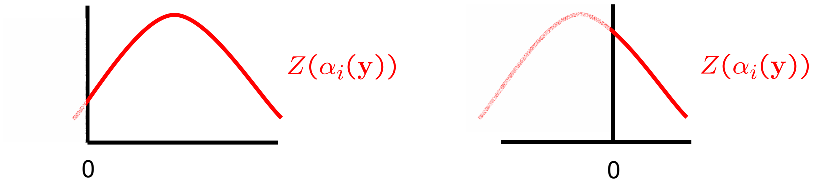
- This is a quadratic program:
  - Can be solved with general QP or convex optimizers
  - But they don't scale well to large problems
  - Cf. maxent models work fine with general optimizers (e.g. CG, L-BFGS)
- How would a special purpose optimizer work?

Coordinate Ascent I

- Consider the separable (soft-margin) SVM problem:

$$\max_{\alpha\geq 0} Z(\alpha) = \max_{\alpha\geq 0} \; -\frac{1}{2}\left\|\sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\left(\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y})\right)\right\|^2 + \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\ell_i(\mathbf{y})$$

- In coordinate ascent, we maximize one variable at a time
- Despite all the mess, Z is just a quadratic in each $\alpha_i(\mathrm{y})$



- If the unconstrained argmin on a coordinate is at a negative α, just clip to zero!

Coordinate Ascent II

- Ordinarily, treating coordinates independently is a bad idea, but here the update is very fast and simple

$$\alpha_i(\mathbf{y}) \leftarrow \max\left(0, \alpha_i(\mathbf{y}) + \frac{\ell_i(\mathbf{y}) - \left(\sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})\left(\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y})\right)\right)^\top \left(\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y})\right)}{\left\|\left(\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y})\right)\right\|^2}\right)$$

- So we visit each axis many times, but each visit is quick
- This approach works fine for the separable case

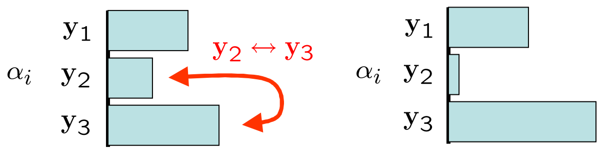
Bi-Coordinate Descent I

- In the non-separable case, it's (a little) harder:

$$\max_{\alpha \geq 0} \quad \Lambda(\alpha) = -\frac{1}{2} \left\| \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y}) \left(\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y}) \right) \right\|^2 + \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y}) \ell_i(\mathbf{y})$$

$$\text{s.t.} \quad \sum_{\mathbf{y}} \alpha_i(\mathbf{y}) = C \quad \forall i$$

- Here, we can't update just a single alpha, because of the sum-to-C constraints
- Instead, we can optimize two at once, shifting "mass" from one y to another:

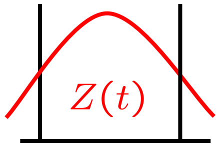


Bi-Coordinate Descent II

- Choose an example i, and two labels y_1 and y_2:

$$t = \frac{(\ell_i(\mathbf{y}_1) - \ell_i(\mathbf{y}_2)) - \left(\sum_{i,\mathbf{y}} \alpha_i(\mathbf{y}) \left(\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y}) \right) \right)^\top (\mathbf{f}_i(\mathbf{y}_2) - \mathbf{f}_i(\mathbf{y}_1))}{||\mathbf{f}_i(\mathbf{y}_2) - \mathbf{f}_i(\mathbf{y}_1)||^2}$$

$$\mathbf{y}_1 \rightarrow \min(\mathbf{y}_1 + t, \mathbf{y}_1 + \mathbf{y}_2)$$
$$\mathbf{y}_2 \rightarrow \max(\mathbf{y}_2 - t, 0)$$



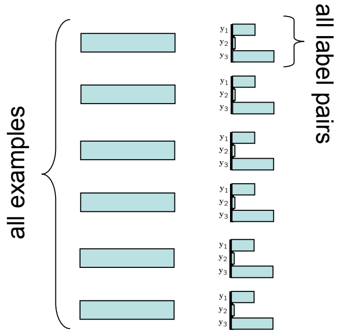
- This is a sequential minimal optimization update, but it's not the same one as in [Platt 98]

SMO

- Naïve SMO:

$$\mathbf{w} = \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y}) \left(\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y}) \right)$$

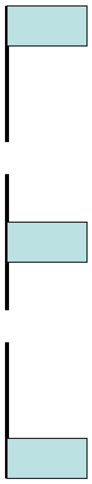
```
$\forall i \quad \alpha_i(\mathbf{y}^i) = C$
while (not converged) {
    visit each example $i$ {
        for each pair of labels $(\mathrm{y}_1, \mathrm{y}_2)$ {
            $bi\text{-}coordinate\text{-}update(i, \mathrm{y}_1, \mathrm{y}_2)$
        }
    }
}
```



- Time per iteration: $O(|x||\mathcal{Y}|^2)$
- Smarter SMO:
  - Can speed this up by being clever about skipping examples and label pairs which will make little or no difference

DOCUMENT	FEATURES	FEATURE DELTAS
	S-win, S-game	-- 0 --
win game	P-win, P-game	PW=1, SW=-1, PG=1,...
	O-win, O-game	OW=1, SW=-1, OG=1,...
	S-win, S-vote	SW=1, PW=-1, SV=1,...
win vote	P-win, P-vote	-- 0 --
	O-win, O-vote	OW=1, PW=-1, OV=1,...
	S-movie	SM=1, OM=-1
movie	P-movie	PM=1, OM=-1
	O-movie	-- 0 --

ALPHAS



WEIGHTS

$$\mathrm{w} = \sum_{i,\mathrm{y}} \alpha_i(\mathrm{y}) \left(\mathrm{f}_i(\mathrm{y}^i) - \mathrm{f}_i(\mathrm{y}) \right)$$

-- 0 --

Outline

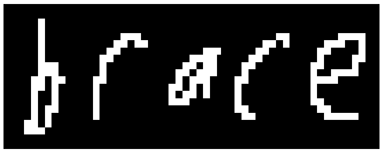
- Part I: Flat Classification
  - Linear classifiers and loss functions
  - Primal and dual SVM formulations
  - Training SVMs
- Part II: Structured Classification
  - Structured linear classifiers
  - Factored learning formulations
  - Experimental results

http://www.cartoonbank.com

"Don't worry, Howard. The big questions are multiple choice."

Handwriting Recognition

x



y

→ **brace**

Sequential structure

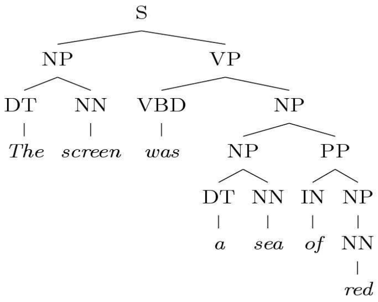
CFG Parsing

x

The screen was a sea of red

→

y



Recursive structure

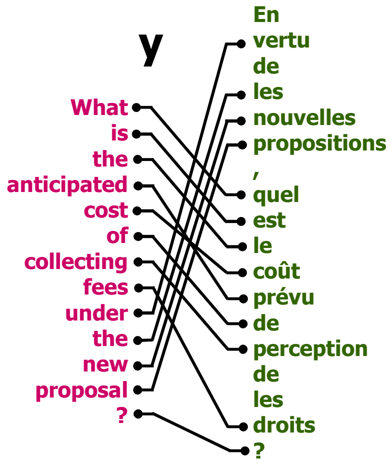
Bilingual Word Alignment

x

What is the anticipated cost of collecting fees under the new proposal?

En vertu des nouvelles propositions, quel est le coût prévu de perception des droits?

y



Combinatorial structure

Structured Models

$$prediction(\mathbf{x},\mathbf{w}) = \operatorname*{arg\,max}_{\mathbf{y}\in\mathcal{Y}(\mathbf{x})} score(\mathbf{x},\mathbf{y},\mathbf{w})$$

space of feasible outputs

Assumption:

$$score(\mathbf{x},\mathbf{y},\mathbf{w}) = \mathbf{w}^\top \mathbf{f}(\mathbf{x},\mathbf{y}) = \sum_p \mathbf{w}^\top \mathbf{f}(\mathbf{x}_p,\mathbf{y}_p)$$

Score = sum of local “part” scores

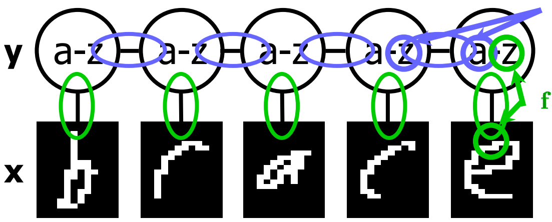
Parts = nodes, edges, productions

Chain Markov Net (aka CRF*)

$$P(\mathbf{y} \mid \mathbf{x}) \propto \prod_j \underbrace{\phi(\mathbf{x}_j, y_j)} \prod_{jk} \underbrace{\phi(\mathbf{x}_{jk}, y_j, y_k)}$$

$$\phi(\mathbf{x}_j, y_j) = \exp\left\{\mathbf{w}_{\mathsf{N}}^\top \mathbf{f}_{\mathsf{N}}(\mathbf{x}_j, y_j)\right\}$$ N = Node

$$\phi(\mathbf{x}_{jk}, y_j, y_k) = \exp\left\{\mathbf{w}_{\mathsf{E}}^\top \mathbf{f}_{\mathsf{E}}(\mathbf{x}_{jk}, y_j, y_k)\right\}$$ E = Edge



$\mathbf{f}_{\mathsf{E}}(\mathbf{x}_{jk}, y_j, y_k)$

$[\cdots \mathrm{I}(y_j = \text{'z'}, y_k = \text{'a'}) \cdots]$

$\mathbf{f}_{\mathsf{N}}(\mathbf{x}_j, y_j)$

$[\cdots \mathrm{I}(\mathbf{x}_j[3,4] = 1, y_j = \text{'z'}) \cdots]$

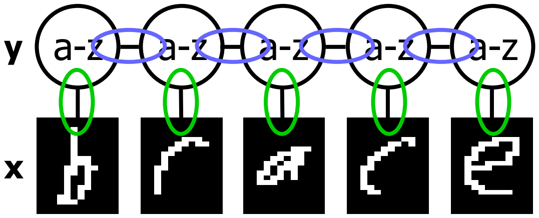
*Lafferty et al. 01

Chain Markov Net (aka CRF*)

$$P(\mathbf{y} \mid \mathbf{x}) \propto \underbrace{\prod_j \phi(\mathbf{x}_j, y_j)} \underbrace{\prod_{jk} \phi(\mathbf{x}_{jk}, y_j, y_k)} = \exp\left\{\mathbf{w}^\top \mathbf{f}(\mathbf{x}, \mathbf{y})\right\}$$

$$\prod_j \phi(\mathbf{x}_j, y_j) = \exp\left\{\sum_j \mathbf{w}_{\mathsf{N}}^\top \mathbf{f}_{\mathsf{N}}(\mathbf{x}_j, y_j)\right\} = \exp\left\{\mathbf{w}_{\mathsf{N}}^\top \mathbf{f}_{\mathsf{N}}(\mathbf{x}, \mathbf{y})\right\}$$

$$\prod_{jk} \phi(\mathbf{x}_{jk}, y_j, y_k) = \exp\left\{\sum_{jk} \mathbf{w}_{\mathsf{E}}^\top \mathbf{f}_{\mathsf{E}}(\mathbf{x}_{jk}, y_j, y_k)\right\} = \exp\left\{\mathbf{w}_{\mathsf{E}}^\top \mathbf{f}_{\mathsf{E}}(\mathbf{x}, \mathbf{y})\right\}$$



$$\mathbf{f}_{\mathsf{N}}(\mathbf{x}, \mathbf{y}) \equiv \sum_j \mathbf{f}_{\mathsf{N}}(\mathbf{x}_j, y_j)$$

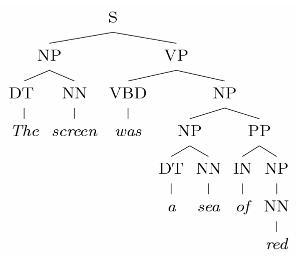
$$\mathbf{f}_{\mathsf{E}}(\mathbf{x}, \mathbf{y}) \equiv \sum_{jk} \mathbf{f}_{\mathsf{E}}(\mathbf{x}_{jk}, y_j, y_k)$$

$$\mathbf{f}(\mathbf{x}, \mathbf{y}) \equiv \begin{pmatrix} \mathbf{f}_{\mathsf{N}}(\mathbf{x}, \mathbf{y}) \\ \mathbf{f}_{\mathsf{E}}(\mathbf{x}, \mathbf{y}) \end{pmatrix} \quad \mathbf{w} \equiv \begin{pmatrix} \mathbf{w}_{\mathsf{N}} \\ \mathbf{w}_{\mathsf{E}} \end{pmatrix}$$

*Lafferty et al. 01

CFG Parsing

$$P(\mathbf{y} \mid \mathbf{x}) \propto \prod_{A \to \alpha \in (\mathbf{x},\mathbf{y})} \phi(A \to \alpha)$$



$\mathbf{f} : \mathcal{X} \times \mathcal{Y} \to \Re^d$

#(NP → DT NN)

...

#(PP → IN NP)

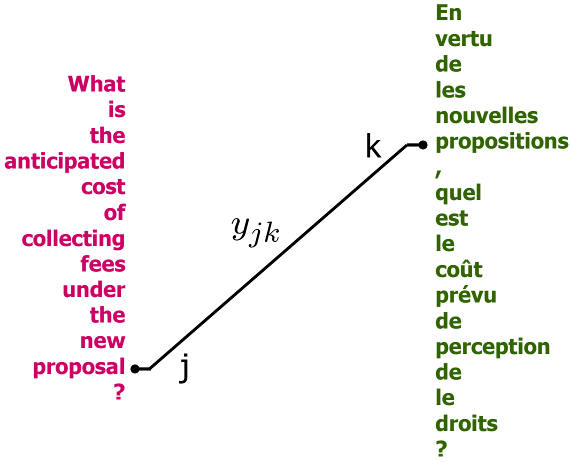
...

#(NN → 'sea')

$$\prod_{A \to \alpha \in (\mathbf{x},\mathbf{y})} \exp\left\{\mathbf{w}^\top \mathbf{f}(A \to \alpha)\right\} = \exp\{\mathbf{w}^\top \mathbf{f}(\mathbf{x},\mathbf{y})\}$$

Bilingual Word Alignment

$$\sum_{y_{jk} \in \mathbf{y}} \mathbf{w}^\top \mathbf{f}(\mathbf{x}_{jk}) = \mathbf{w}^\top \mathbf{f}(\mathbf{x},\mathbf{y})$$



$\mathbf{f}(\mathbf{x}_{jk})$

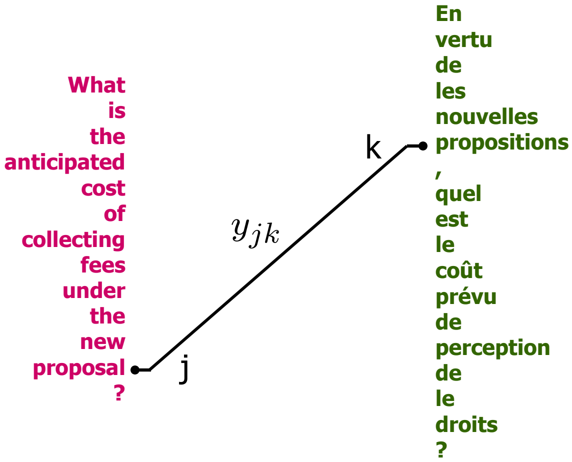
- association
- position
- orthography

Probabilistic Alignment?

$$P(\mathbf{y} \mid \mathbf{x}) = \frac{\exp\{\mathbf{w}^\top \mathbf{f}(\mathbf{x}, \mathbf{y})\}}{\sum_{\mathbf{y}'} \exp\{\mathbf{w}^\top \mathbf{f}(\mathbf{x}, \mathbf{y}')\}}$$

← ***#P-Complete***

Need to sum over all possible matchings



OCR Example

- We want:

$\arg\max_{\mathbf{y}}\ \mathbf{w}^\top \mathbf{f}($ $, \mathbf{y}) \quad = \quad$ “brace”

- Equivalently:

$\mathbf{w}^\top \mathbf{f}($, “brace”$) \quad > \quad \mathbf{w}^\top \mathbf{f}($, “aaaaa”$)$

$\mathbf{w}^\top \mathbf{f}($, “brace”$) \quad > \quad \mathbf{w}^\top \mathbf{f}($, “aaaab”$)$

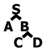
…

$\mathbf{w}^\top \mathbf{f}($, “brace”$) \quad > \quad \mathbf{w}^\top \mathbf{f}($, “zzzzz”$)$

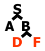
$\Big\}$ **a lot!**

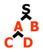
Parsing Example

- We want:

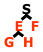
$\arg\max_{y}\ w^{\top} f($ 'It was red' $, y) =$ 

- Equivalently:

$w^{\top} f($'It was red', $) > w^{\top} f($'It was red',  $)$

$w^{\top} f($'It was red', $) > w^{\top} f($'It was red',  $)$

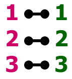
$\ldots$

$w^{\top} f($'It was red', $) > w^{\top} f($'It was red',  $)$

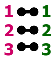
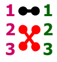
a lot!

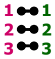
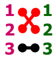
Alignment Example

- We want:

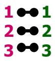
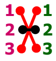
$\arg\max_{y}\ w^{\top} f($ 'What is the' 'Quel est le' $, y) =$ 

- Equivalently:

$w^{\top} f($'What is the' 'Quel est le',  $) > w^{\top} f($'What is the' 'Quel est le',  $)$

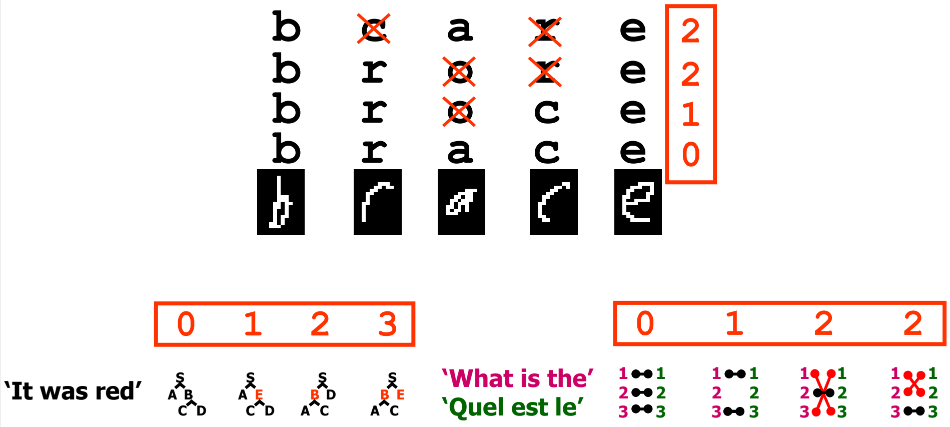
$w^{\top} f($'What is the' 'Quel est le',  $) > w^{\top} f($'What is the' 'Quel est le',  $)$

$\ldots$

$w^{\top} f($'What is the' 'Quel est le',  $) > w^{\top} f($'What is the' 'Quel est le',  $)$

a lot!

Structured Loss



Max Margin Estimation

- Given training example $\mathbf{x}^i, \mathbf{y}^i$ we want:

$$\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) > \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) \quad \forall i, \mathbf{y} \neq \mathbf{y}^i$$

$$\boxed{\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) \geq \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \gamma \ell_i(\mathbf{y}) \quad \forall i, \mathbf{y}}$$

- Maximize loss weighted margin: $\gamma \ell_i(\mathbf{y})$

$\ell_i(\mathbf{y}) = \sum_j I(y_j^i \neq y_j)$ # of mistakes in **y**

*Collins 02, Altun et al 03, Taskar 03

Large margin estimation

- Brute force enumeration

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

$$\text{s.t.} \quad \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) + \xi_i \geq \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y}), \quad \forall i, \mathbf{y}$$

- Min-max formulation

$$\min_{\mathbf{w}} \quad \frac{1}{2}||\mathbf{w}||^2 - C\left(\sum_i \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \max_{\mathbf{y}}\left[\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y})\right]\right)$$

  - Plug-in linear program for loss-augmented inference

$$\max_{\mathbf{y}}\left[\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y})\right]$$

Min-max formulation

$$\max_{\mathbf{y}}\left[\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y})\right]$$

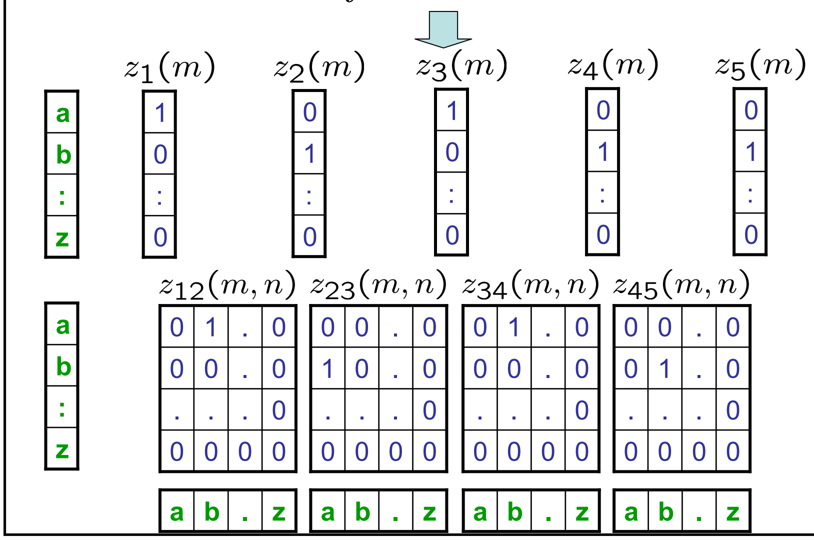
Assume linear loss (Hamming): $\ell_i(\mathbf{y}) = \sum_p \ell_{i,p}(\mathbf{y}_p)$

DP Inference $\max_{\mathbf{y}}\left[\sum_p \mathbf{w}^\top \mathbf{f}(\mathbf{x}_p, \mathbf{y}_p) + \ell_{i,p}(\mathbf{y}_p)\right]$

LP inference $\max_{\substack{\mathbf{z}\geq 0;\\ \mathbf{A}\mathbf{z}\leq\mathbf{b};}} \mathbf{q}^\top \mathbf{z}$

y ⇒ z Map for Markov Nets

$\mathbf{y}$ = 'ababb'



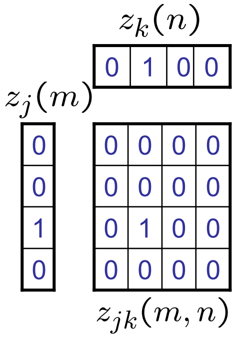
$$z_1(m)=\begin{array}{c|c} \mathsf a & 1\\ \mathsf b & 0\\ : & :\\ \mathsf z & 0\end{array}\quad z_2(m)=\begin{array}{c}0\\1\\:\\0\end{array}\quad z_3(m)=\begin{array}{c}1\\0\\:\\0\end{array}\quad z_4(m)=\begin{array}{c}0\\1\\:\\0\end{array}\quad z_5(m)=\begin{array}{c}0\\1\\:\\0\end{array}$$

$$z_{12}(m,n)=\begin{array}{c|cccc} & \mathsf a & \mathsf b & . & \mathsf z\\ \hline \mathsf a & 0 & 1 & . & 0\\ \mathsf b & 0 & 0 & . & 0\\ : & . & . & . & 0\\ \mathsf z & 0 & 0 & 0 & 0\end{array}\quad z_{23}(m,n)=\begin{array}{c|cccc} & \mathsf a & \mathsf b & . & \mathsf z\\ \hline \mathsf a & 0 & 0 & . & 0\\ \mathsf b & 1 & 0 & . & 0\\ : & . & . & . & 0\\ \mathsf z & 0 & 0 & 0 & 0\end{array}$$

$$z_{34}(m,n)=\begin{array}{c|cccc} & \mathsf a & \mathsf b & . & \mathsf z\\ \hline \mathsf a & 0 & 1 & . & 0\\ \mathsf b & 0 & 0 & . & 0\\ : & . & . & . & 0\\ \mathsf z & 0 & 0 & 0 & 0\end{array}\quad z_{45}(m,n)=\begin{array}{c|cccc} & \mathsf a & \mathsf b & . & \mathsf z\\ \hline \mathsf a & 0 & 0 & . & 0\\ \mathsf b & 0 & 1 & . & 0\\ : & . & . & . & 0\\ \mathsf z & 0 & 0 & 0 & 0\end{array}$$

Markov Net Inference LP

$$\left.\begin{aligned}\max_{\mathbf z}\ &\sum_{j,m} z_j(m)\left[\mathbf w^\top \mathbf f_{\mathsf N}(\mathbf x_j,m)+\ell_j(m)\right]\\ +&\sum_{jk,m,n} z_{jk}(m,n)\left[\mathbf w^\top \mathbf f_{\mathsf E}(\mathbf x_{jk},m,n)+\ell_{jk}(m,n)\right]\end{aligned}\right\}\begin{array}{l}\mathbf q^\top\mathbf z\\ \mathbf q=\mathbf F^\top\mathbf w+\boldsymbol\ell\end{array}$$



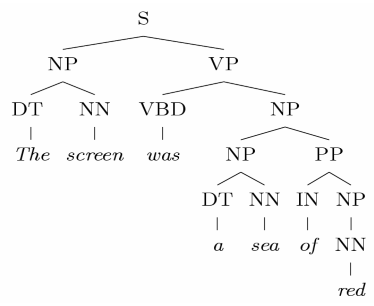
$$z_k(n)=\begin{array}{cccc}0 & 1 & 0 & 0\end{array}\qquad z_j(m)=\begin{array}{c}0\\0\\1\\0\end{array}\qquad z_{jk}(m,n)=\begin{array}{cccc}0&0&0&0\\0&0&0&0\\0&1&0&0\\0&0&0&0\end{array}$$

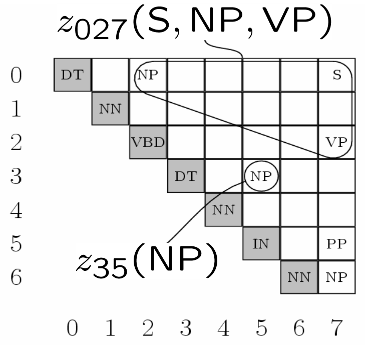
$z_j(m)\geq 0;\qquad z_{jk}(m,n)\geq 0;$

$$\left.\begin{array}{ll}\text{normalization} & \sum_m z_j(m)=1\\[2ex] \text{agreement} & \sum_n z_{jk}(m,n)=z_j(m)\end{array}\right\}\ A\mathbf z=\mathbf b$$

Has integral solutions **z** for chains, trees

CFG Chart





- CNF tree = set of two types of parts:
  - Constituents (A, s, e)
  - CF-rules (A → B C, s, m, e)

$$\mathbf{f}(\mathbf{x},\mathbf{y}) = \textstyle\sum_{p\in\mathbf{y}} \mathbf{f}(\mathbf{x},p)$$

CFG Inference LP

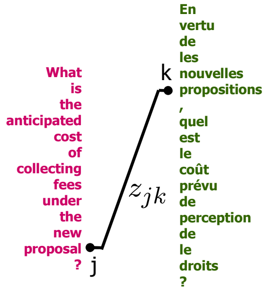
$$\max_{\mathbf{z}} \left.\sum_{\substack{s<m<e\\ A\to B\ C}} z_{sme}(ABC)\left[\mathbf{w}^\top \mathbf{f}(\mathbf{x}_{sme}, ABC) + \ell_{sme}(ABC)\right]\right\} \begin{array}{l}\mathbf{q}^\top\mathbf{z}\\ \mathbf{q} = \mathbf{F}^\top\mathbf{w} + \ell\end{array}$$

$$\text{s.t.} \quad z_{se}(A) \geq 0 \qquad\qquad z_{sme}(ABC) \geq 0$$

$$\left.\begin{array}{ll}
\text{root} & \displaystyle\sum_{A} z_{0,n}(A) = 1 \\
\text{inside} & \displaystyle z_{se}(A) = \sum_{\substack{s<m<e\\ B,C}} z_{sme}(A,B,C) \\
\text{outside} & \displaystyle z_{se}(A) = \sum_{\substack{e<m\leq n\\ B,C}} z_{sme}(B,A,C) + \sum_{\substack{0\leq m<s\\ B,C}} z_{sme}(B,C,A)
\end{array}\right\} A\mathbf{z} = \mathbf{b}$$

Has integral solutions **z**

Matching Inference LP


What is the anticipated cost of collecting fees under the new proposal ?
En vertu de les nouvelles propositions , quel est le coût prévu de perception de le droits ?
k
j
z_{jk}

$$\max_{\mathbf{z}} \quad \left.\sum_{jk} z_{jk}\left[\mathbf{w}^\top \mathbf{f}(\mathbf{x}_{jk}) + \ell_{jk}\right]\right\} \mathbf{q}^\top \mathbf{z} \qquad \mathbf{q} = \mathbf{F}^\top \mathbf{w} + \ell$$

$$\text{s.t.} \quad z_{jk} \geq 0$$

degree

$$\left.\begin{array}{l} \sum_{k} z_{jk} \leq 1 \\ \sum_{j} z_{jk} \leq 1 \end{array}\right\} A\mathbf{z} \leq \mathbf{b}$$

Has integral solutions **z**

LP Duality Recap

- Linear programming duality
  - Variables ⇒ constraints
  - Constraints ⇒ variables
- Optimal values are the same
  - When both feasible regions are bounded

$$\begin{array}{ll} \max_{\mathbf{z}} & \mathbf{c}^\top \mathbf{z} \\ \text{s.t.} & \mathbf{A}\mathbf{z} \leq \mathbf{b}; \\ & \mathbf{z} \geq 0. \end{array} \quad \Longleftrightarrow \quad \begin{array}{ll} \min_{\lambda} & \mathbf{b}^\top \lambda \\ \text{s.t.} & \mathbf{A}^\top \lambda \geq \mathbf{c}; \\ & \lambda \geq 0. \end{array}$$

Min-max formulation

$$\min_{\mathbf{w}} \ \frac{1}{2}||\mathbf{w}||^2 - C\left(\sum_i \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \max_{\mathbf{y}}\left[\mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y})\right]\right)$$

$$\max_{\substack{\mathbf{A}_i \mathbf{z}_i \leq \mathbf{b}_i \\ \mathbf{z}_i \geq 0}} \ \mathbf{q}_i^\top \mathbf{z}_i \quad \Longleftrightarrow \quad \min_{\substack{\mathbf{A}_i^\top \lambda_i \geq \mathbf{q}_i \\ \lambda_i \geq 0}} \ \mathbf{b}_i^\top \lambda_i$$

LP duality

$$\boxed{\begin{array}{ll} \min\limits_{\mathbf{w},\lambda} & \frac{1}{2}||\mathbf{w}||^2 - C\left(\sum\limits_i \ \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \mathbf{b}_i^\top \lambda_i\right) \\ \text{s.t.} & \mathbf{A}_i^\top \lambda_i \geq \mathbf{q}_i; \quad \lambda_i \geq 0 \end{array}}$$

$$\mathbf{q}_i = \mathbf{F}_i^\top \mathbf{w} + \ell_i$$

Min-max formulation summary

$$\boxed{\begin{array}{ll} \min\limits_{\mathbf{w},\lambda} & \frac{1}{2}||\mathbf{w}||^2 - C\left(\sum\limits_i \ \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \mathbf{b}_i^\top \lambda_i\right) \\ \text{s.t.} & \mathbf{A}_i^\top \lambda_i \geq \mathbf{F}_i^\top \mathbf{w} + \ell_i; \quad \lambda_i \geq 0, \ \forall i. \end{array}}$$

- Formulation produces concise QP for
  - Low-treewidth Markov networks
  - Context free grammars
  - Bipartite matchings
  - Many other problems with compact LP inference

*Taskar et al 04

Factored Primal/Dual

$$\min_{\mathbf{w},\lambda} \quad \frac{1}{2}||\mathbf{w}||^2 - C\left(\sum_i \; \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) - \mathbf{b}_i^\top \lambda_i\right)$$

$$\text{s.t.} \quad \mathbf{A}_i^\top \lambda_i \geq \mathbf{F}_i^\top \mathbf{w} + \ell_i; \qquad \lambda_i \geq 0, \quad \forall i.$$

By QP duality $\mathbf{w} = \sum_i C\mathbf{f}_i(\mathbf{y}^i) - \mathbf{F}_i\mu_i$

$$\max_{\mu} \quad \sum_i \ell_i^\top \mu_i - \frac{1}{2}\left|\left|\sum_i C\mathbf{f}_i(\mathbf{y}^i) - \mathbf{F}_i\mu_i]\right|\right|^2$$

$$\text{s.t.} \quad \mathbf{A}_i\mu_i \leq C\mathbf{b}_i; \quad \mu_i \geq 0, \quad \forall i.$$

Dual inherits structure from problem-specific inference LP

Variables $\boldsymbol{\mu}$ correspond to a decomposition of $\boldsymbol{\alpha}$ variables of the flat case

Unfactored Primal/Dual

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

$$\text{s.t.} \quad \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}^i) + \xi_i \geq \mathbf{w}^\top \mathbf{f}_i(\mathbf{y}) + \ell_i(\mathbf{y}), \quad \forall i, \mathbf{y}$$

By QP duality $\mathbf{w} = \sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})[\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y})]$

$$\max_{\alpha} \quad \sum_{i,\mathbf{y}} \ell_i(\mathbf{y})\alpha_i(\mathbf{y}) - \frac{1}{2}\left|\left|\sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})[\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y})]\right|\right|^2$$

$$\text{s.t.} \quad \sum_{\mathbf{y}} \alpha_i(\mathbf{y}) = C; \quad \alpha_i \geq 0, \quad \forall i.$$

Exponentially many constraints/variables

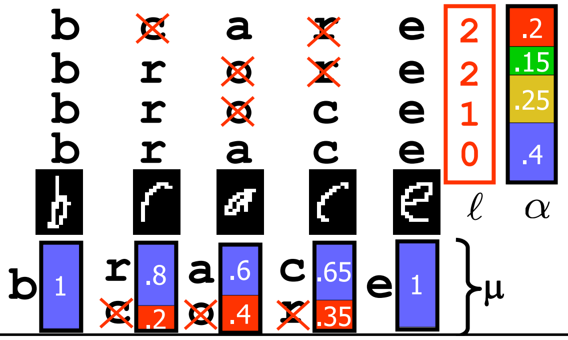
The Connection

$$\max_{\mu} \sum_i \ell_i^\top \mu_i - \frac{1}{2}\left\|\sum_i C\mathbf{f}_i(\mathbf{y}^i) - \mathbf{F}_i\mu_i\right\|^2$$

s.t. $\mathbf{A}_i\mu_i = C\mathbf{b}_i; \quad \mu_i \geq 0, \quad \forall i.$

$$\max_{\alpha} \sum_{i,\mathbf{y}} \ell_i(\mathbf{y})\alpha_i(\mathbf{y}) - \frac{1}{2}\left\|\sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})[\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y})]\right\|^2$$

s.t. $\sum_{\mathbf{y}} \alpha_i(\mathbf{y}) = C; \quad \alpha_i \geq 0, \quad \forall i.$



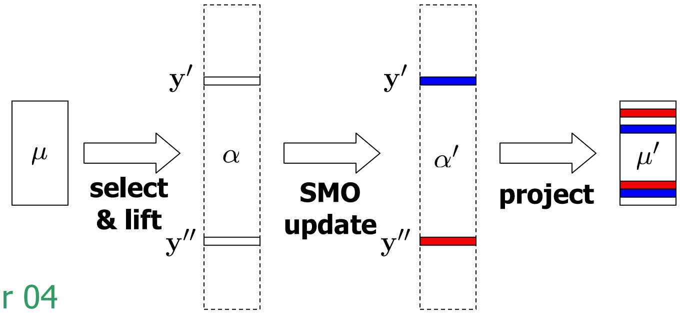
Structured SMO

$$\max_{\mu} \sum_i \ell_i^\top \mu_i - \frac{1}{2}\left\|\sum_i C\mathbf{f}_i(\mathbf{y}^i) - \mathbf{F}_i\mu_i\right\|^2$$

s.t. $\mathbf{A}_i\mu_i = C\mathbf{b}_i; \quad \mu_i \geq 0, \quad \forall i.$

$$\max_{\alpha} \sum_{i,\mathbf{y}} \ell_i(\mathbf{y})\alpha_i(\mathbf{y}) - \frac{1}{2}\left\|\sum_{i,\mathbf{y}} \alpha_i(\mathbf{y})[\mathbf{f}_i(\mathbf{y}^i) - \mathbf{f}_i(\mathbf{y})]\right\|^2$$

s.t. $\sum_{\mathbf{y}} \alpha_i(\mathbf{y}) = C; \quad \alpha_i \geq 0, \quad \forall i.$



*Taskar 04

Outline

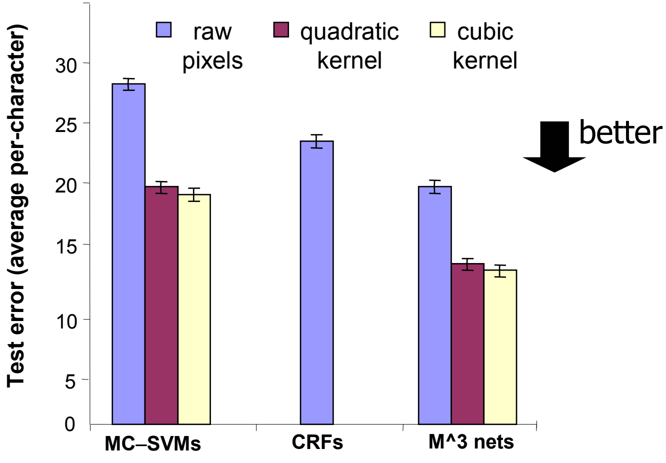
- Part I: Flat Classification
  - Linear classifiers and loss functions
  - Primal and dual SVM formulations
  - Training SVMs

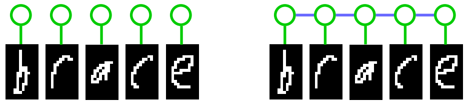
- Part II: Structured Classification
  - Structured linear classifiers
  - Factored learning formulations
  - Experimental results

Handwriting Recognition

Length: ~8 chars
Letter: 16x8 pixels
10-fold Train/Test
5000/50000 letters
600/6000 words

Models:
Multiclass-SVMs
CRFs
M^3 nets





*Taskar et al 03

Experimental Setup

- Standard Penn treebank split (2-21/22/23)
- Generative baselines
  - Klein & Manning 03 and Collins 99
- Discriminative
  - Basic = max-margin version of K&M 03
  - Lexical & Lexical + Aux
- Lexical features (on constituent parts only)

t_{s-1} [t_s … t_e] t_{e+1} ← predicted tags

x_{s-1} [x_s … x_e] x_{e+1}

- Auxillary features
  - Flat classifier using same features
  - Prediction of K&M 03 on each span

Results for sentences ≤40 words

Model	LP	LR	F_1
Generative	86.37	85.27	85.82
Lexical+Aux*	**87.56**	**86.85**	**87.20**
Collins 99*	85.33	85.94	85.73

*Trained only on sentences ≤20 words

*Taskar et al 04

Example

The Egyptian president said he would visit Libya today to resume the talks.

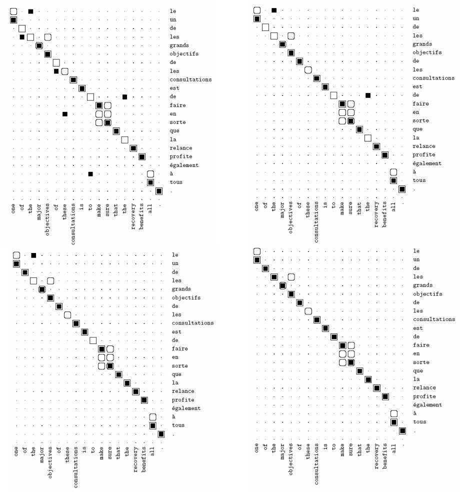
Generative model: Libya today is base NP

Lexical model: today is a one word constituent

Word Alignment Results

- Hansards, 2M unlabeled, 100 labeled sentences

Model	AER
Dice	36.0
IBM 4	9.7
MM-Dice	29.8
+Distance	17.2
+Shape/Freq	14.3
+Next/Common	9.6



Generative/Discriminative Trade-offs

- Inference on training:
  - Discriminative methods require (repeated) inference on the training set, over the domains where the parameters interact
  - Generative models are primarily estimated from statistics of the training set (counting)
  - Inference can be much, much slower than counting
- Accounting for interactions:
  - Discriminative estimates take into account feature interactions, non-independence (note that conjunctive features are required to actually model interactions)
- Bias / variance
  - Discriminative methods tend to have higher variance, generative ones tend to have higher bias – but in general the discriminative techniques win on accuracy if properly regularized

Likelihood/Margin Trade-offs

- Same as maxent vs. SVMs:
  - Sparse solutions, robust to “feature jitter”
  - Margin-based training often more accurate when posteriors are not needed
- Plus: unnormalizable models
  - For some models (e.g., matchings and a subclass of Markov networks), margin is tractable, likelihood is not!

Conclusions

- Today's tutorial:
  - Flat SVMs from scratch
    - Objective functions and properties
    - Primal and dual formulations
    - How to learn them
  - Structured max-margin models
    - Concise, factored form
    - Efficient algorithms, strong empirical results
    - Applications: sequences, trees, matchings
- Coming soon:
  - Sequence modeling toolkit including M3Ns

http://www.cs.berkeley.edu/~klein
http://www.cs.berkeley.edu/~taskar

References

Y. Altun, I. Tsochantaridis, and T. Hofmann. *Hidden Markov support vector machines.* ICML03.

M. Collins. *Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms*. EMNLP02

K. Crammer and Y. Singer. *On the algorithmic implementation of multiclass kernel-based vector machines.* JMLR01

J. Lafferty, A. McCallum, and F. Pereira. *Conditional random fields: Probabilistic models for segmenting and labeling sequence data.* ICML04

B. Taskar, C. Guestrin, D. Koller. *Maximum margin Markov Networks*. NIPS*03*

B. Taskar, D. Klein, M. Collins, D. Koller, C. Manning. *Maximum margin Parsing*. EMNLP*04*

B. Taskar. *Learning structured prediction models: a large margin approach. Stanford Univ. Thesis, 2004*