CS446 Introduction to Machine Learning (Fall 2013)
University of Illinois at Urbana-Champaign
http://courses.engr.illinois.edu/cs446

LECTURE 12: MULTICLASS CLASSIFICATION

Prof. Julia Hockenmaier
juliahmr@illinois.edu

Last lecture's key concepts

Review of SVMs

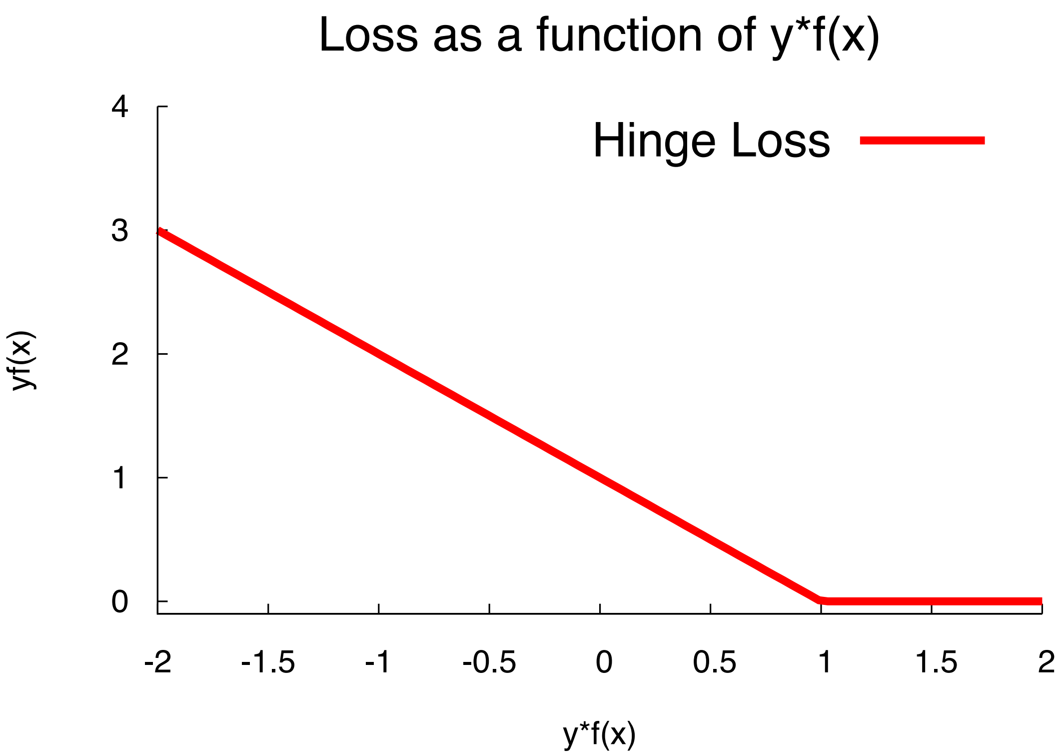
Dealing with outliers: Soft margins

Soft margin SVMs and Regularization

SGD for soft margin SVMs

Hinge loss and SVMs

$$L_{hinge}(y^{(n)}, f(\mathbf{x}^{(n)})) = \max(0, 1 - y^{(n)}f(\mathbf{x}^{(n)}))$$



Case 0: $f(\mathbf{x}) = 1$
$\mathbf{x}$ is a support vector
Hinge loss = 0

Case 1: $f(\mathbf{x}) > 1$
$\mathbf{x}$ outside of margin
Hinge loss = 0

Case 2: $0 < yf(\mathbf{x}) < 1$:
$\mathbf{x}$ inside of margin
Hinge loss = $1 - yf(\mathbf{x})$

Case 3: $yf(\mathbf{x}) < 0$:
$\mathbf{x}$ misclassified
Hinge loss = $1 - yf(\mathbf{x})$

(Hard) SVMs

If the training data is linearly separable, there will be a decision boundary $\mathbf{w}\mathbf{x} + b = 0$ that perfectly separates it, and where all the items have a functional distance of at least 1: $y^{(i)}(\mathbf{w}\mathbf{x}^{(i)} + b) \geq 1$

We can find $\mathbf{w}$ and b with a quadratic program:

$$\underset{\mathbf{w},b}{\operatorname{argmin}} \frac{1}{2}\mathbf{w}\cdot\mathbf{w}$$
$$\textit{subject to}$$
$$y_i(\mathbf{w}\cdot\mathbf{x}_i + b) \geq 1 \;\; \forall i$$

Soft margin SVMs

$$\underset{\mathbf{w},b,\xi_i}{\operatorname{argmin}} \frac{1}{2}\mathbf{w}\cdot\mathbf{w} + C\sum_{i=1}^{n}\xi_i$$

$$\textit{subject to}$$

$$\xi_i \geq 0 \quad \forall i$$

$$y_i(\mathbf{w}\cdot\mathbf{x}_i + b) \geq (1-\xi_i)\forall i$$

ξ_i (slack): hinge loss of $\mathbf{x}_i$

C (cost): how much do we have to pay for misclassifying $\mathbf{x}_i$

We want to minimize $C\sum_i \xi_i$ and maximize the margin

C controls the tradeoff between margin and training error

Soft SVMs = Regularized Hinge Loss:

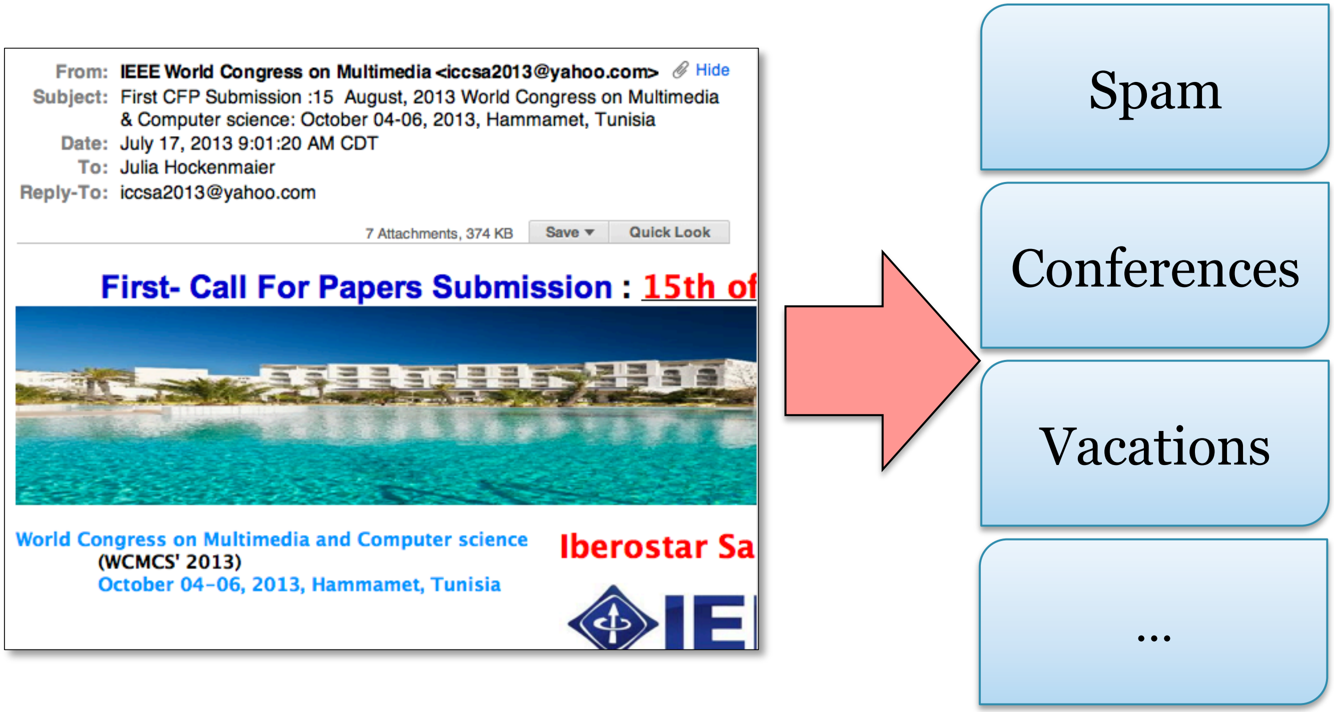
$$\underset{\mathbf{w},b}{\operatorname{argmin}} \frac{1}{2}\mathbf{w}\cdot\mathbf{w} + C\sum_{i=1}^{n} L_{hinge}(y^{(n)}, \mathbf{x}^{(n)})$$

We minimize both the l2-norm of the weight vector $||\mathbf{w}|| = \sqrt{\mathbf{ww}}$ and the hinge loss.

Minimizing the norm of **w** is called regularization.

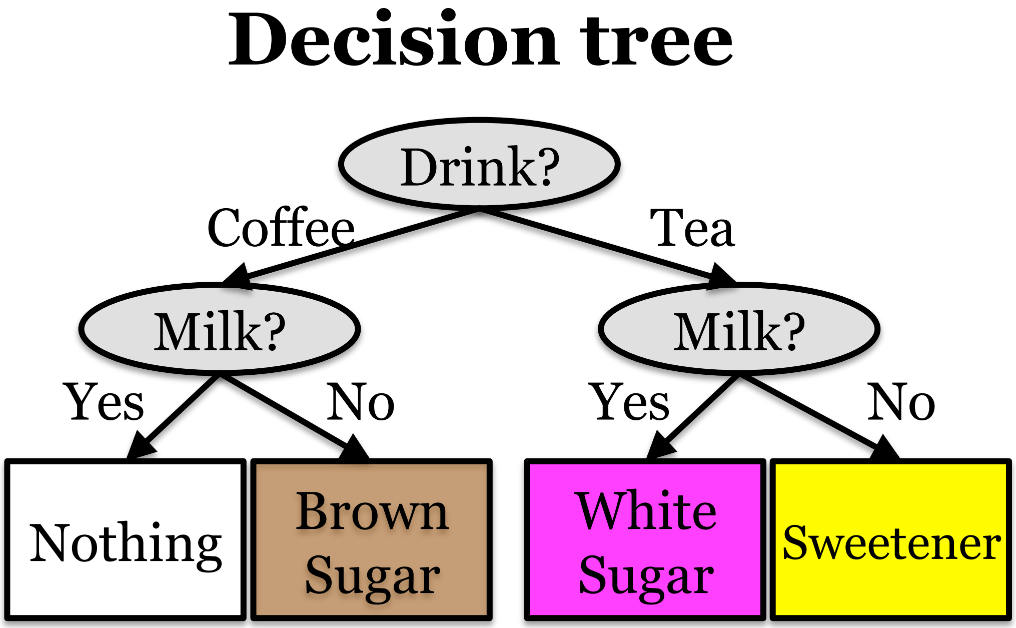
Multiclass Classification

Multiclass classification



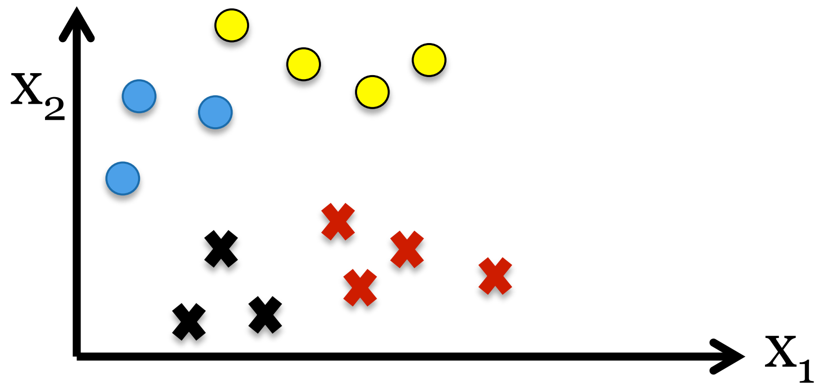
Assign one of *k* labels to the input
{Spam, Conferences, Vacations,…}

Decision trees



Decision trees can be easily applied to multiclass problems

Linear classifiers for multi-class classification



Using binary classifiers:

- **One vs. all:** one binary classifier for each class
  Pick the one with the highest score f($\mathbf{x}$)
- **All vs. all:** one binary classifier for each
  pair of classes; pick the majority vote

One-vs.-all

Train K (or K-1) classifiers $\mathbf{f_k(x)}$

– One binary classifier $f_k(\mathbf{x})$ for each class k

– Negative examples: drawn from all other K-1 classes

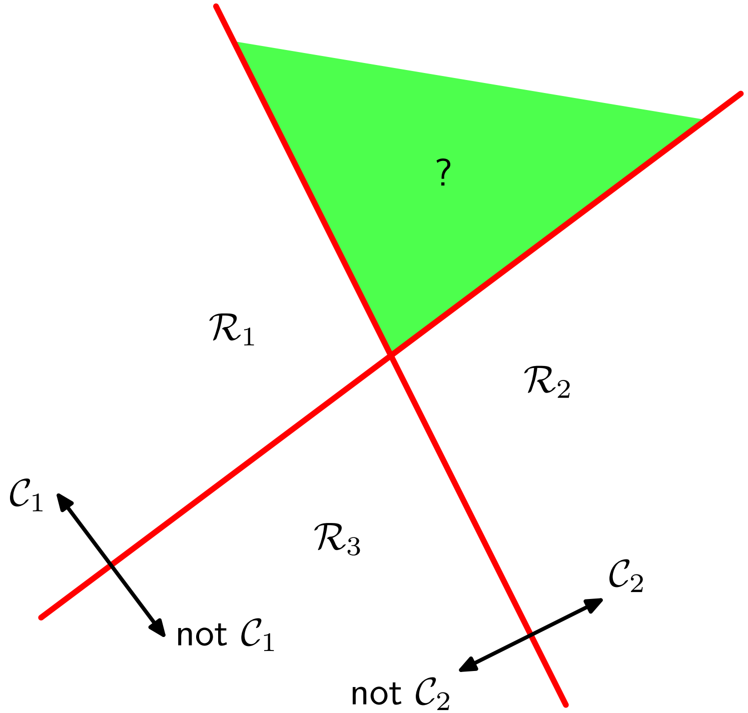
Testing:

– Pick class $k^* = \text{argmax}_k\ f_k(\mathbf{x})$

Which classifier is most confident about **x**?

It's also possible to train K-1 classifiers for classes 1..K-1, and assign class K if $f_k(\mathbf{x}) < 0$ for all $k \in \{1 \ldots K-1\}$

One-vs.-all: what can go wrong?



Green region is in both C_1 and C_2

All-vs.-all

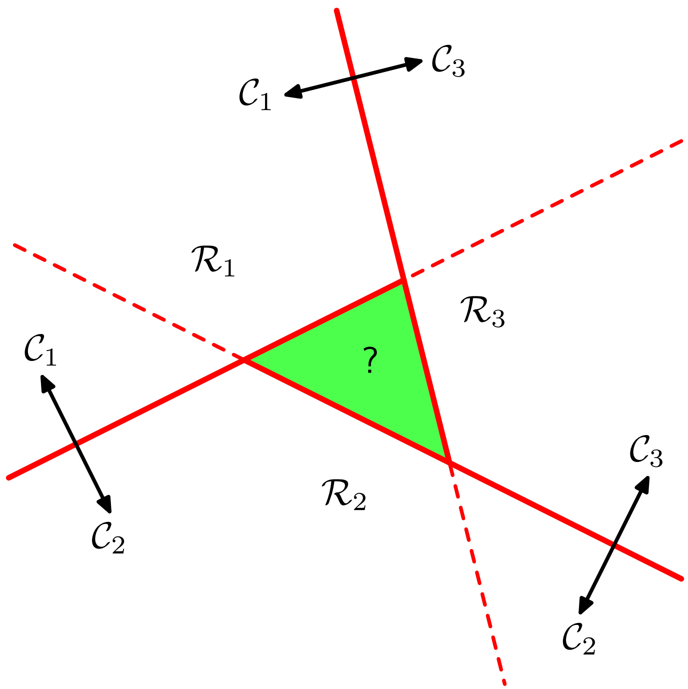
Train K×(K–1) classifiers $\mathbf{f}_{kl}(\mathbf{x})$:

- One binary classifier $f_{kl}(\mathbf{x})$ for each pair of classes k, l (trained on the examples from classes k, l)
- Problem: sparsity (few training examples for each classifier)

Testing:

- Pick class k^* by majority vote:
  $k^* = \mathrm{argmax}_k \mid k\colon f_{k'l'}(\mathbf{x}) = k \mid$
  $\mid k\colon f_{k'l'}(\mathbf{x}) = k \mid$ = #classifiers that prefer k over some other class l

All-vs.-all: what can go wrong?



Green region is not in any class.

Multiclass classifier

A *single* K-class discriminant function, consisting of K linear functions of the form

$$f_k(\mathbf{x}) = \mathbf{w}_k\mathbf{x} + w_{k0}$$

Assign k if $f_k(\mathbf{x}) > f_j(\mathbf{x})$ for all $j \neq k$

Multiclass classifier

Decision boundary between C_k and C_j :

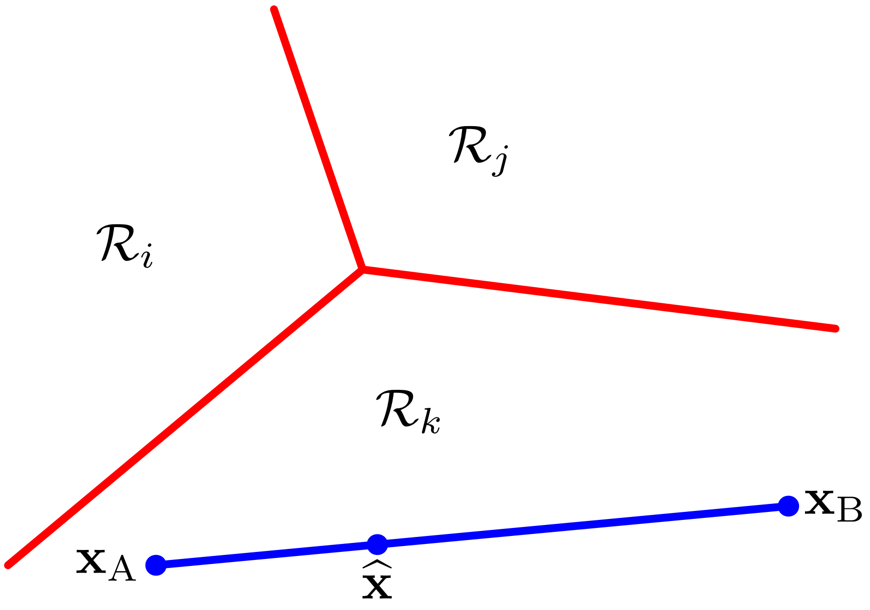
$$f_k(\mathbf{x}) = f_j(\mathbf{x})$$

$$\mathbf{w}_k\mathbf{x} + w_{ko} = \mathbf{w}_j\mathbf{x} + w_{jo}$$

$$(\mathbf{w}_k - \mathbf{w}_j)\mathbf{x} + (w_{ko} - w_{jo}) = 0$$

Multiclass classifier

Each class is associated with a convex region in the example space:



Multiclass classifier

Task: Learn K linear functions of the form

$$f_k(\mathbf{x}) = \mathbf{w}_k\mathbf{x} + w_{ko}$$

such that for each training example $(\mathbf{x}, k)$,

$$f_k(\mathbf{x}) > f_j(\mathbf{x}) \text{ for all } j \neq k$$

How do we train the $\mathbf{w}_k$s?

What are the negative examples?

Binary classification again…

With $Y = \{+1, -1\}$: $f(\mathbf{x}) = \text{sign}(\mathbf{wx})$

Since $Y = \{+1, -1\}$, this can be rewritten as

$$f(\mathbf{x}) = \text{argmax}_{y \in Y}(\mathbf{w} \cdot y\mathbf{x})$$

Note that $\mathbf{w} \cdot y\mathbf{x}$ contains the class label y.

Think of $y\mathbf{x}$ as a class-sensitive feature mapping.

We can generalize this to any class-sensitive feature mapping $F(y,\mathbf{x})$: $Y \times X \rightarrow R^d$

This means that features can now depend on the class label y

Class-sensitive features

Idea: Let the value of feature x_j depend on y.

Task: Document Classification

X = Documents (sets of words)

Y = K topics

Class-dependent TF-IDF scores:

- Term frequency $tf(j, \mathbf{x})$: frequency of word j in $\mathbf{x}$
- Class-dependent document frequency $df(j, k)$
  #training documents that are *not* of topic k in which word j occurs
- m = total number of training documents

$$\text{TF-IDF}(j,\mathbf{x},y) = tf(j,x) \times \log(m/df(j,y))$$

Multi-vector representation

(aka Kesler construction)
Idea: Map n-dimensional feature vectors to (sparse) $K \times n$-dimensional vector $F(y,\mathbf{x})$ in which each class corresponds to n dimensions:

$$Y = \{1 \ldots K\}, \quad X = R^n \quad F: X \times Y \rightarrow R^{Kn}$$

$F(1,\mathbf{x}) = [x_1,\ldots,x_n, 0, \ldots,0]$
$F(i,\mathbf{x}) = [0, \ldots,0, x_1,\ldots,x_n, 0, \ldots,0]$
$F(K,\mathbf{x}) = [0, \ldots,0, x_1,\ldots,x_n]$

Now $\mathbf{w} = [\mathbf{w}_1;\ldots; \mathbf{w}_K]$, and $\mathbf{w}F(y,\mathbf{x}) = \mathbf{w}_y\mathbf{x}$

Multiclass classification

Learning a multiclass classifier: Find **w** such that for all training items ($\mathbf{x}$,y_i)

$$y_i = \operatorname{argmax}_y \mathbf{w}F(y,\mathbf{x})$$

Equivalently, for all ($\mathbf{x}$, y_i) and all k≠i:

$$\mathbf{w}F(y_i,\mathbf{x}) > \mathbf{w}F(y_k,\mathbf{x})$$

So what are the negative examples?

We don't have negative examples anymore. Instead, we want the score of the correct class, $\mathbf{w}F(y_i,\mathbf{x})$, to be higher than the scores of all other classes for that item:

$$\mathbf{w}F(y_i,\mathbf{x}) > \mathbf{w}F(y_k,\mathbf{x})$$

Or: $\mathbf{w}F(y_i,\mathbf{x}) - \mathbf{w}F(y_k,\mathbf{x}) > 0$

$$\mathbf{w}[F(y_i,\mathbf{x}) - F(y_k,\mathbf{x})] > 0$$

In the multi-vector representation:

$F(y_i,\mathbf{x}) - F(y_k,\mathbf{x}) = [0;\ldots 0;\ +\mathbf{x};\ 0;\ldots 0;\ -\mathbf{x};\ 0;\ldots 0;]$

Loss functions for multiclass classification

0-1 loss:

$l(y, f(\mathbf{x})) = 1$ if $y \neq f(\mathbf{x})$

In general, we can use a cost-sensitive loss function (which we can define ourselves, depending on domain knowledge):

$l(y, y') > 0$ if $y \neq y'$

Generalized hinge loss

We return $h(\mathbf{x}) = \mathrm{argmax}_y \mathbf{w}F(y,\mathbf{x})$

For any class y (including the correct one):

$$\mathbf{w}F(y,\mathbf{x}) \leq \mathbf{w}F(h(\mathbf{x}),\mathbf{x})$$

This gives the generalized hinge loss for multiclass classification. For ($\mathbf{x}$,y) and weight vector **w:**

$$\mathrm{loss}(\mathbf{w}, (\mathbf{x},y)) = \max_{y'}(l(y,y') + \mathbf{w}(F(y',\mathbf{x}) - F(y,\mathbf{x}))$$

Interpretation: The score of the correct label y has to be greater than the score of any other label y' by at least l(y,y')

Multiclass SVM

Training data: $(\mathbf{x}^{(1)},y^{(1)})\ldots(\mathbf{x}^{(m)},y^{(m)})$

Parameters:

Regularization parameter $\lambda > 0$

Loss function $l: Y \times Y \rightarrow R_{+}$

Class-sensitive feature mapping F

Objective function: find $\mathbf{w}^*$

$$\mathbf{w}^* = \min_{\mathbf{w}} (\lambda \|\mathbf{w}\|^2 + \frac{1}{m} \sum_{i=1} \max_{y'} (l(y',y) + \mathbf{w}(F(y',\mathbf{x}^{(i)}) - F(y^{(i)},\mathbf{x}^{(i)}))$$

Today's key concepts

Multiclass classification

Relationship to linear classifiers

Class-sensitive features

Generalized hinge loss

Multiclass SVMs