Beyond Binary Classification
Handling more than two classes

- SVM formulation designed for multi-class problems:

\[
\begin{align*}
\text{minimize} & \; \frac{1}{2} \sum_{i=1}^{k} ||w_i||^2 + C \sum_{i=1}^{n} \sum_{m \neq y_i} \xi_i^m \\
\text{subject to:} & \; w_{y_i}^T x_i + b_{y_i} \geq w_m^T x_i + b_m + 1 - \xi_i^m, \; m \neq y_i, \\
& \; \xi_i \geq 0, \; i = 1, \ldots, n.
\end{align*}
\]

- There is a more efficient multi-class SVM formulation by Crammer and Singer (one slack variable per example).

Handling more than two classes

Alternative:
- Use a binary classifier to do multi-class classification.
- How to evaluate multi-class classifiers? (Some measures of classifier accuracy such as area under the ROC curve are specific to binary classifiers).
Multi-class classification with binary classifiers

You would like to perform multi-class classification, and assume you can only construct binary classifiers.

How to construct a multi-class classifier out of binary classifiers?

There multiple ways of doing this...
one-versus-one

Approach:
- Train $k(k-1)/2$ classifiers on each pair of classes
- Testing: classify example to the class that receives the largest number of votes

In scikit-learn:
There is a multiclass module with a class that implements this strategy (multiclass.OneVsOneClassifier)
one-versus-the-rest

Approach:
- Train $k$ binary classifiers, where classifier $i$ is trained to separate class $C_i$ from the rest
- In testing: assign a class label according to the classifier that provides the highest score to a test example

Simple and accurate!

See:
http://jmlr.org/papers/v5/rifkin04a.html

In scikit-learn:
multiclass.OneVsRestClassifier
If you feed an svm instance with multi-class data, it will automatically use this strategy.