Department of Computer Science
UNIVERSITY OF COLORADO BOULDER

Machine Learning: Yoshinari Fujinuma

University of Colorado Boulder

LECTURE 13

Slides adapted from Chenhao Tan, Jordan Boyd-Graber, Chris Ketelsen

Logistics

- HW1 grade will be released by Monday
- HW2 is due today
  - Please don't add new cells for the submitted version
- Next Monday will be the second hands on session using notebooks
- Next Wednesday will be in-class quizzes
- Final project team formulation due on March 1st

Logistics: In-class Quizzes

- Open notebook
- It will be available on Canvas
- Multiple choice questions + short answer questions
- Releasing sample questions on Monday
- This Zoom seesion will be open for Q&A

Logistics: In-class Quizzes

Topics include

- Decision Trees
- Bias-Variance trade-off
- k-NN
- Perceptron
- Feature Engineering
- Logistic Regression
- Naive Bayes
- Gradient Descent and Stochastic Gradient Descent

Logistics: Final Project

- Team formulation due date is March 1st
- 1 to 4 people per team
- Suggesting to form a team with 2+ people

Logistics: Final Project Expectations (Subject to Change)

Depends on what your project is, but typically

- Reading and preprocessing data
- Implementing baseline (method to compare against)
- Implementing what is proposed in the proposal
- Quantitative comparison between the methods
- Analysis
  - Ablation study of features
  - Error Analysis

We will have more detailed announcement when the proposal due date apporaches

Learning objectives

- Use binary classifiers for multi-class classifications
- A deep dive into regularization (bonus)

Classifiers

For classifiers that are basically binary

- Perceptron
- Logistic Regression

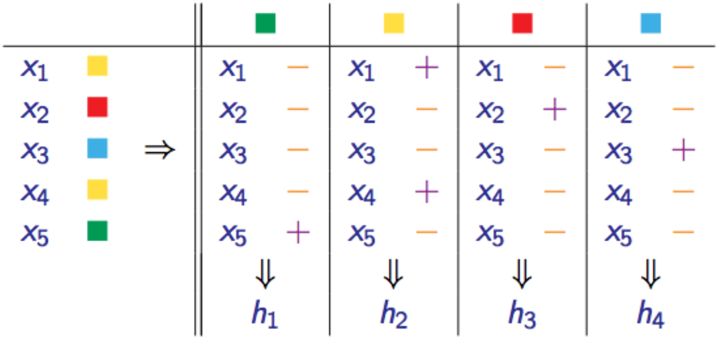
Is there anything that we can do?

Reduction

Two strategies

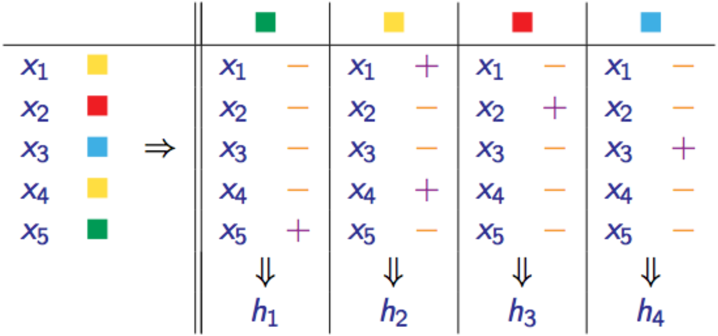
- One against all
- All pairs

One against all



- Colors represent separate 4 classes
- Break k-class problem into k binary problems and solve separately
- Evaluate with all h's, hope exactly one is $+$ (otherwise, take highest confidence)

One against all



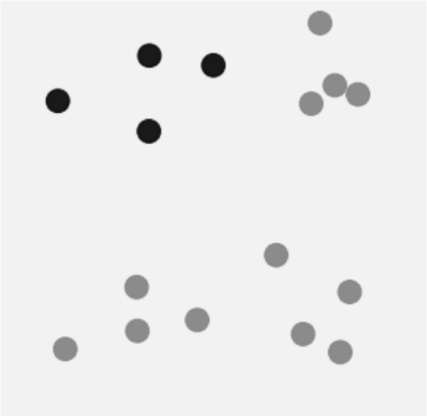
$$h(x) = \arg\max_{c \in C} h_c(\boldsymbol{x})$$

One against all

Build C binary classifiers of the form Class c vs Class $\neg c$

One against all

Build C binary classifiers of the form Class c vs Class $\neg c$
Black vs. not black



One against all

Build C binary classifiers of the form Class c vs Class $\neg c$

Red vs. not red

One against all

Build C binary classifiers of the form Class c vs Class $\neg c$
Yellow vs. not yellow

One against all

Build C binary classifiers of the form Class c vs Class $\neg c$

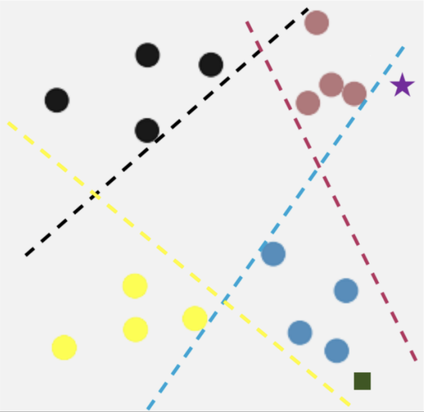
Blue vs. not blue

One against all

Build C binary classifiers of the form Class c vs Class $\neg c$

Predict class with highest confidence

- Predict green square
- Predict purple star



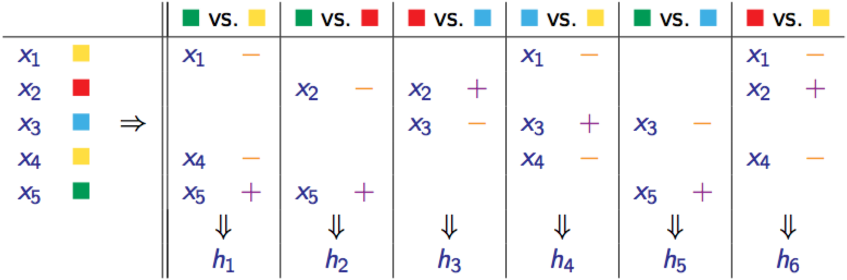
One against all

Can you see any pitfalls of the one-against-all method?

One against all

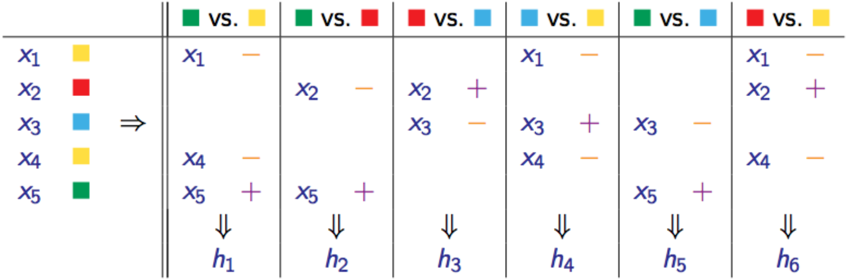
Can you see any pitfalls of the one-against-all method?
A big one is that if you start with a balanced training data, you immediately create imbalanced data.

All pairs



- Break k-class problem into $k(k-1)/2$ binary problems and solve separately
- Combine predictions: evaluate all h's, take the one with highest sum confidence

All pairs



$$h(x) = \arg\max_{c \in C} \sum_{c' \neq c} h_{c'c}(\boldsymbol{x})$$

Time Comparison

- One-against-all: Train/Test $O(k)$ classifiers, each classifier trained on **all** examples
- All-pairs: Train/Test $O(k^2)$ classifiers, each classifier trained on **subset** of examples

- One-against-all better for testing time
- All-pairs better for training
- All-pairs usually better for performance

Outline

Regularization (bonus)

Ridge vs. Lasso

Ridge Regression or ℓ_2-Regularization:

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2 + \lambda \sum_{k=1}^{D} w_k^2$$

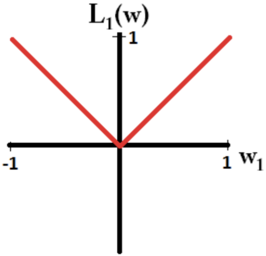
Lasso Regression or ℓ_1-Regularization:

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2 + \lambda \sum_{k=1}^{D} |w_k|$$

Different penalty terms lead to different character of models

L1 vs. L2

Coefficients shrink to zero faster in L1



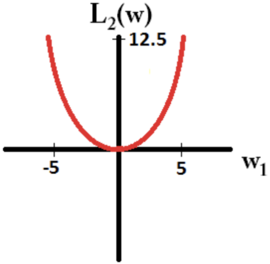


Image from `https://www.kaggle.com/amrmahmoud123/advanced-regularization`

The constrained optimization explanation

Consider the minimizer of

$$\min_{\mathbf{w}} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2 + \lambda \sum_{k=1}^{D} w_k^2 \quad \text{or} \quad \min_{\mathbf{w}} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2 + \lambda \sum_{k=1}^{D} |w_k|$$

For each objective function, can show that for a given λ there is an equivalent s such that the usual solution also solves

$$\text{Ridge:} \quad \min_{\mathbf{w}} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2 \quad \text{s.t.} \quad \sum_{k=1}^{D} w_k^2 \leq s$$

$$\text{Lasso:} \quad \min_{\mathbf{w}} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2 \quad \text{s.t.} \quad \sum_{k=1}^{D} |w_k| \leq s$$

The constrained optimization explanation

Think of the constraint as a budget on the size of the parameters
For a given budget s (corresponding to a given λ), find the $\mathbf{w}$ that minimizes the loss while staying inside the constrained region
Lasso Region for Two Features: Diamond

$$|w_1| + |w_2| \leq s$$

Ridge Region for Two Features: Circle

$$w_1^2 + w^2 \leq s$$

The constrained optimization explanation

Minimum is more likely to be at point of diamond with Lasso, causing some feature weights to be set to zero.