ONLINE MASTERS IN **DATA SCIENCE**

DSC 255 - MACHINE LEARNING FUNDAMENTALS

# BOOSTING WEAK LEARNERS

SANJOY DASGUPTA, PROFESSOR

UC San Diego

COMPUTER SCIENCE & ENGINEERING

HALICIOĞLU DATA SCIENCE INSTITUTE

# Choosing A Classifier

So many choices:

- Nearest neighbor
- Different generative models
- Linear predictors with different loss functions
- Different kernels
- Neural nets
- etc.

Can one **combine** them?
And get a classifier that is better than any of them individually?

# Combining Simple Classifiers

1. No one classifier is going to be the final product.
   So why not keep the individual components simple?

# Combining Simple Classifiers

1. No one classifier is going to be the final product.
   So why not keep the individual components simple?

2. How to train each constituent classifier?
   On the full training set?

# Combining Simple Classifiers

1. No one classifier is going to be the final product.
   So why not keep the individual components simple?

2. How to train each constituent classifier?
   On the full training set?

3. The full (combined) models may get enormous.
   Is this bad for generalization?

# Weak Learners

It is often easy to come up with a **weak classifier**, one that is marginally better than random guessing:

$$\Pr(h(X) \neq Y) \leq \frac{1}{2} - \epsilon$$

A learning algorithm that can consistently generate such classifiers is called a **weak learner**.

Is it possible to systematically boost the quality of a weak learner??

# The Blueprint For Boosting

Given: data set $\left(x^{(1)}, y^{(1)}\right), \ldots, \left(x^{(n)}, y^{(n)}\right)$.

- Initially give all points equal weight.
- Repeat for $t = 1, 2, \ldots$ :
  - Feed weighted data set to the weak learner, get back a weak classifier $h_t$
  - Reweight data to put more emphasis on points that $h_t$ gets wrong
- Combine all these $h_t$'s linearly