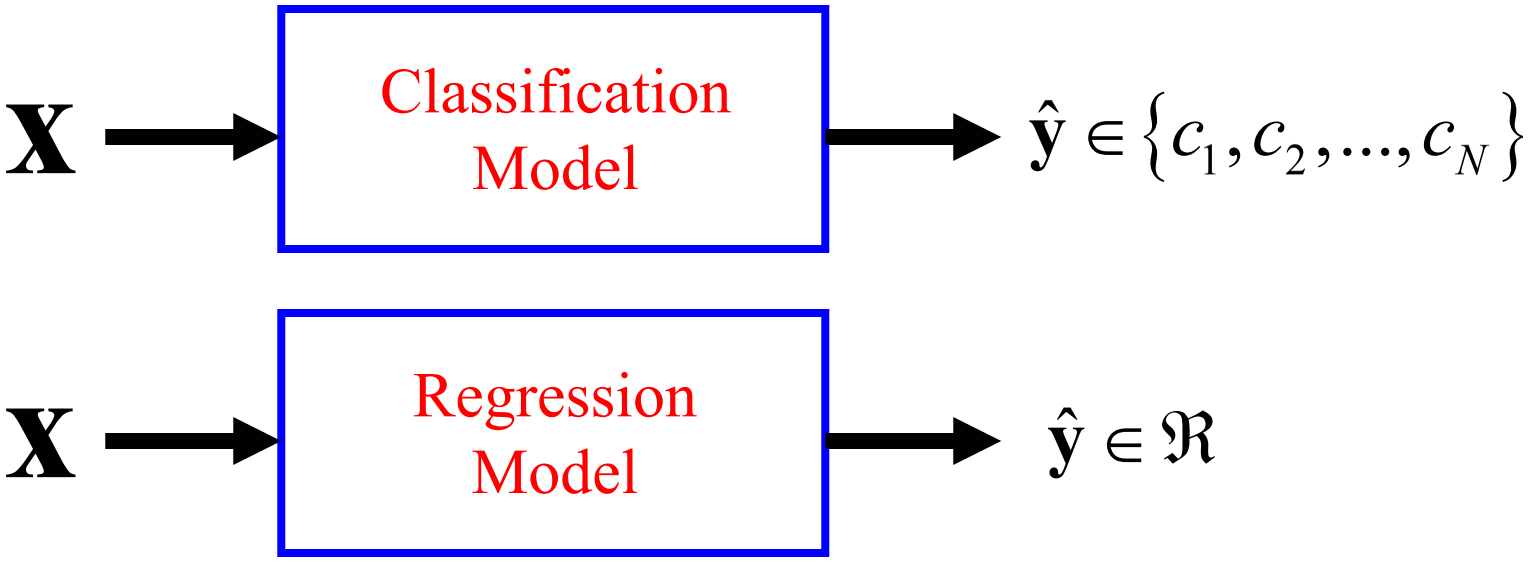
# Class Project: Artificial Intelligence 2: Machine Learning CSCI 4202

Greg Grudic

# Most Supervised Learning Algorithms Build Models That Just Predict Outputs

Given data: $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$

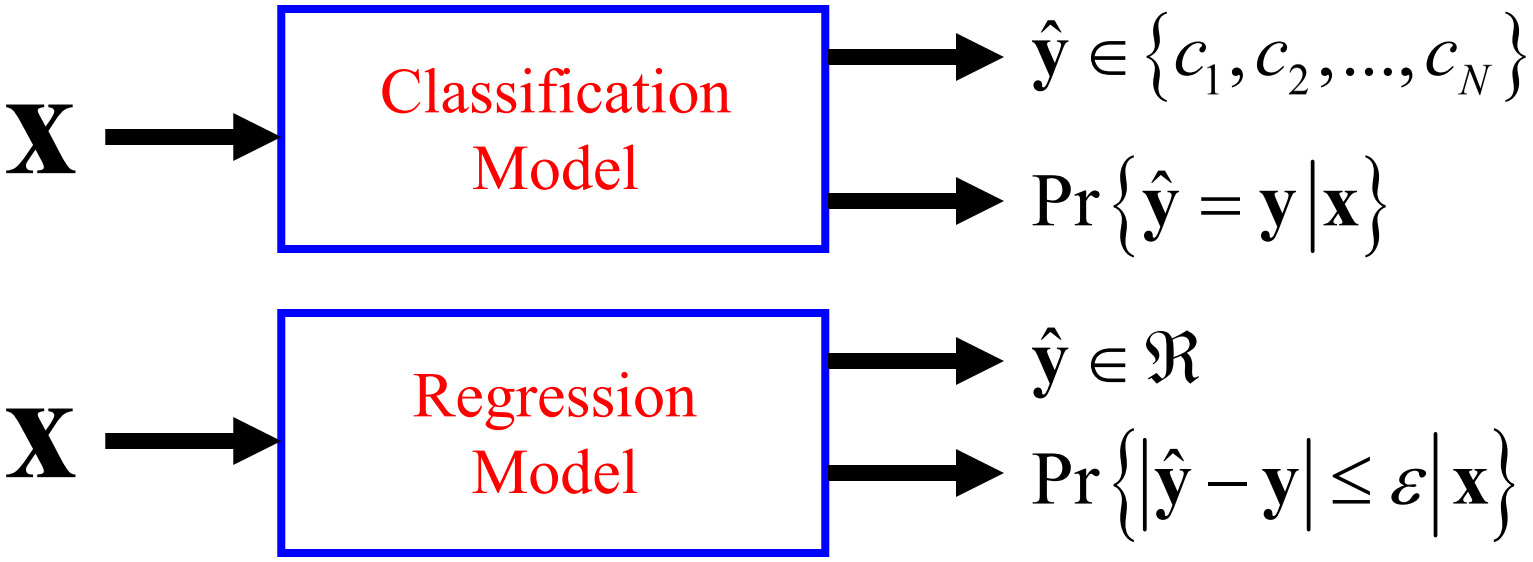
A model is computed:



# *Some* Supervised Learning Algorithms Build Models That Also Predict The Probability of Correct Outputs

Given data: $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$

A model is computed:



# Open Research Question

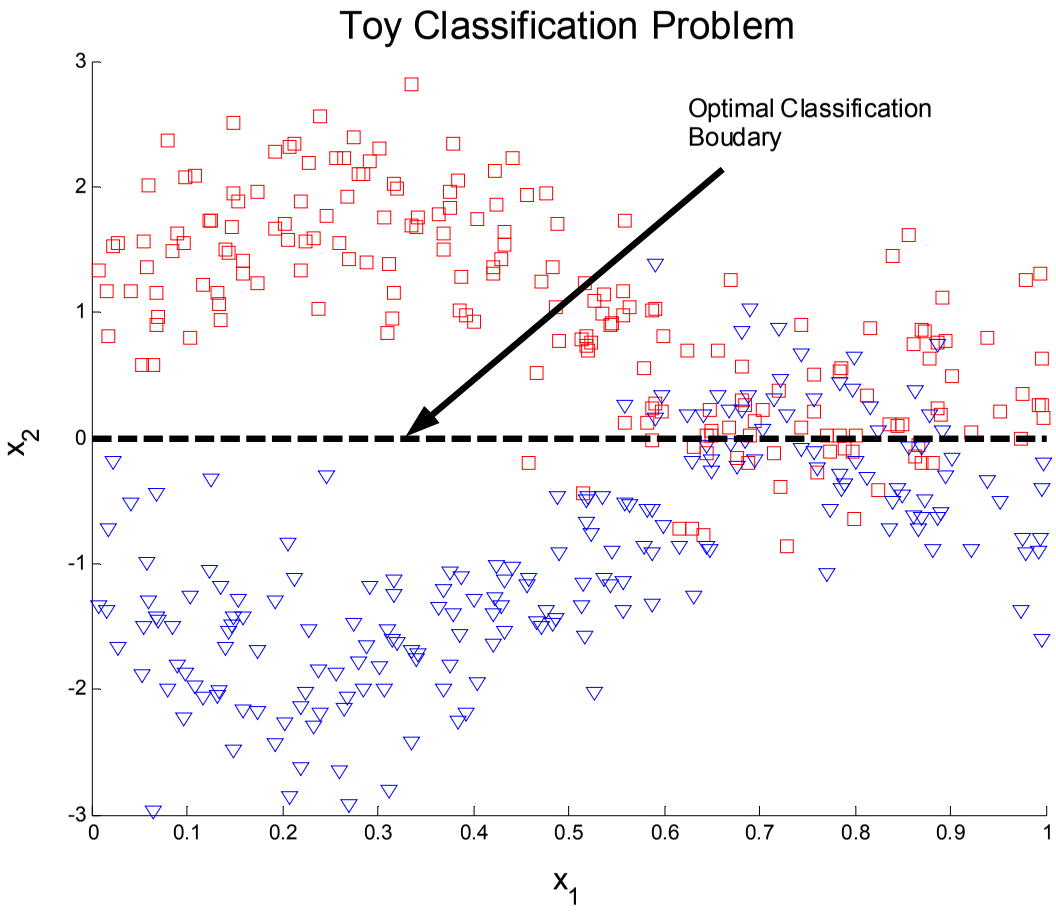
- *No one has EVER evaluated how good these algorithms are!*

# Project Goals

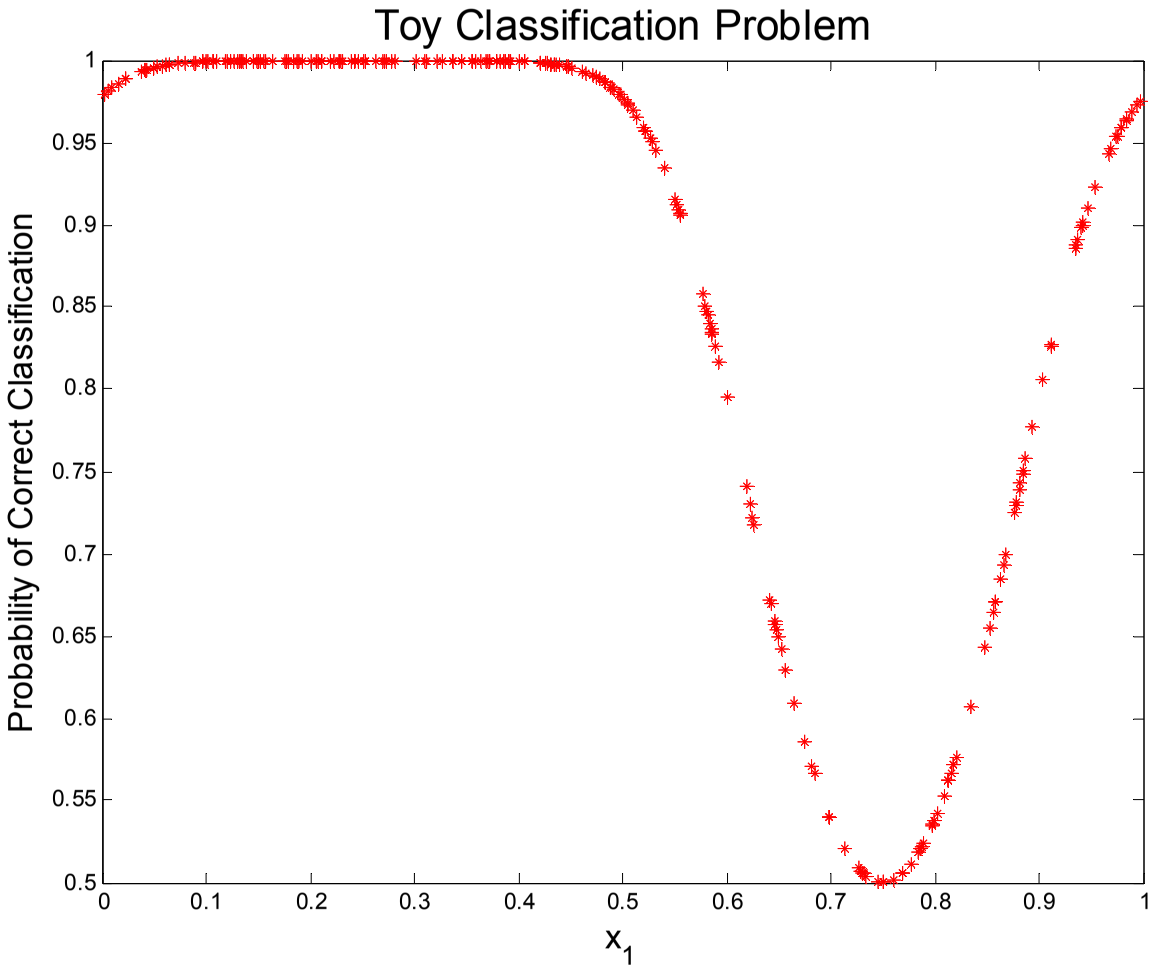
- To evaluate existing CLASSIFICATION algorithms that give Point Specific Estimates of the probability that the model output is correct

- Submit a conference paper that summarizes our findings
  - Everyone will be an author!

# Classification Example: The Probability is Not the Same Everywhere!



# True Probability of Correct Classification



# Project Objectives

1) Evaluate Platt's Probabilistic SVM
   a. By Jan 29 – choose an SVM program you want to use
   b. By Feb 5 – Implement Platt's probabilistic SVM algorithm and test on Toy Data
   c. By Feb 12 – verify the results reported by Platt
2) Evaluate a second algorithm
   a. By Feb 19 – identify possible algorithms and choose one
   b. By March 19 – Implement and evaluate on data
3) Evaluate a third algorithm by April 19
4) Write up results in conference format by April 31

# Admin Stuff

- You can work by yourself, or in groups of 2 or 3
- Grading:
  - Homework 0%
  - Project 60%
  - Class participation 10%
  - Final exam 30%

# SVM Model Structure

$$f(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i K(\mathbf{x}_i, \mathbf{x}) + b$$

$$\hat{y} = \text{sgn}\left(f(\mathbf{x})\right)$$

$$\text{sgn}\left(f(\mathbf{x})\right) = \begin{cases} 1 & \text{if } f(\mathbf{x}) > 0 \\ 0 & \text{if } f(\mathbf{x}) = 0 \\ -1 & \text{if } f(\mathbf{x}) < 0 \end{cases}$$

# Making SVM Probabilistic

- Probability of Class given an input is modeled as:

$$\Pr\left(y=1\middle|\mathbf{x}\right)=\frac{1}{1+\exp\left(Af\left(\mathbf{x}\right)+B\right)}$$

$$A,B\in\Re$$

# Platt's Algorithm

- Finds $A, B \in \Re$ from the training data

$$\min_{z=(A,B)} F(z)$$

$N_+$ Number of +1 examples

$N_-$ Number of -1 examples

$l$ Total number examples

$$F(z) = -\sum_{i=1}^{l} \left( t_i \log(p_i) + (1 - t_i) \log(1 - p_i) \right),$$

$$p_i = \frac{1}{1 + \exp(Af_i + B)}, f_i = f(x_i), \text{ and } t_i = \begin{cases} \frac{N_+ + 1}{N_+ + 2} & \text{if } y_i = 1 \\ \frac{1}{N_- + 2} & \text{if } y_i = -1 \end{cases}, i = 1, 2, \ldots, l$$

# Toy Data

- The course web page has
  - Learning Data called *Toy_Data_Lrn.txt*
  - Test Data called *Toy_Data_Tst.txt*
    - Each test data point has an associated true probability of error contained in the file *p_correct_tst.txt*

- There are 500 training and 500 testing examples

# Toy Data Format (text files)

(Toy_Data_Lrn.txt and Toy_Data_Tst.txt )

- Each row contains one example formatted as: $x_1 \;\; x_2 \;\; y$

0.495327 0.256688 +1
0.210797 1.00159 +1
0.735431 0.592157 +1
0.0972693 1.87381 +1
0.454368 1.24619 +1
0.724791 0.636121 +1
0.606225 -0.08338 -1
0.744476 0.247182 +1
0.183687 1.59675 +1
0.805341 -0.435892 -1
0.762054 -0.484573 -1

# The True Probabilities For the Test Data

- Contained in the file *p_correct_tst.txt*
- Each row in the file contains a true probability of correctly classifying the corresponding test point in *Toy_Data_Tst.txt*

# Next Thursday

- I will email each of you 5 training sets, and 5 test sets from this toy data.
- By Friday you will:
  - Build 5 models
  - Email me 5 text files where each row contains the predicted probability for each $x^{(1)}$ test example (of the corresponding test set).
  - Email me your Code
    - Including all info on how to run it