Lecture 17: Machine Learning Review

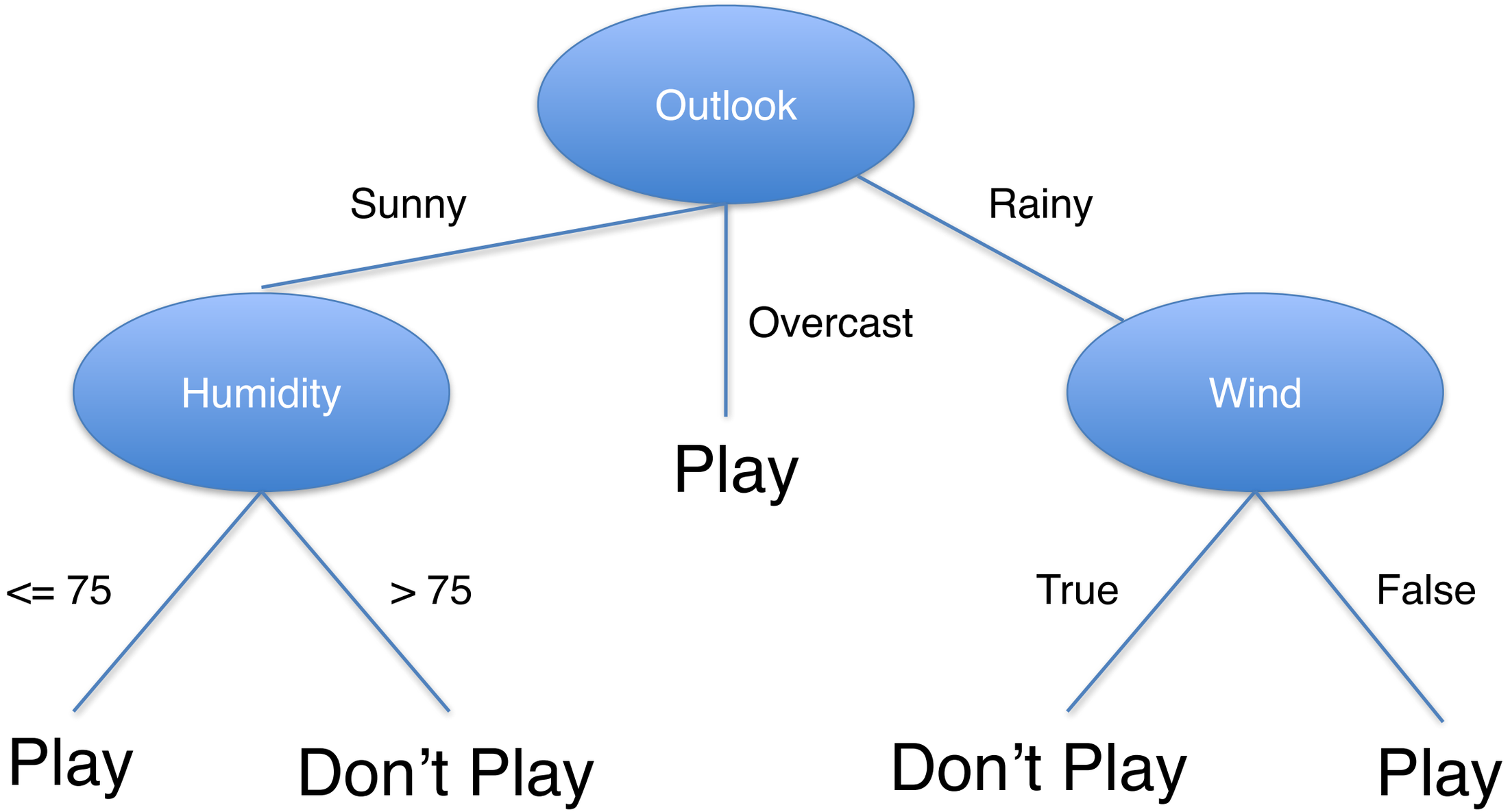
Core 109S IDWT?, Spring 2017

Michael Hay

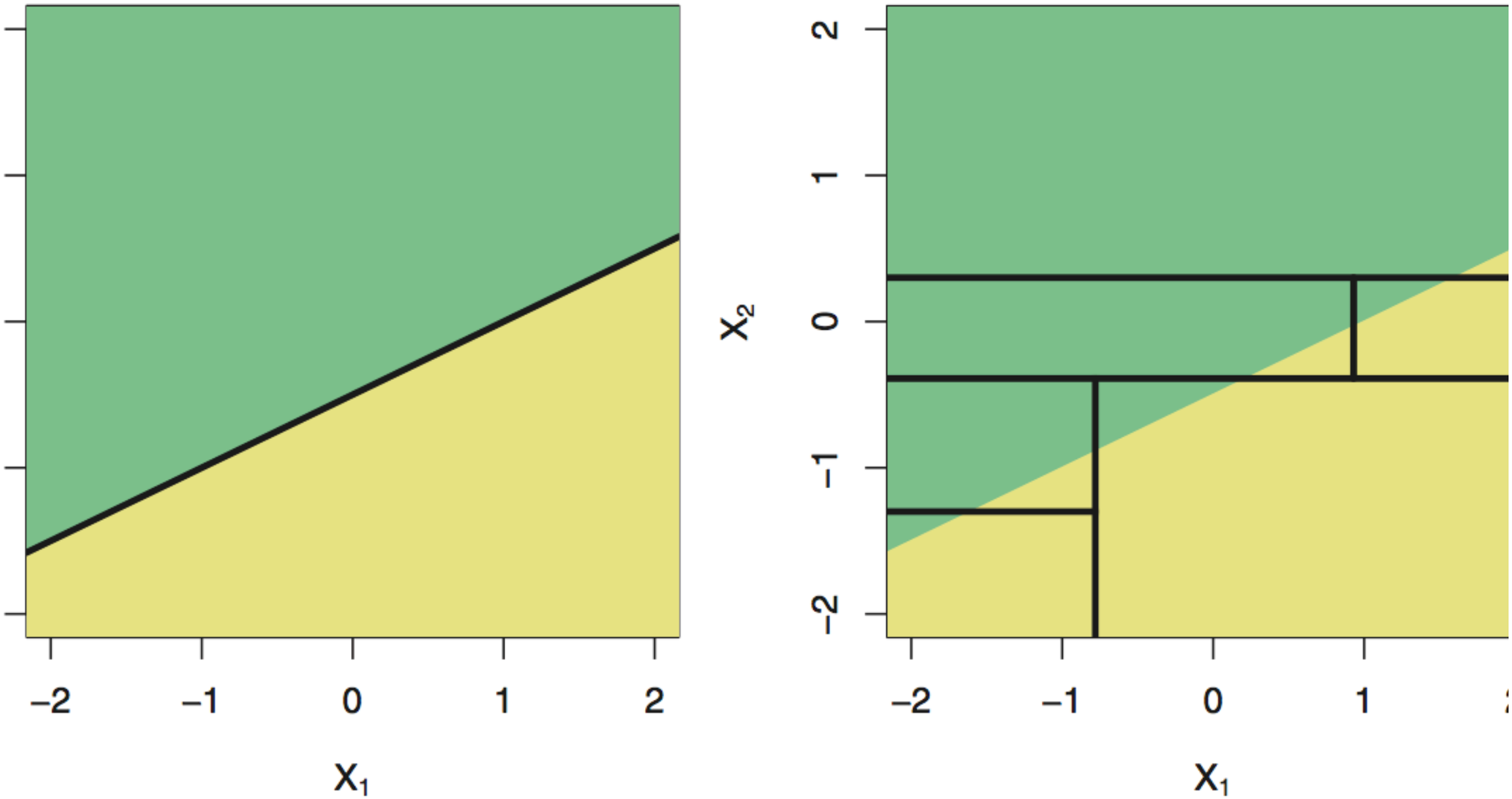
Today

- Review what we've learned so far about
  - Decision trees
  - Perceptrons (linear discriminants)
- Review is oriented through *exercises*
  - Work in groups, take your own notes, ask me questions as we go.
  - We will review answer to each question
- Wednesday: look at one more subtle but important issue with machine learning (overfitting)

“Is it a good day to play golf?”



Linear vs. trees



True boundary is linear; tree can only approximate with axis-parallel splits

Credit:James et al. http://www-bcf.usc.edu/~gareth/ISL/

Complete Q1 on Worksheet

Example: perceptron

- Given input $x = (x_1, \ldots, x_d)$ *(the d attributes of customer app.)*

- Assign a **weight** w_i for each attribute value x_i. and choose some **threshold** value

$$\text{Approve credit if} \quad \sum_{i=1}^{d} w_i x_i \geq \text{threshold}$$

$$\text{Deny credit if} \quad \sum_{i=1}^{d} w_i x_i < \text{threshold}$$

Credit: Yaser Abu Mostafa https://work.caltech.edu/

Writing as function *h(x)*

$$h(x) = \text{sign}\left(\left(\sum_{i=1}^{d} w_i x_i\right) - \text{threshold}\right)$$

$$= \text{sign}\left(\left(\sum_{i=1}^{d} w_i x_i\right) + w_0\right)$$

$$= \text{sign}\left(\left(\sum_{i=1}^{d} w_i x_i\right) + w_0 x_0\right)$$

$$= \text{sign}\left(\sum_{i=0}^{d} w_i x_i\right)$$

sign(x) = 1 if x ≥ 0
-1 if x < 0

(rename "threshold" as w_0)

(introduce artificial attribute x_0=1)

Complete Q2 on Worksheet

Algorithm 1 Algorithm for learning weights of a perceptron

procedure PERCEPTRONLEARNER(training data)
    Training data: collection of (x, y) pairs where $x = (x_1, \ldots, x_d)$ and $y = +1$ or $y = -1$.
    Initialize $w_0 = w_1 = \cdots = w_d = 0$
    repeat
        ▷ Pick a misclassified example:
        Find some (x, y) pair such that $h(x) = \text{sign}\left(\sum_{i=0}^{d} w_i x_i\right) \neq y$
        ▷ Update the weight vector:
        for $i = 1$ to d **do**
            Set $w_i \leftarrow w_i + y \times x_i$
        end for
    until There are no more misclassified points (or some other stopping criterion)
    Return weights $w_0, \ldots, w_d$.
end procedure

Intuition:

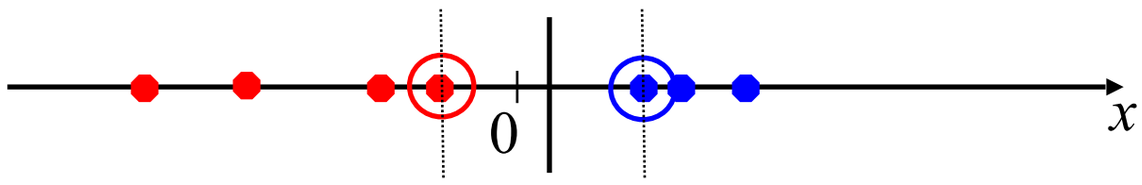
- Suppose Bob makes $15K in salary
- Perceptron incorrectly denies Bob credit
    - h(Bob) = -1 but y = 1
- Weight update will *increase* the weight on salary, and therefore *increase* h(Bob)

Complete Q3 on Worksheet

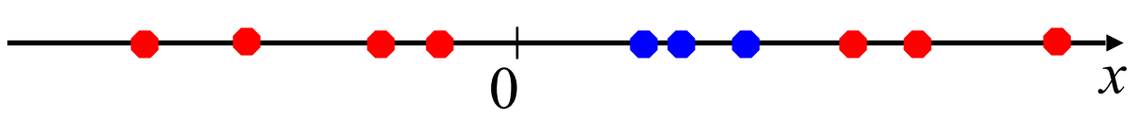
Complete Q4 on Worksheet

Non-Linear Separators

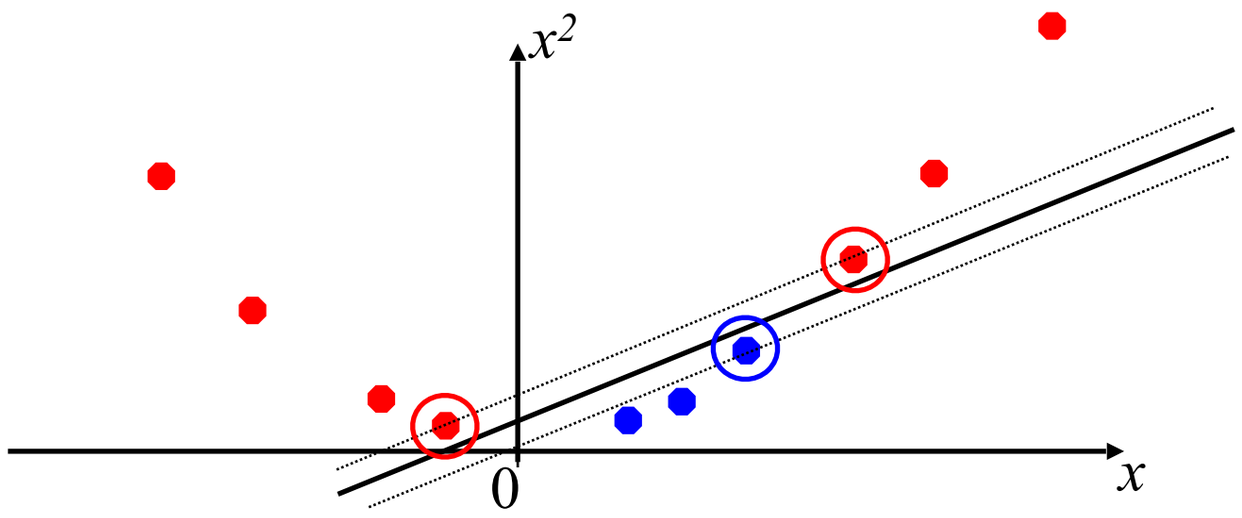
- Data that is linearly separable (with some noise) works out great:



- But what are we going to do if the dataset is just too hard?



- How about… mapping data to a higher-dimensional space:



This slide adapted from Ray Mooney, UTexas

Feature engineering

- A perceptron (aka linear discriminant) finds a line in feature space that separates the data into two groups.

- But we are free to *add* new features!

- While data may not be linearly separable in "original" feature space, it may be separable in augmented feature space.

- In general, feature engineering is a key component of ML.

Example #2: Spam Filter

- Input: email
- Predict: "spam" or "ham"
- Data: old emails, each labeled “spam” or “ham”

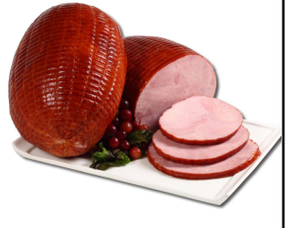
Dear Sir.

First, I must solicit your confidence in this transaction, this is by virture of its nature as being utterly confidencial and top secret. …

- **Features**: The attributes used to make the ham / spam decision
  - Words: FREE!
  - Text Patterns: $dd, CAPS, spelling errors, …
  - Non-text: SenderInContacts
  - …

TO BE REMOVED FROM FUTURE MAILINGS, SIMPLY REPLY TO THIS MESSAGE AND PUT "REMOVE" IN THE SUBJECT.

99 MILLION EMAIL ADDRESSES FOR ONLY $99



Ok, I know this is blatantly OT but I'm beginning to go insane. Had an old Dell Dimension XPS sitting in the corner and decided to put it to use, I know it was working pre being stuck in the corner, but when I plugged it in, hit the power nothing happened.

Example #3: Digit Recognition

- Input: images / pixel grids
- Predict: which digit 0-9
- Data: images of hand-written digits, each labeled

0

1

- **Features**: The attributes used to make the digit decision
  - Pixels: (6,8)=ON
  - Shape Patterns: NumComponents, AspectRatio, NumLoops, Vertical/ Horizontal Symmetry
  - …

2

1

??

Complete Q5 and Q6 on Worksheet