Machine Learning with Perceptrons

Machine Learning

Using statistical methods to make computers “learn” to do tasks when given example data. The methods improve the more example data is given.

Some Vocab

Classification

- Taking unknown input and assigning it to one or more known classes of data
- What ML is trying to do most of the time

Feature

- A subset of a data point with all the "good" data

Training

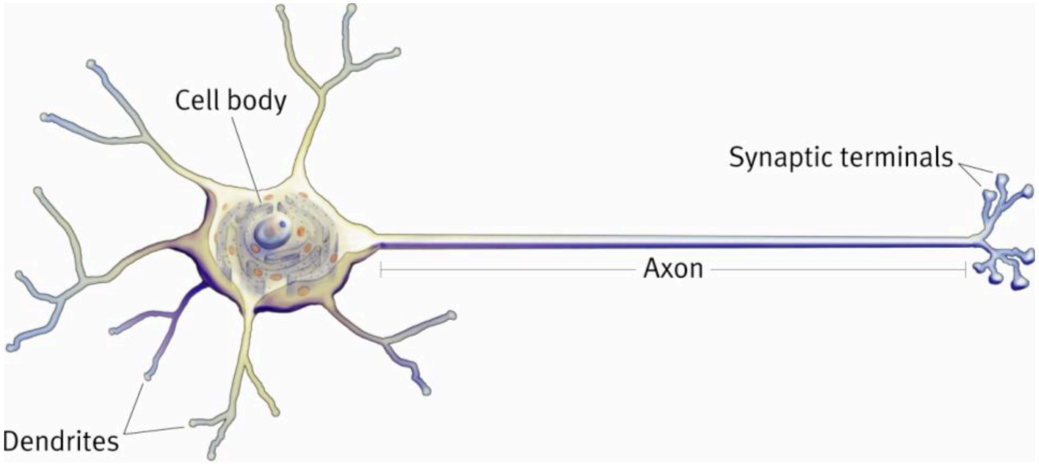
- Giving the ML algorithm example data to improve the classifier

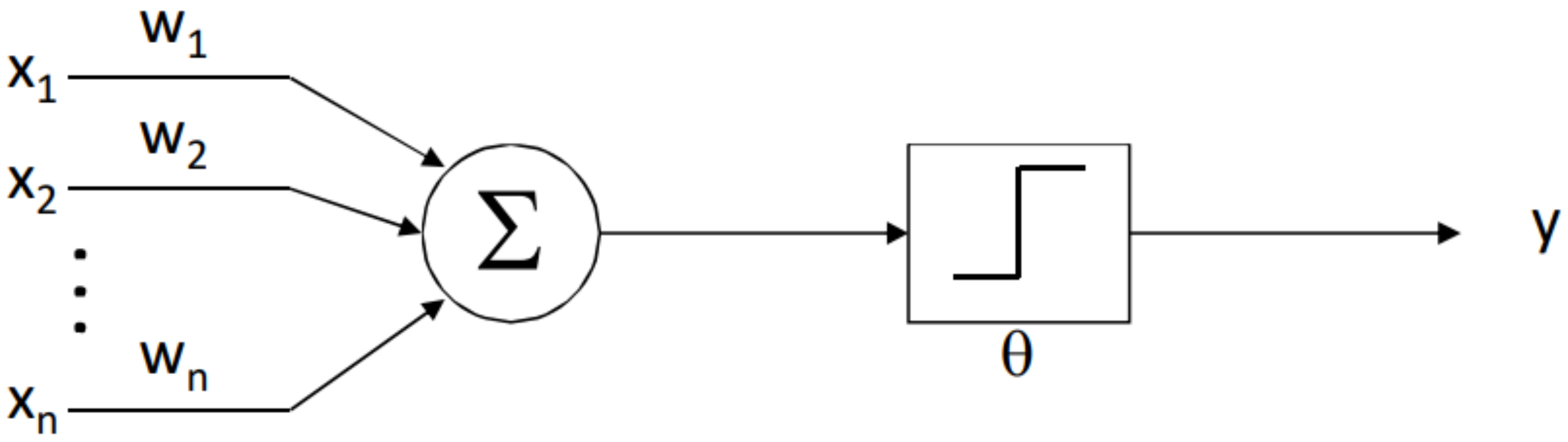
Testing

- Using different data from training to assess the classifier

Perceptrons

- Attempt to mimic biological neurons
- Multiple inputs summed together -> single binary output
- Mathematically, actually pretty simple
- Can be stacked to increase complexity
  - Leads to things like *neural* networks and deep learning





thresholding ***output***

weights

inputs ***summation***

$$y \;=\; 1 \;\; \text{if} \;\; \sum w_i\, x_i \geq \theta;$$
$$0, \;\; \text{otherwise.}$$

+

Letter Example

Weight Vector

1	1	1	1
1	-1	-1	1
1	1	1	1
1	-1	-1	1

$\Theta = 11.5$

Test Data 1

1	1	1	1
1	0	0	1
1	1	1	1
1	0	0	1

Test Data 2

1	0	0	1
1	1	1	1
1	0	0	1
1	0	0	1

What will this perceptron output for each of these two inputs?

Training the Perceptron

- Get a whole bunch of negative and positive examples **X**
- Create arbitrary initial classifier
  - $w_0 = (0,0,0,\ldots)$
  - $\Theta_0 = 1$
- Create constant to determine how “important” each new data point is
  - c = 1 (to keep it simple)

Training Algorithm

Classify X_k with w_k and Θ_k

```
Was it correct?
        Move on to the next one
If not
        Was it a false negative?
                $w_{k+1} = w_k + c * X_k$    and    $\Theta_{k+1} = \Theta_k – c$
        Was it a false positive?
                $w_{k+1} = w_k - c * X_k$    and    $\Theta_{k+1} = \Theta_k + c$
```

Repeat the above until the classifier stops being wrong on any training data

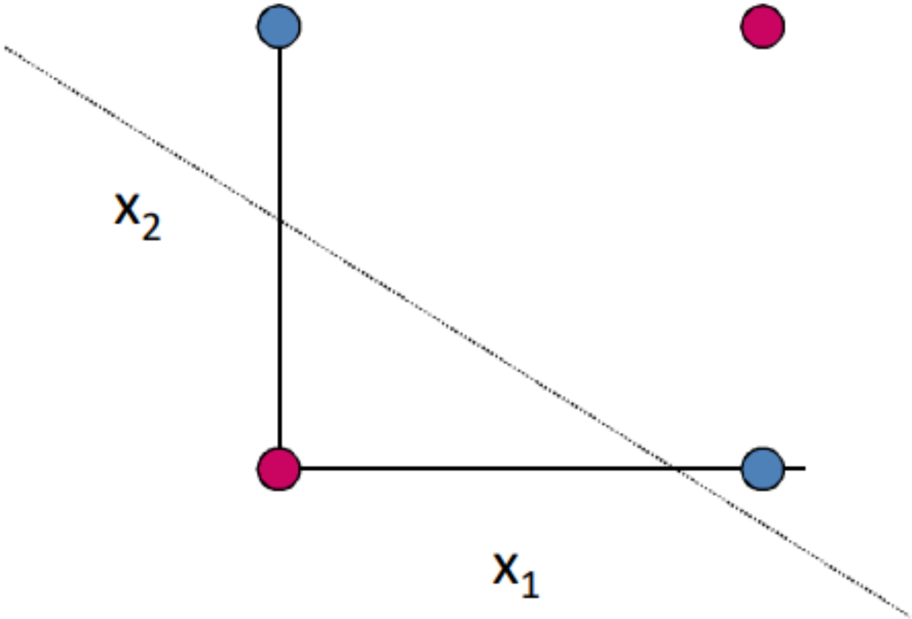
+

Pseudocode

```
while (not converged)
    for (xk in trainingData)
        output = xk * wk
        if (output is false-negative)
            wk = wk + c*xk
            threshold = threshold – c
        else if (output is false-positive)
            wk = wk – c*xk
            threshold = threshold + c
```

Convergence Issues

- Classifier won't always converge
- Linear Separability

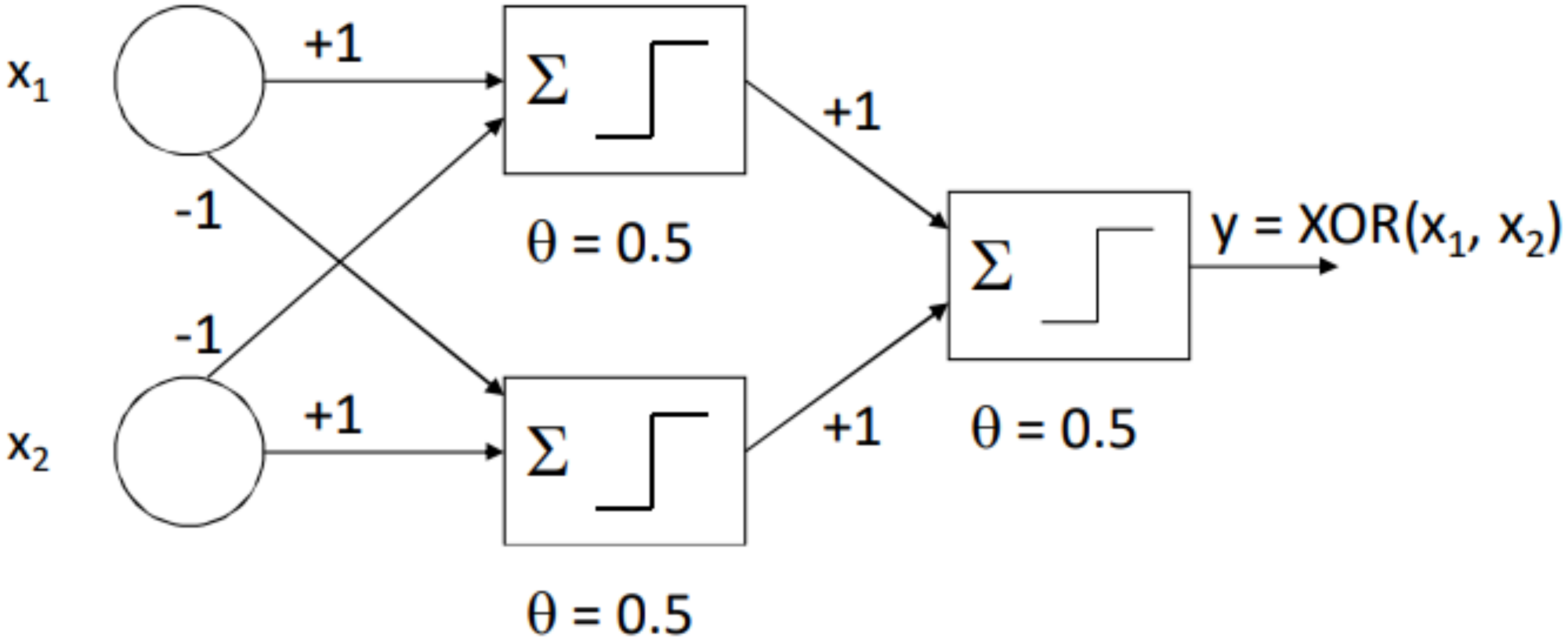


$X^+ = \{ (0, 1), (1, 0) \}$

$X^- = \{ (0, 0), (1, 1) \}$

$X = X^+ \cup X^-$

Solution: Add More Perceptrons!



Let's Make and Train a Writing Classifier

- Get a (REAL) dataset of drawn numbers
  - https://archive.ics.uci.edu/ml/datasets/Optical+Recognition+of+Handwritten+Digits
  - Get the "optdigits.tra" file
- Get the skeleton code from the class website
- We're writing this in R
  - But I've done most of the syntax work for you

Very Brief R Intro

- Statistical programming language
- Good for math and creating data visualizations
- Free!
- Can program in a console like Unix
  - Or write scripts like most programming languages
- Assignment operation looks like "variable <- value"