INFO5002: Intro to Python for Info Sys

Week 14

Slides created by: Zachary Doucet

LVX
VERITAS
VIRTVS

Northeastern University

Week 14

I. Perceptron

II. XOR

III. MLP

Review

Ridge Regression

$$ridge = \sum \beta_i^2$$

$$\nabla_\beta ridge = \begin{pmatrix} \beta_0 \\ 2\alpha\beta_1 \\ \dots \\ 2\alpha\beta_d \end{pmatrix}$$

Ridge coefficient

Lasso Regression

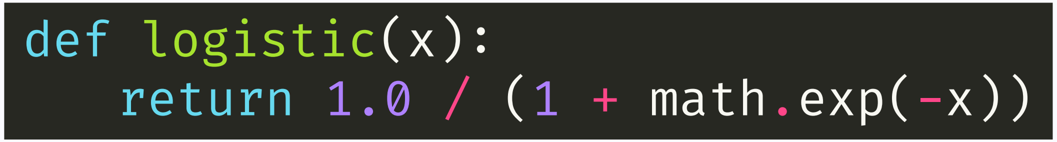
$$lasso = \sum |\beta_i|$$

This derivative is a bit difficult.

Essentially each feature is +1, -1, or [-1, 1]

Logistic Function

$$logistic(x) = \frac{1}{1+e^{-x}}$$



```python
def logistic(x):
    return 1.0 / (1 + math.exp(-x))
```

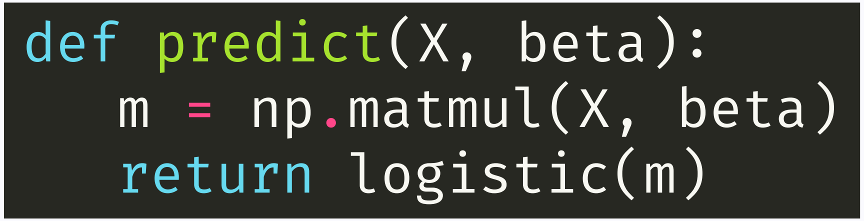
- As x increases, e^-x gets smaller => closer to 1.

- As x decreases, e^-x gets bigger => closer to 0.

Logistic Regression

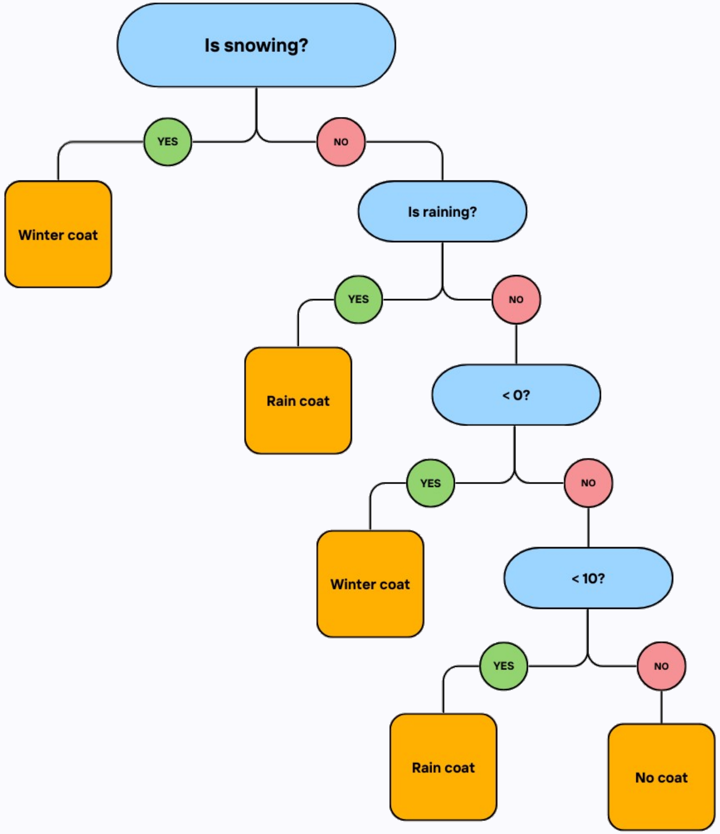
$$\hat{y} = f(X\beta)$$

Where f is the logistic function



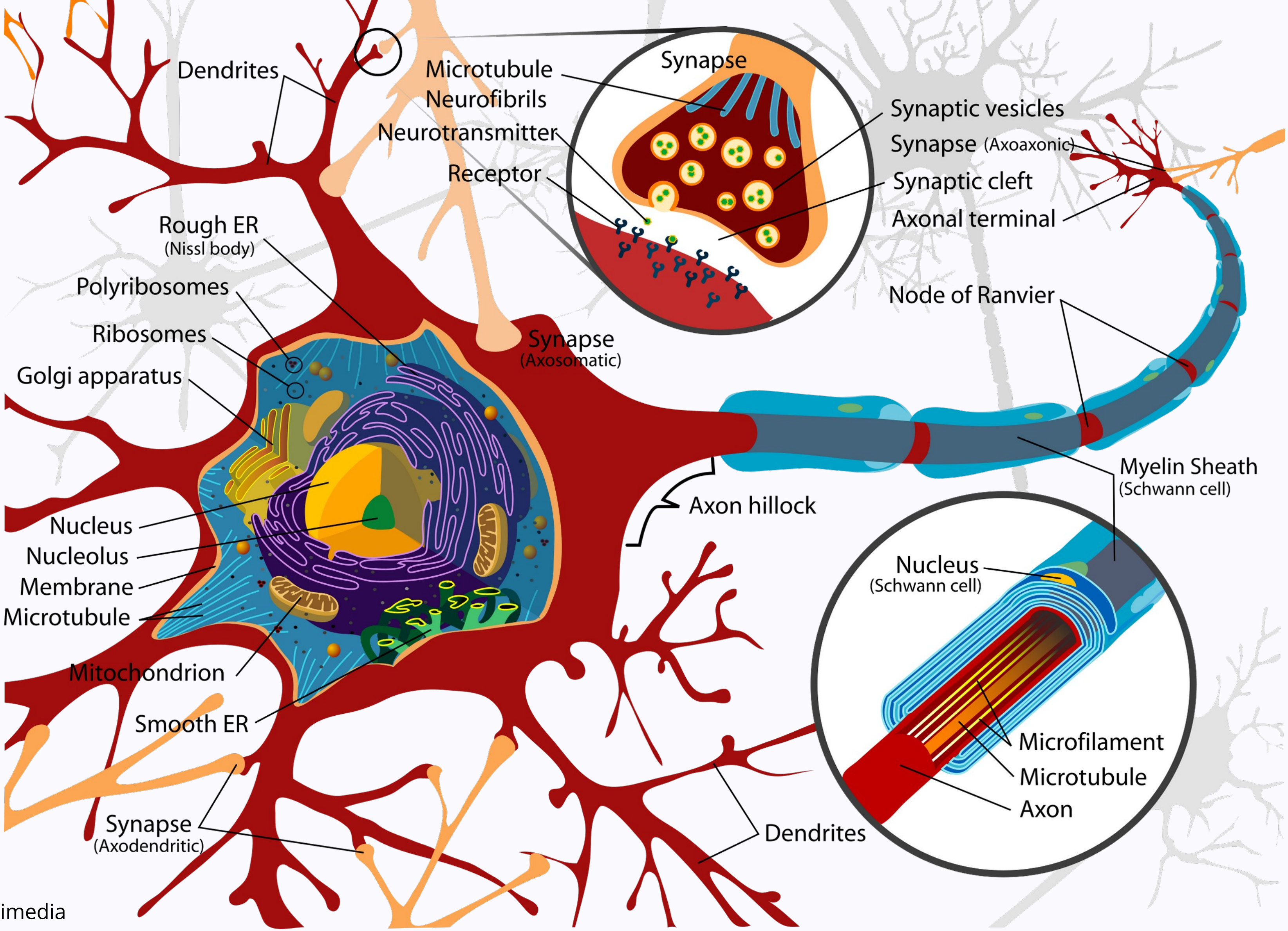
```python
def predict(X, beta):
   m = np.matmul(X, beta)
   return logistic(m)
```

Decision Trees



Perceptron

DSfS 227-229



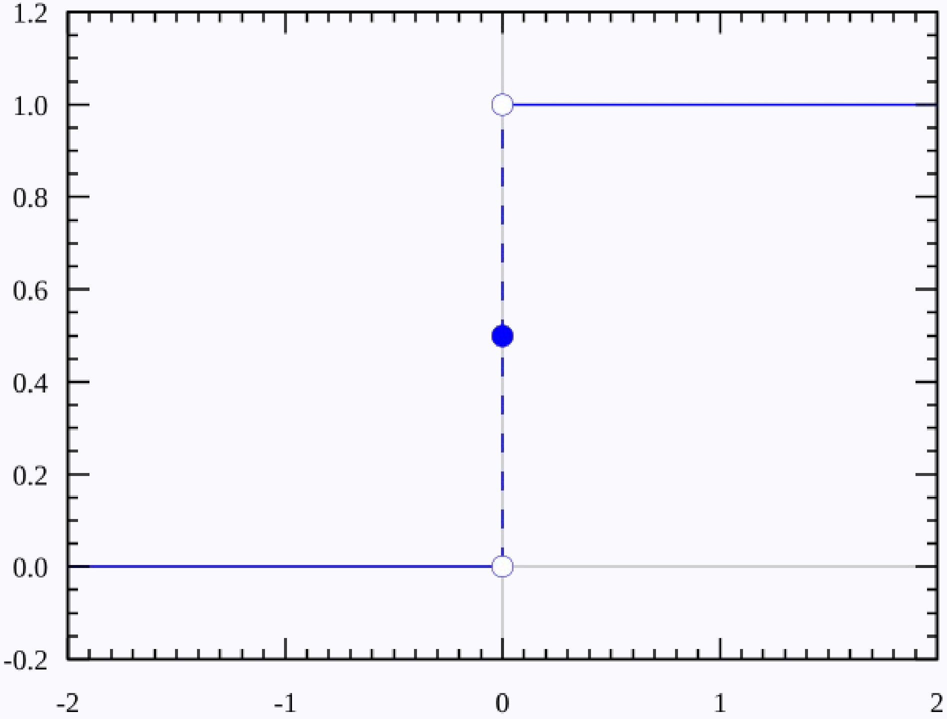
Source: Wikimedia

Perceptron

- Aims to simulate a single neuron.
  - Invented by Warren McCulloch & Walter Pitts
- Frank Rosenblatt simulated the first perceptron on an IBM 704.
- Can think of a logistic function but instead of logistic, you use a step function.

Step Function

- All or nothing function.



>= 0 then 1
< 0 then 0

Called the Heaviside function

Source: Wikimedia

The perceptron

$$f(x) = h(w \cdot x + b)$$

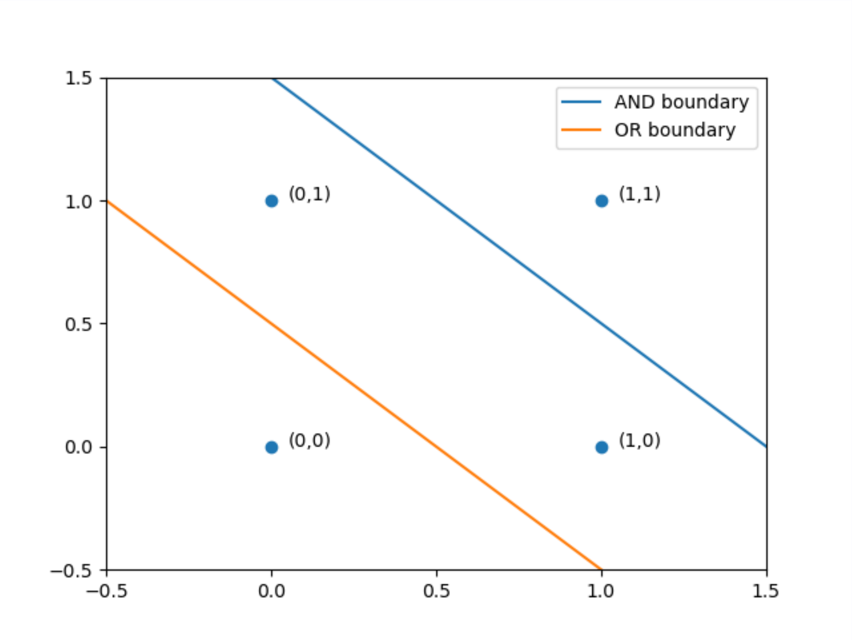
You can pass the bias in like we did w/ MLR

$$f(x) = h(w' \cdot x')$$

$$h(y) = \begin{cases} 0, if\ y < 0 \\ 1, if\ y \geq 0 \end{cases}$$

We can solve many problems

- As long as they are linearly separable.
- You can represent an AND gate, an OR gate, a NOT gate.

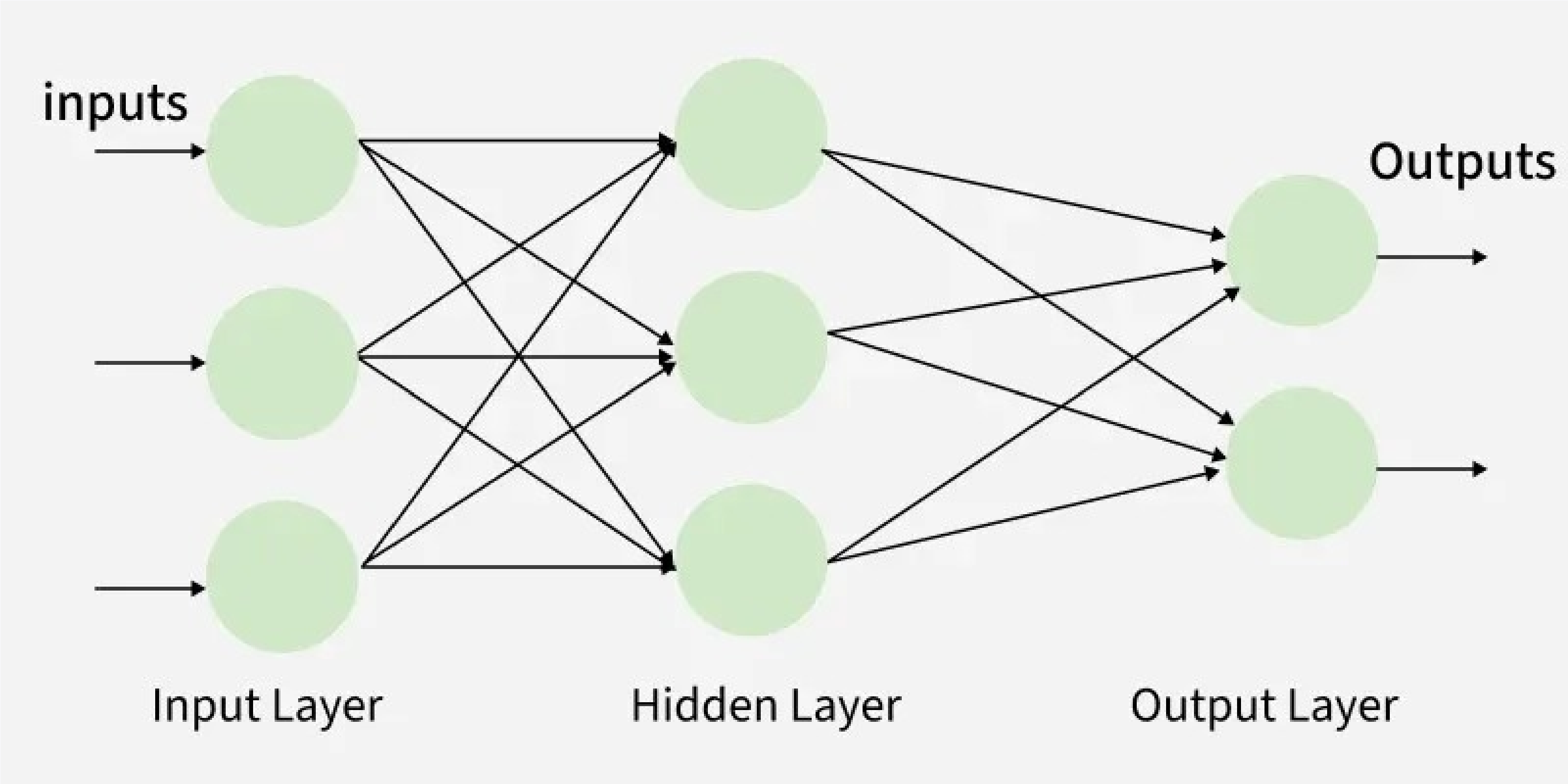


But can we solve XOR?

MLP

DSfS 227-229

We can solve by stacking



Source: GeekForGeek

- By combining perceptrons we can create an MLP

Combine

- The perceptron of one can be fed into the next one
- To allow for learning (which requires derivatives) we need to use a continuous function.
    - We can mimic heaviside function w/ sigmoid function
- This simple idea is the foundation of modern ML