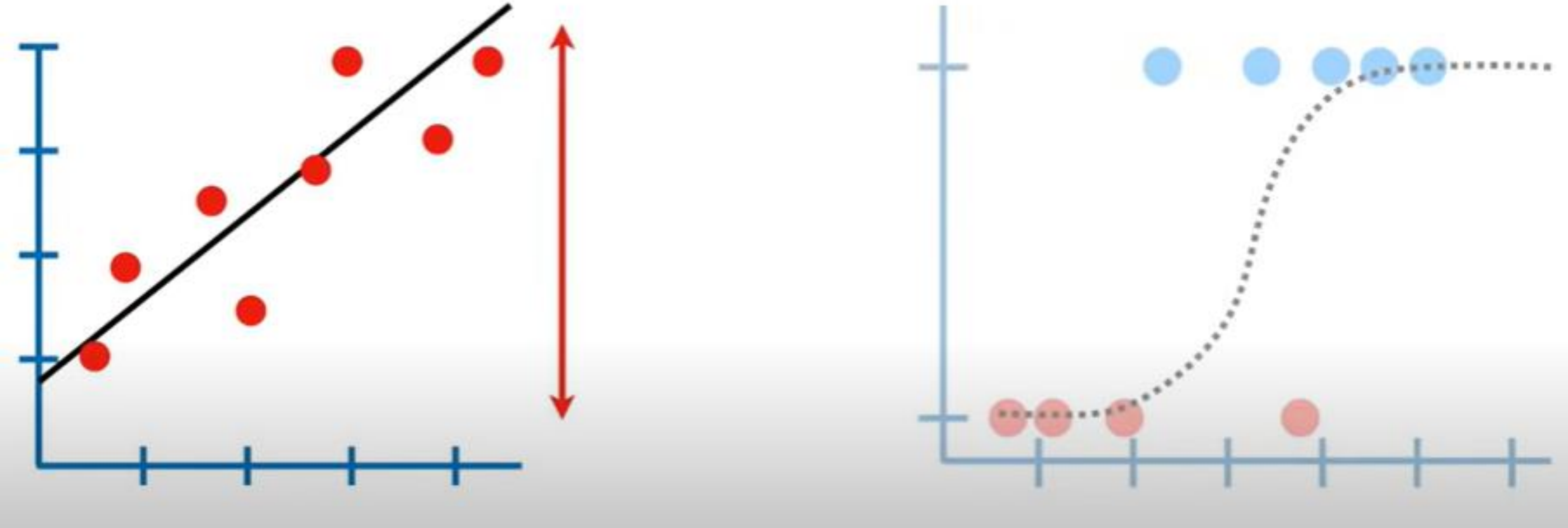
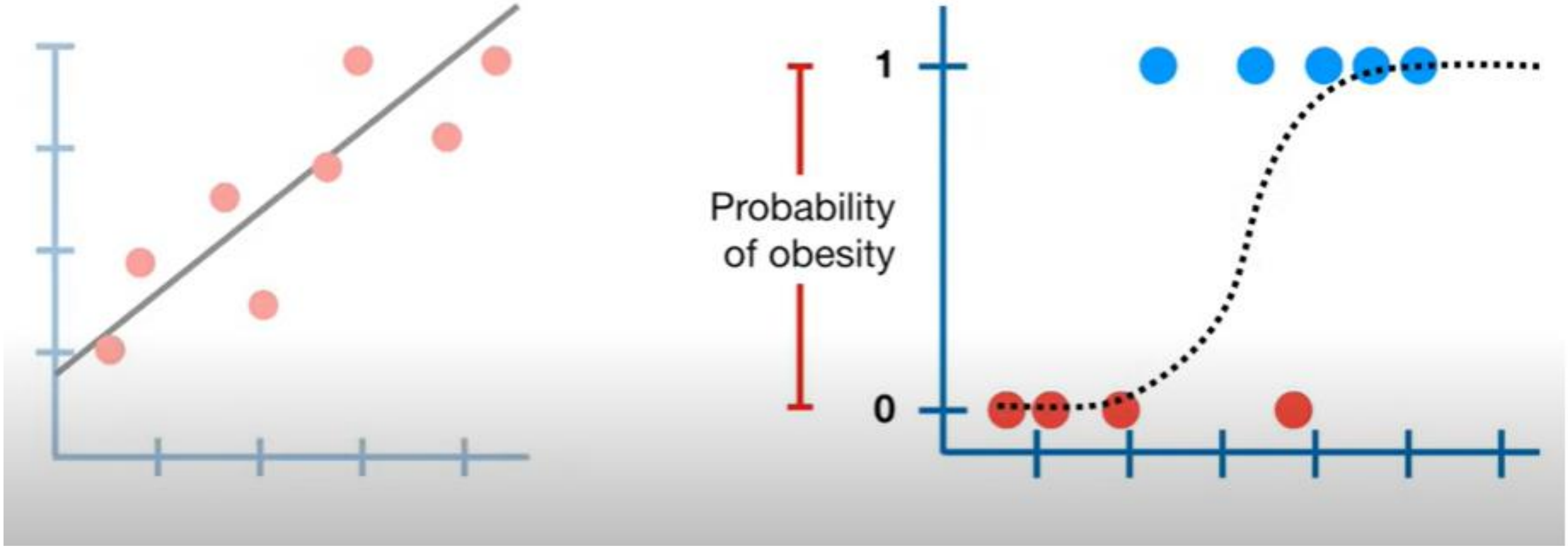
# Logistic Regression

Dr.Mrunal Rane

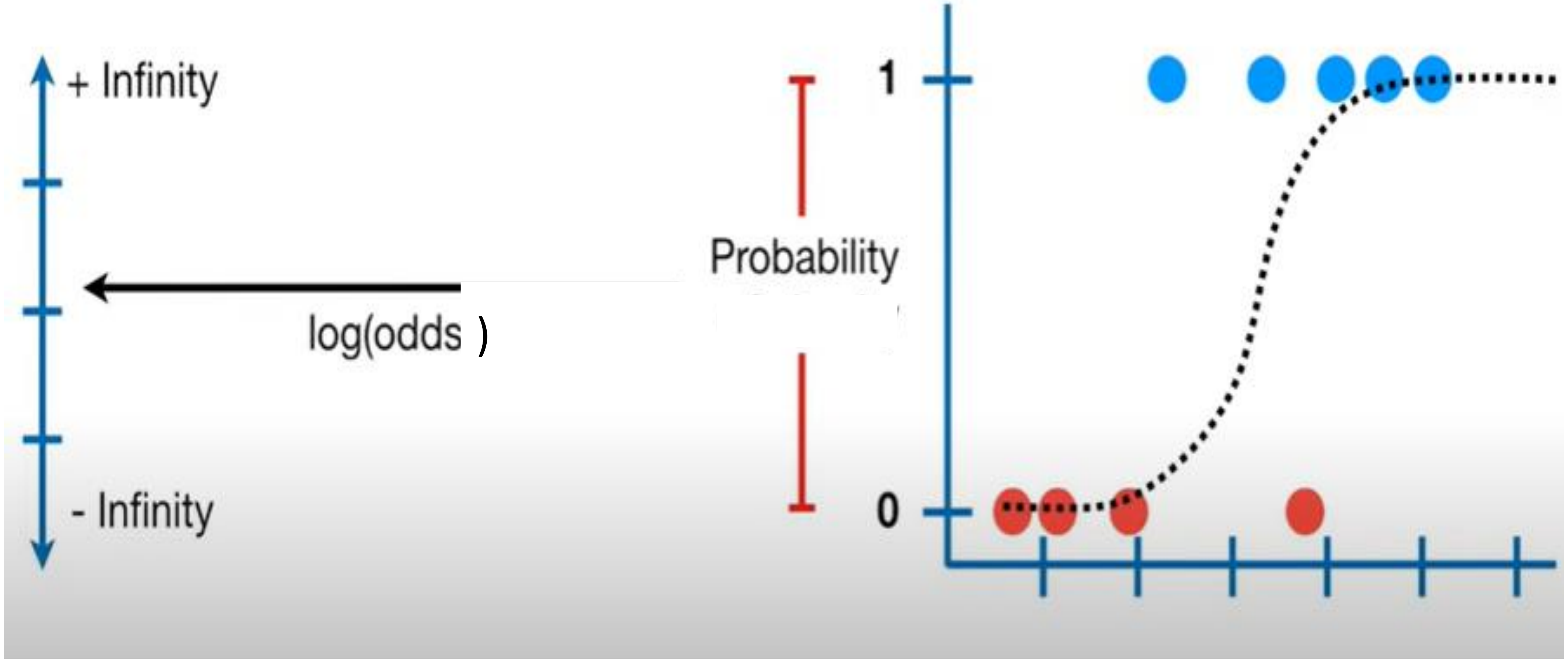
With linear regression, the values on the y-axis can, in theory, be any number…

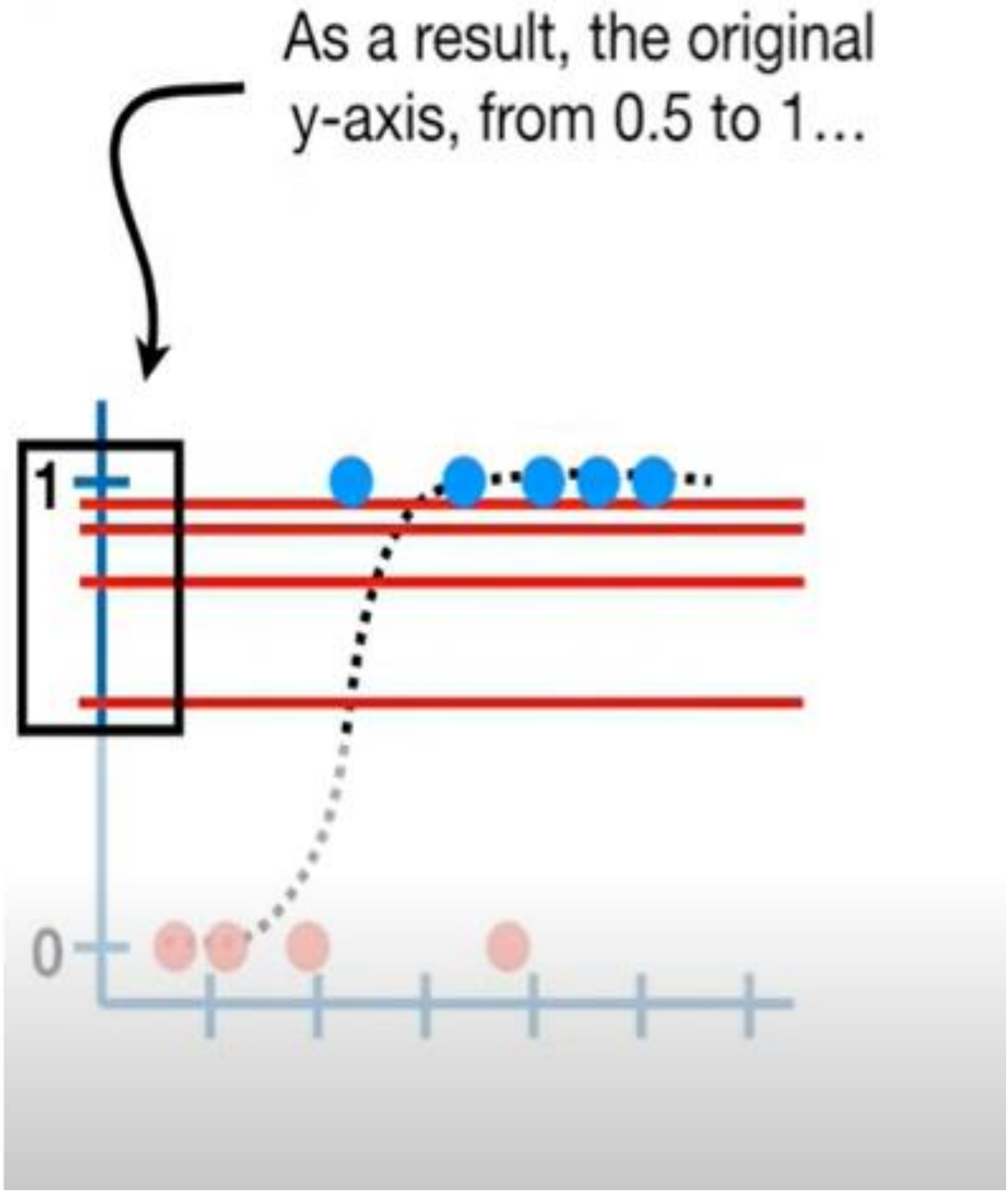


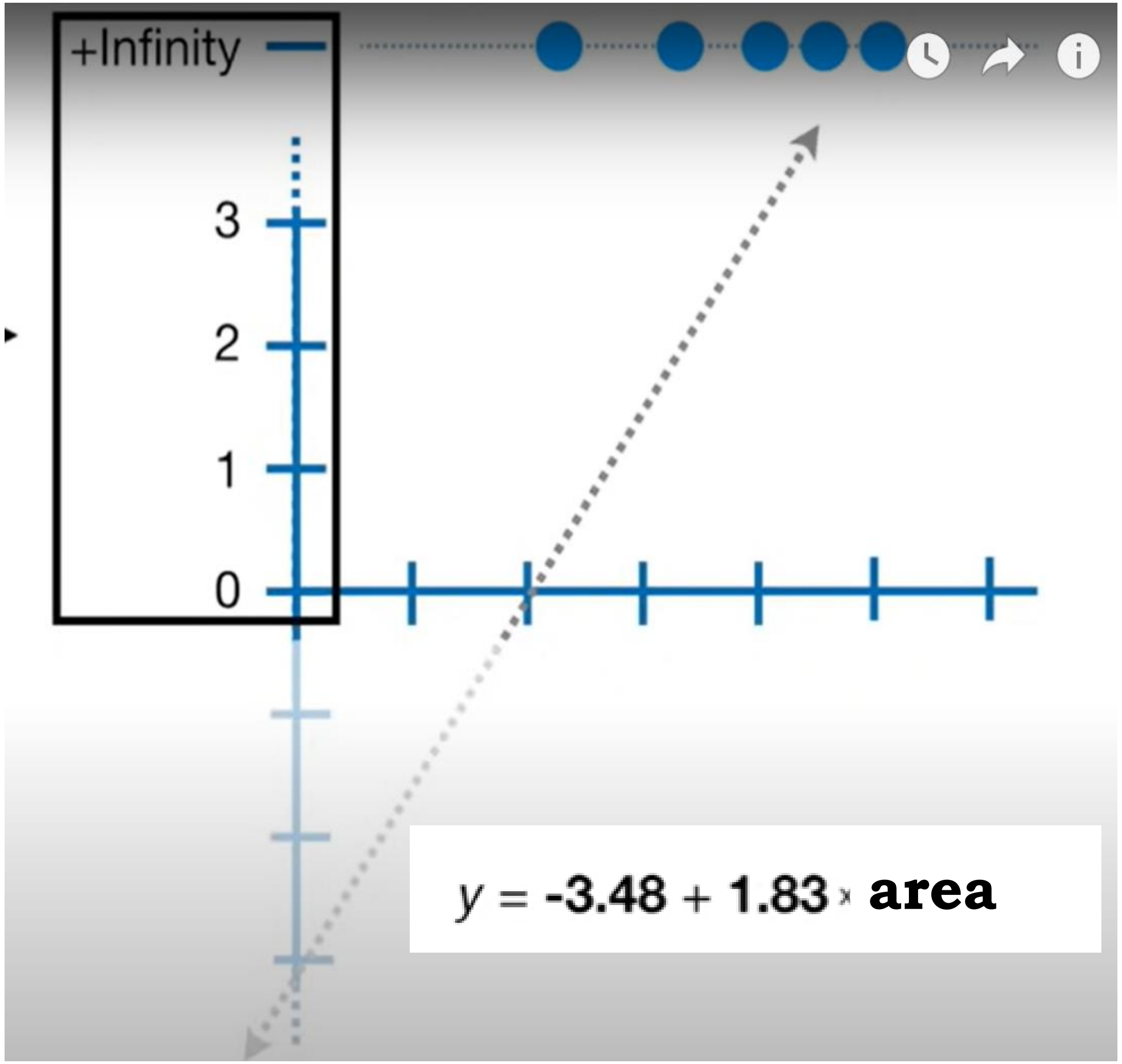
...unfortunately, with logistic regression, the y-axis is confined to probability values between 0 and 1.

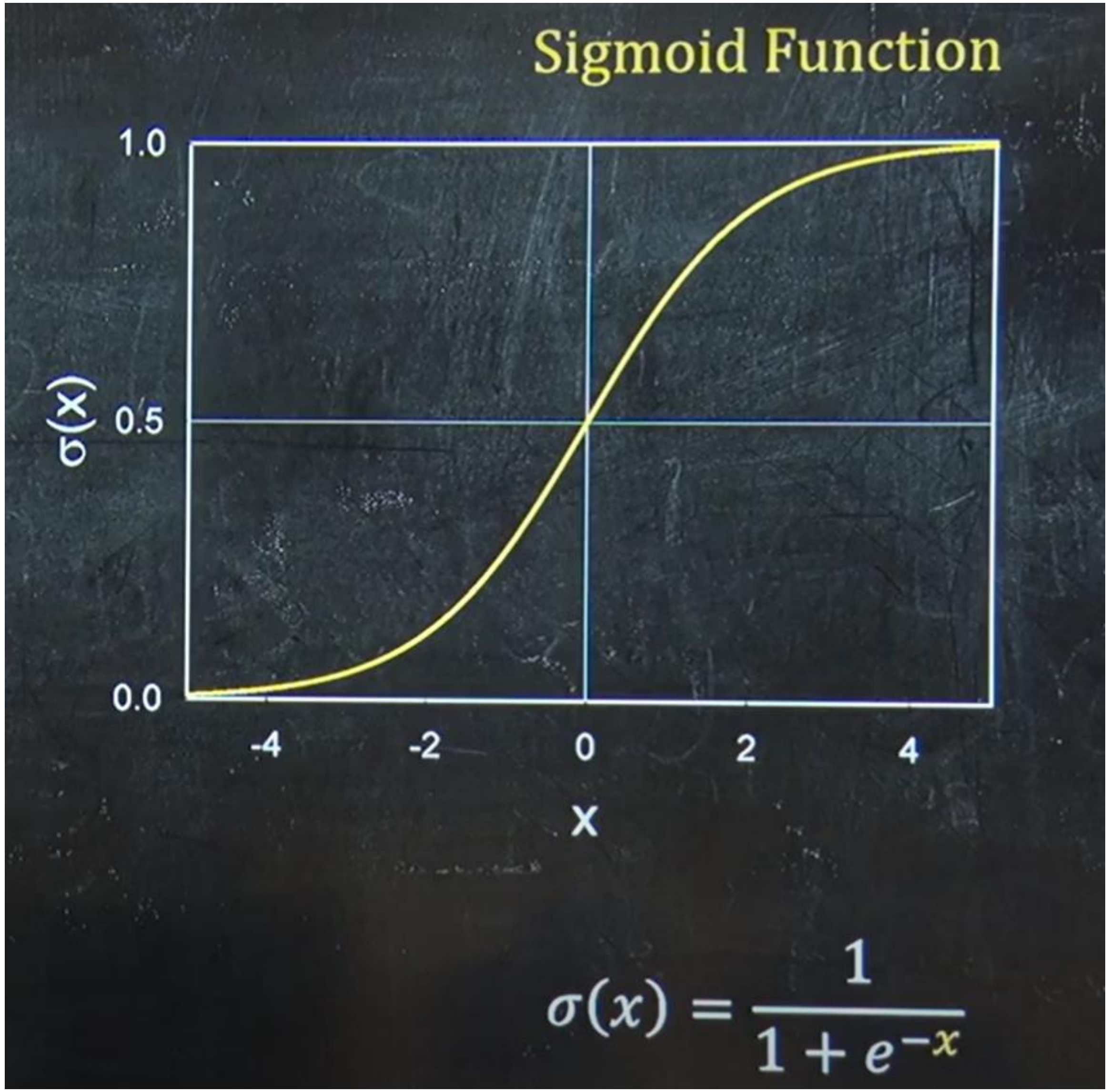


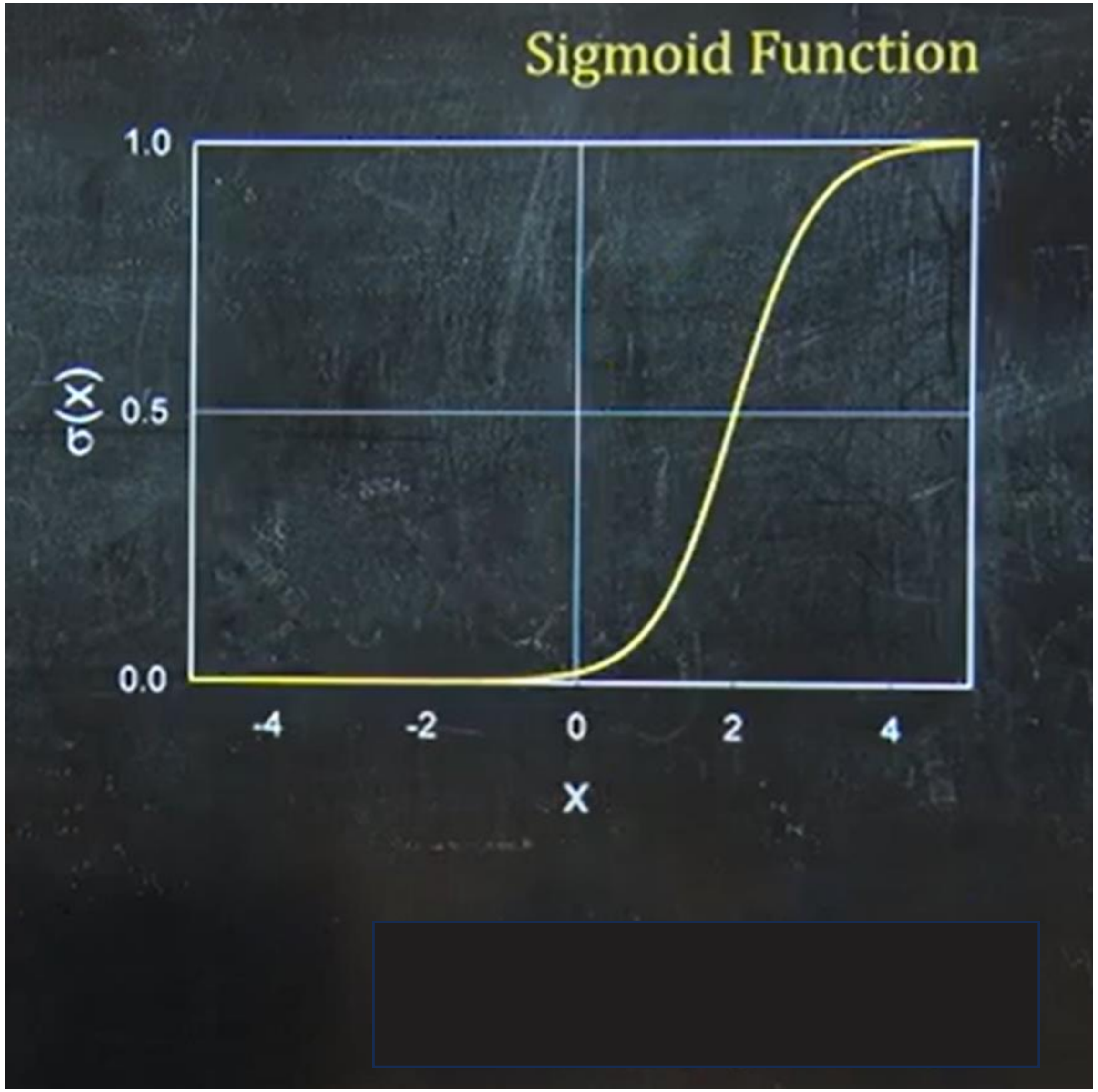
# Transformation of y-axis




As a result, the original
y-axis, from 0.5 to 1...
1
0


+Infinity
3
2
1
0
y = -3.48 + 1.83 × area


Sigmoid Function
1.0
0.5
0.0
σ(x)
-4
-2
0
2
4
x
$\sigma(x) = \frac{1}{1+e^{-x}}$


Sigmoid Function
1.0
0.5
0.0
σ(x)
-4
-2
0
2
4
x

$$\sigma(x) = \frac{1}{1 + e^{-(a+bx)}}$$

$$0 \leq \sigma(x) \leq 1$$

$$\Pr(y = 1 | x = x_i) = p(x_i) = \sigma(x_i) = \frac{1}{1 + e^{-(a+bx_i)}}$$

Probability of being in category 1

$$z = a + bx_i = \ln\left[\frac{p(x_i)}{1 - p(x_i)}\right]$$

Probability of being in category 0

# Fitting the curve using Maximum Likelihood

**Probability for being in category 1**

$$\Pr(y = 1|x = x_i) = p(x_i) = \sigma(x_i) = \frac{1}{1 + e^{-(a+bx_i)}}$$

Probability for being in category 1

$$\Pr(y=1|x=x_i) = p(x_i) = \sigma(x_i) = \frac{1}{1+e^{-(a+bx_i)}}$$

Probability for being in category 0

$$\Pr(y=0|x=x_i) = 1 - p(x_i)$$

$$P(y;k) = p^k(1-p)^{1-k}$$

[redacted]

$$p(x_i) = \frac{1}{1 + e^{-(a+bx_i)}}$$

$$\Pr(y = y_i | x = x_i) = p(x_i)^{y_i}[1 - p(x_i)]^{1-y_i}$$

$$y_i = 1 \; or \; 0$$

For n data points the likelihood function:

$$L = \prod_{i=1}^{n} p(x_i)^{y_i}[1 - p(x_i)]^{1-y_i}$$

Optimization: find a and b that maximizes L or log(L)

# MLE

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

$$\frac{d}{dx}\sigma(x) = \sigma(x)\left[1-\sigma(x)\right]$$

log function:

$$\log L(\omega) = \sum_{i=1}^{N} y_i \log\left(\frac{1}{1+e^{-x_i\omega}}\right) + (1-y_i)\log\left(1-\frac{1}{1+e^{-x_i\omega}}\right)$$

$$= \sum_{i=1}^{N} y_i \log(\sigma(x_i\omega)) + (1-y_i)\log(1-\sigma(x_i\omega))$$

Taking a derivative $\frac{d}{d\omega}(\log L(\omega)) \Rightarrow$

$$= \sum_{i=1}^{N} y_i \frac{d}{d\omega}\log(\sigma(x_i\omega)) + (1-y_i)\frac{d}{d\omega}\log(1-\sigma(x_i\omega))$$

$$= \sum_{i=1}^{N} y_i \frac{1}{\sigma(x_i\omega)}\frac{d}{d\omega}\sigma(x_i\omega) + (1-y_i)\frac{1}{1-\sigma(x_i\omega)}\left(\frac{d}{d\omega}(1) - \frac{d}{d\omega}\sigma(x_i\omega)\right)$$

$$= \sum_{i=1}^{N} y_i \frac{1}{\cancel{\sigma(x_i\omega)}}\cancel{\sigma(x_i\omega)}(1-\sigma(x_i\omega))x_i + (1-y_i)\frac{1}{\cancel{1-\sigma(x_i\omega)}}\left(0 - x_i(\sigma(x_i\omega)\cancel{(1-\sigma(x_i\omega))}\right)$$

$$= \sum_{i=1}^{N} y_i x_i - y_i x_i\sigma(x_i\omega) + (1-y_i)(-x_i\sigma(x_i\omega))$$

$$= \sum_{i=1}^{N} y_i x_i - \cancel{y_i x_i\sigma(x_i\omega)} + \cancel{y_i x_i\sigma(x_i\omega)} - x_i\sigma(x_i\omega)$$

$$= \sum_{i=1}^{N}\left[y_i - \sigma(x_i\omega)\right]x_i$$

$$\boxed{= \sum_{i=1}^{N}\left[y_i - \frac{1}{1+e^{-x_i\omega}}\right]x_i} \leftarrow \text{Gradient of log function.}$$

# Log-Loss

- Log-loss is indicative of how close the prediction probability is to the corresponding actual/true value (0 or 1 in case of binary classification), penalizing inaccurate predictions with higher values.
- Lower log-loss indicates better model performance.

$$log\ loss = -1/N \sum_{i=1}^{N} (log\,(Pi))$$

# Log Loss Example

| ID | Actual | Predicted Probabilities | Corrected Probabilities | Log |
|---|---|---|---|---|
| ID6 | 1 | 0.94 | 0.94 | -0.02687 |
| ID1 | 1 | 0.9 | 0.9 | -0.04576 |
| ID7 | 1 | 0.78 | 0.78 | -0.10791 |
| ID8 | 0 | 0.56 | 0.44 | -0.35655 |
| ID2 | 0 | 0.51 | 0.49 | -0.3098 |
| ID3 | 1 | 0.47 | 0.47 | -0.3279 |
| ID4 | 1 | 0.32 | 0.32 | -0.49485 |
| ID5 | 0 | 0.1 | 0.9 | -0.04576 |

# Steps to find Log Loss

1. **To find corrected probabilities.**

2. **Take a log of corrected probabilities.**

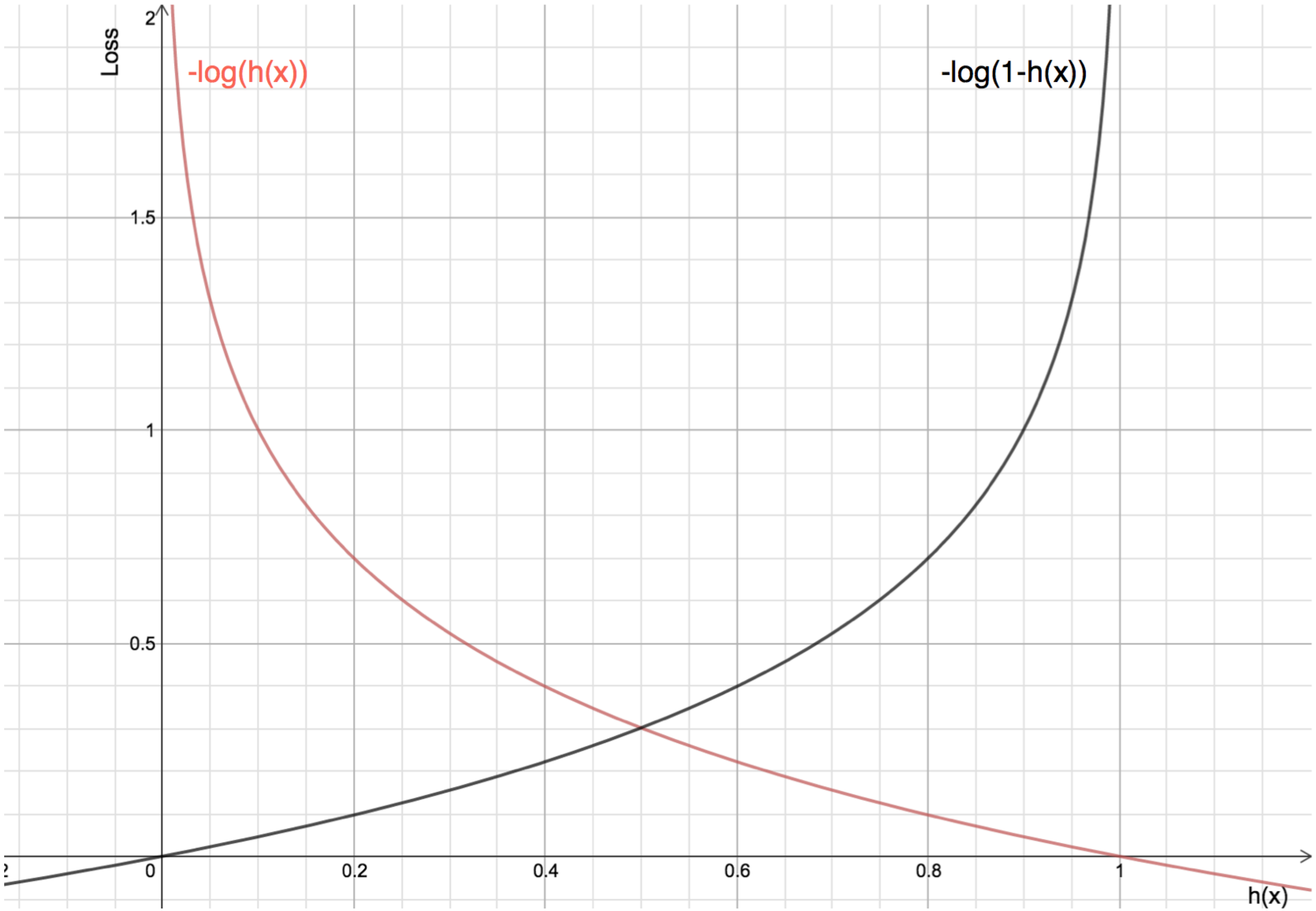
3. **Take the negative average of the values we get in the 2nd step.**

$$-\frac{1}{N}\sum_{i=1}^{N} y_i \cdot log(p(y_i)) + (1 - y_i) \cdot log(1 - p(y_i))$$

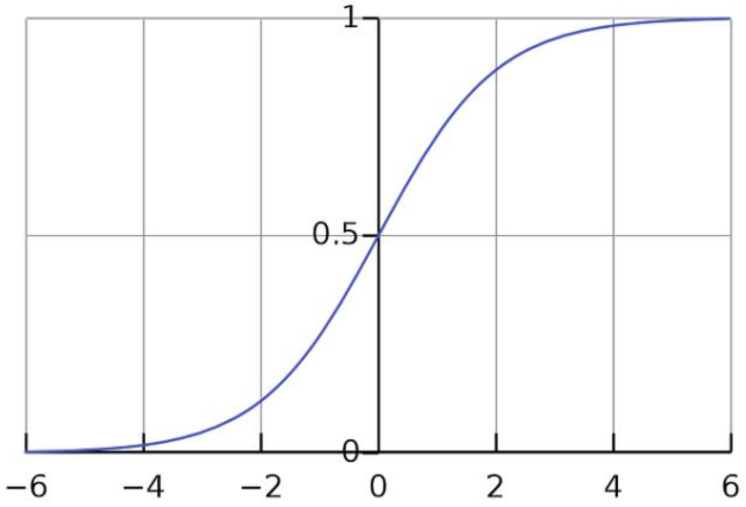
Binary Cross-Entropy / Log Loss

- Here Yi represents the actual class and log(p(yi) is the probability of that class.
- p(yi) is the probability of 1.
- 1-p(yi) is the probability of 0.

# Log Loss



## Sigmoidal Function




$$g(z) = \frac{1}{1+e^{-z}}$$

## Hypothesis

y_hat = $\dfrac{1}{1+e^{-(\boldsymbol{w}\cdot\boldsymbol{x}+b)}}$

P(y = 1 | X; w, b) = y_hat

P(y = 0 | X; w, b) = (1-y_hat)

## Loss function

$$J(w,b) = \frac{1}{m}\sum_{i=1}^{m} L(\hat{y}^{(i)}, y^{(i)}) = -\frac{1}{m}\sum_{i=1}^{m}[(y^{(i)}\log(\hat{y}^{(i)}) + (1-y^{(i)})\log(1-\hat{y}^{(i)})]$$

**Notations —**

- `n` →number of features
- `m` →number of training examples
- `X` →input data matrix of shape ( `m` x `n` )
- `y` →true/ target value (**can be 0 or 1 only**)
- `x(i), y(i)` →ith training example
- `w` → weights (parameters) of shape ( `n` x 1)
- `b` →bias (parameter), a real number that can be broadcasted.
- `y_hat` (y with a cap/hat)→ hypothesis (**outputs values between 0 and 1**)