- Introduction to Decision Tree by examples
- Building a Decision Tree using Entropy and Information Gain for Classification

Lecture-6,7,8

- Practical of Decision Tree with Regression
- (Salary and Production Cost Prediction)

Supervised Learning

(Logistic Regression)

Dr. Virendra Singh Kushwah

Assistant Professor Grade-II

School of Computing Science and Engineering

Virendra.Kushwah@vitbhopal.ac.in

7415869616

What is Regression?

- Regression analysis is a powerful statistical analysis technique. A dependent variable of our interest is used to predict the values of other independent variables in a data-set.
- We come across regression in an intuitive way all the time. Like predicting the weather using the data-set of the weather conditions in the past.
- It uses many techniques to analyses and predict the outcome, but the emphasis is mainly on relationship between dependent variable and one or more independent variable.
- Logistic regression analysis predicts the outcome in a binary variable which has only two possible outcomes.

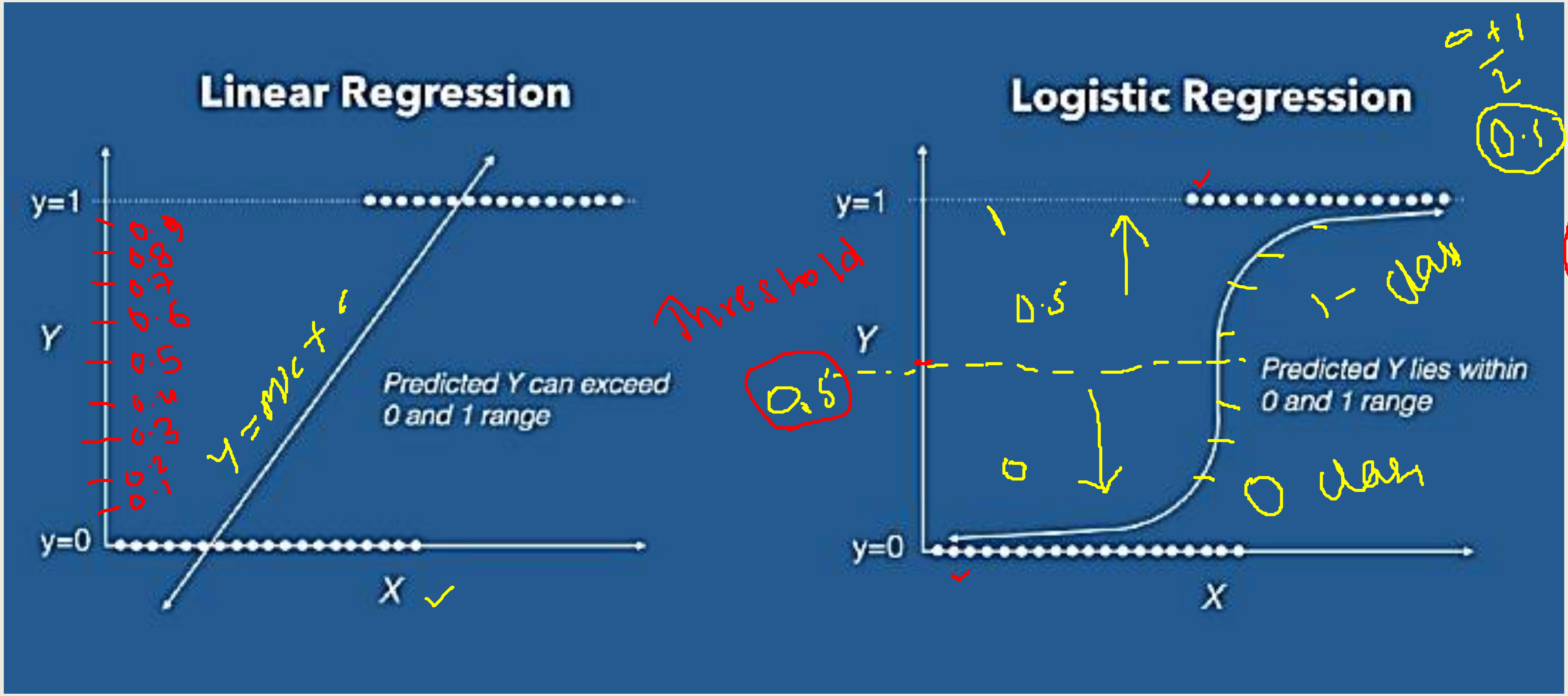
What Is Logistic Regression?

- Logistic regression is a *classification algorithm*, used when the value of the target variable is categorical in nature. Logistic regression is most commonly used when the data in question has binary output, so when it belongs to one class or another, or is either a 0 or 1.

- Remember that classification tasks have discrete categories, unlike regressions tasks.

- Logistic Regression is a Machine Learning algorithm which is used for the classification problems, it is a predictive analysis algorithm and based on the concept of probability.

x = ?
y = 0.3
0 or 1



0 < 0.5 < 1
threshold

Logistic Regression

I
S
B
S
y
→ D
0
1
1
0

- It is a technique to analyze a data-set which has a dependent variable and one or more independent variables to predict the outcome in a binary variable, meaning it will have only two outcomes.

(0,1) - Binary

- The dependent variable is categorical in nature. Dependent variable is also referred as target variable and the independent variables are called the predictors.

or features

$w = 50$ $0 - 1$ 0.5

$y = \boxed{36}$ $n = 14$

$36 >= 25$

$\boxed{y = m_1x_1^2 + m_1x_1 + c}$

$y = mx + c$

$y = m_1x_1 + m_2x_2 + \cdots + m_nx_n + c$

- **Logistic regression is a special case of linear regression** where we only predict the outcome in a categorical variable. It predicts the probability of the event using the **log function**.

logit

$y = 9$ $y \rightarrow 0$ or 1

- *We use the Sigmoid function/curve to predict the categorical value. The threshold value decides the outcome(win/lose).*

$r = 0.5$

$y = 0.5 \rightarrow \boxed{1}$

$0 < y <= 1$

0.5

$\boxed{y = 0.5432} >= 0.5$

$y = mx + c$

- We can call a Logistic Regression a Linear Regression model but the Logistic Regression uses a more complex cost function, this cost function can be defined as the 'Sigmoid function' or also known as the 'logistic function' instead of a linear function.
- The hypothesis of logistic regression tends it to limit the cost function between 0 and 1. Therefore linear functions fail to represent it as it can have a value greater than 1 or less than 0 which is not possible as per the hypothesis of logistic regression.

$$0 \leq \boxed{h_\theta(x)} \leq 1 \rightarrow \text{Threshold}$$

What is the Sigmoid Function?

- In order to map predicted values to probabilities, we use the Sigmoid function. The function maps any real value into another value between 0 and 1. In machine learning, we use sigmoid to map predictions to probabilities.

- The sigmoid function/logistic function is a function that resembles an “S” shaped curve when plotted on a graph. It takes values between 0 and 1 and “squishes” them towards the margins at the top and bottom, labeling them as 0 or 1.

$y = \boxed{\frac{1}{1+e^{-x}}}$

- The equation for the Sigmoid function is this:

$$\underset{\text{dependent variable}}{y} = 1/(1+e^{\overset{\text{independent}}{-x}})$$

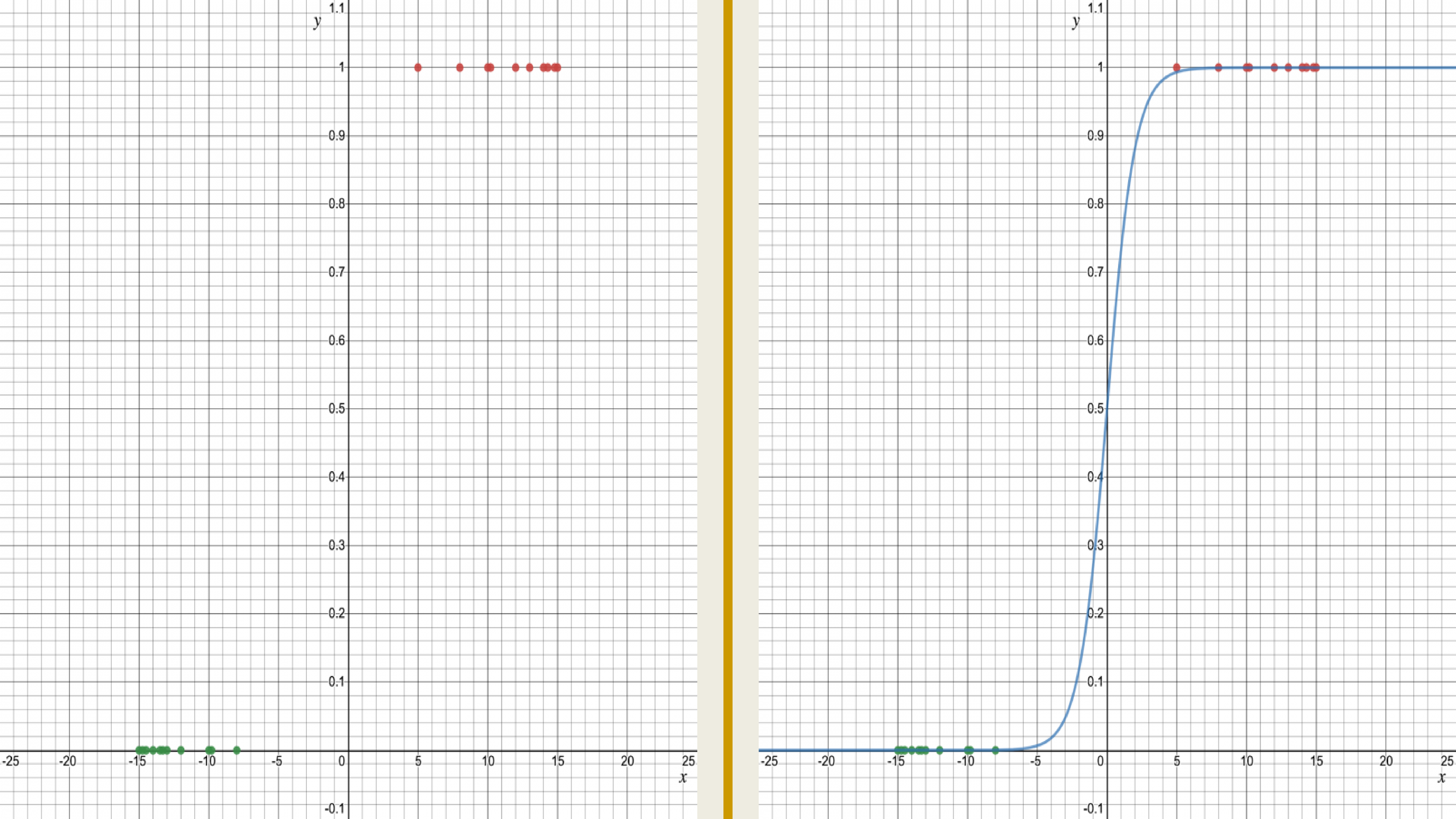
$\boxed{y = mx + c}$

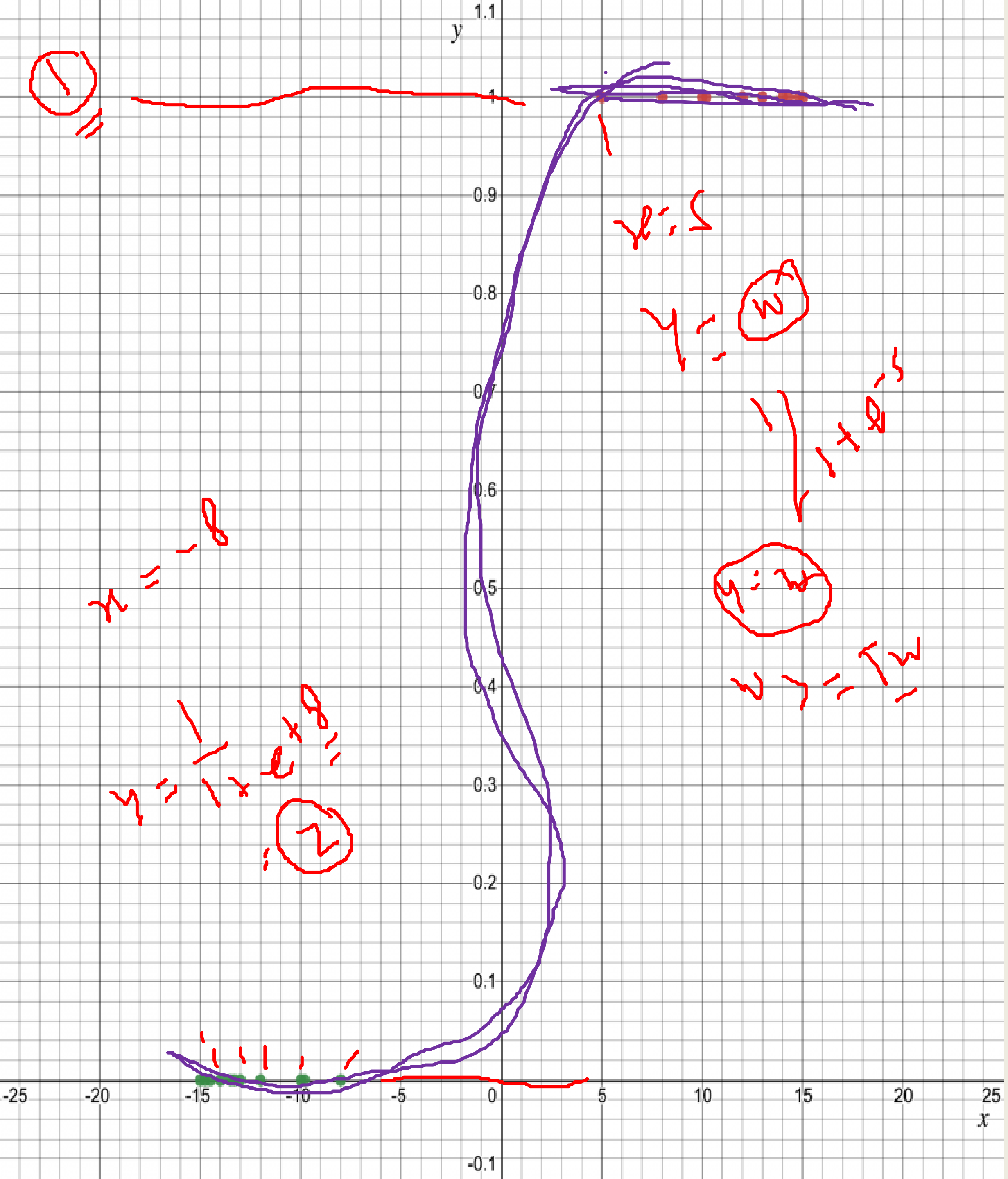
$\boxed{y = \frac{1}{1+e^{-x}}}$

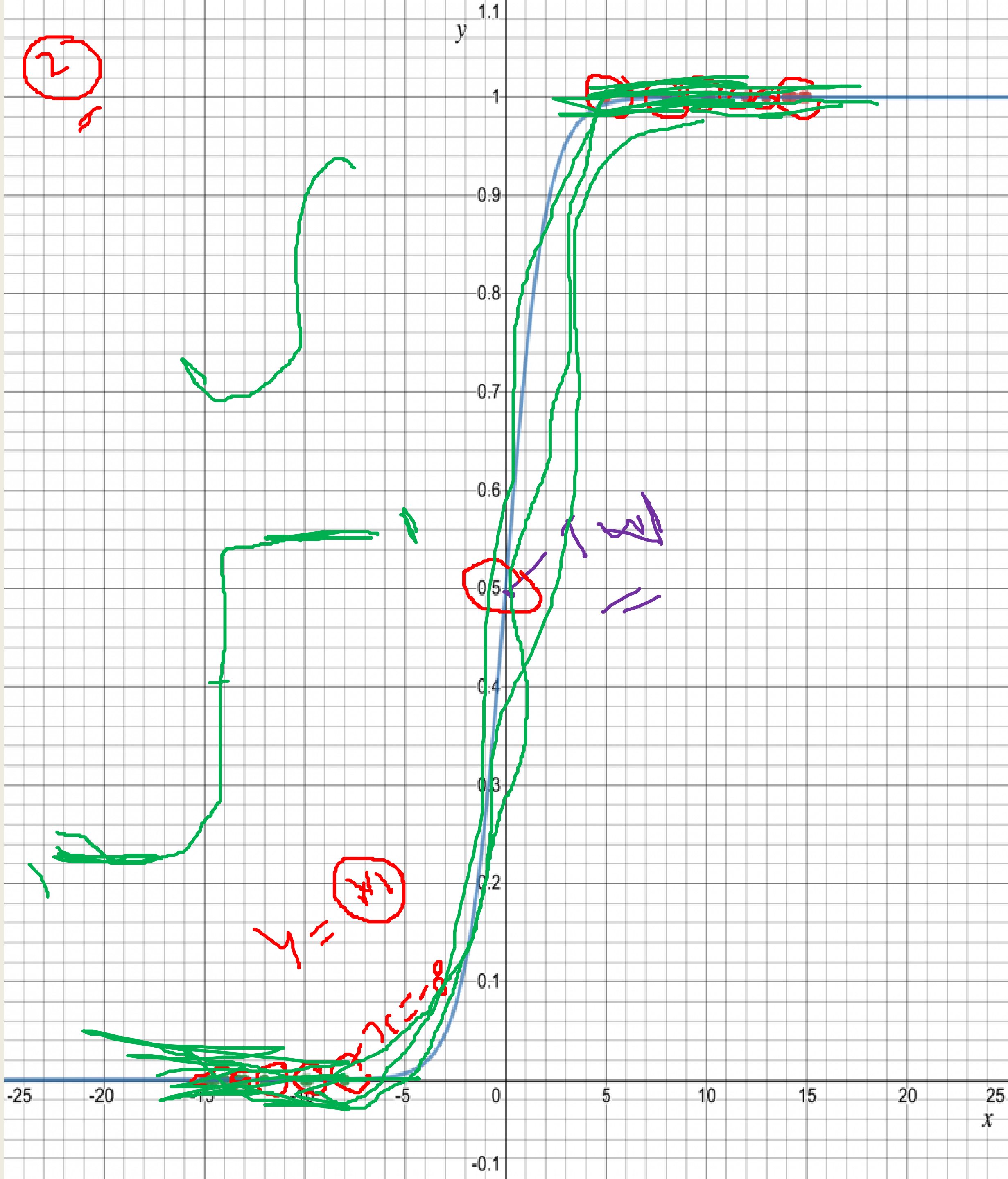
- What is the variable e in this instance? The e represents the exponential function or exponential constant, and it has a value of approximately 2.71828.

$y = \frac{1}{1+e^{-(b_1x_1+b_2x_2 \cdots b_nx_n)}}$

LR by MS

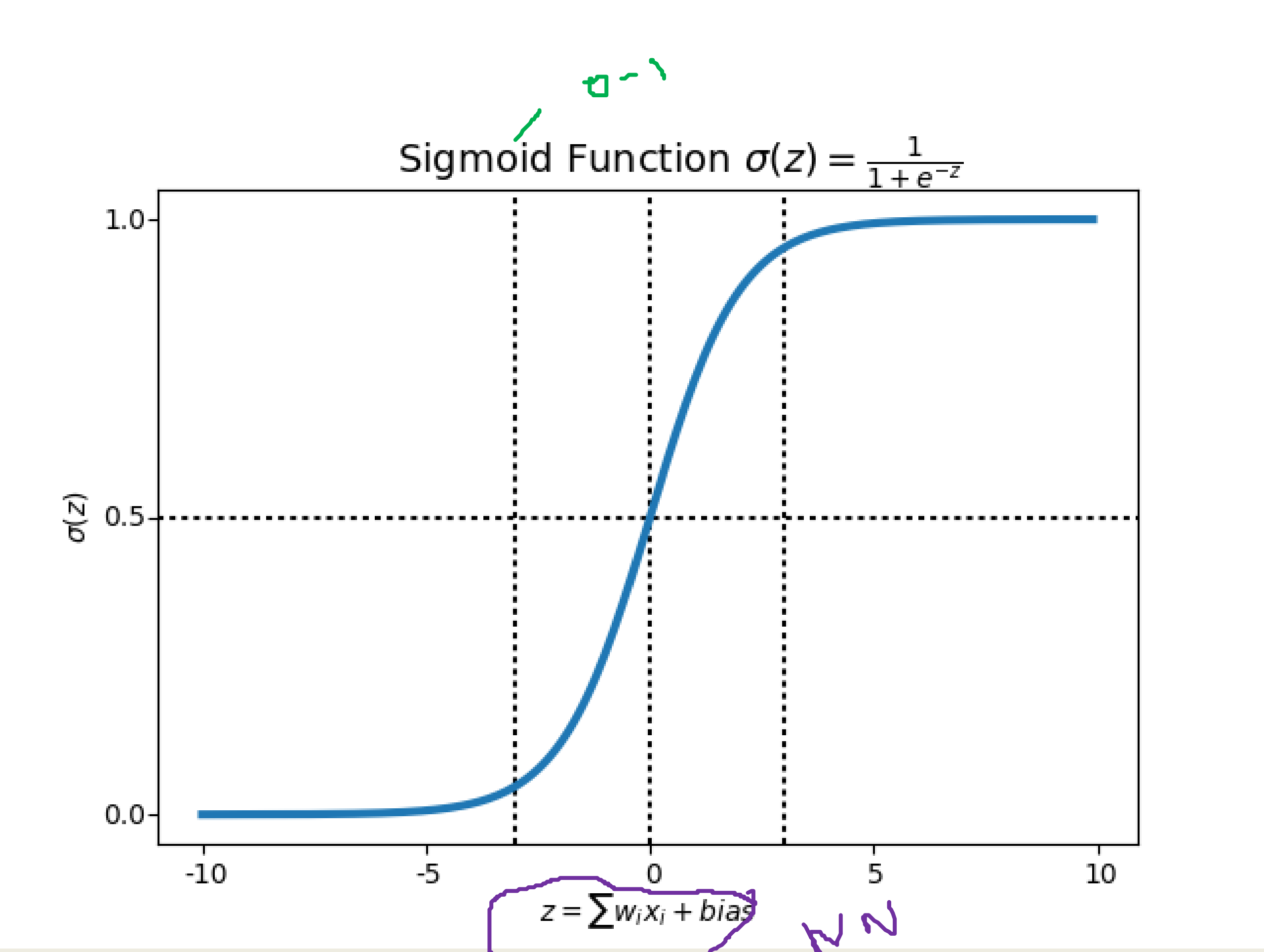

y
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
-0.1
-25
-20
-15
-10
-5
0
5
10
15
20
25
x
y
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
-0.1
-25
-20
-15
-10
-5
0
5
10
15
20
25
x

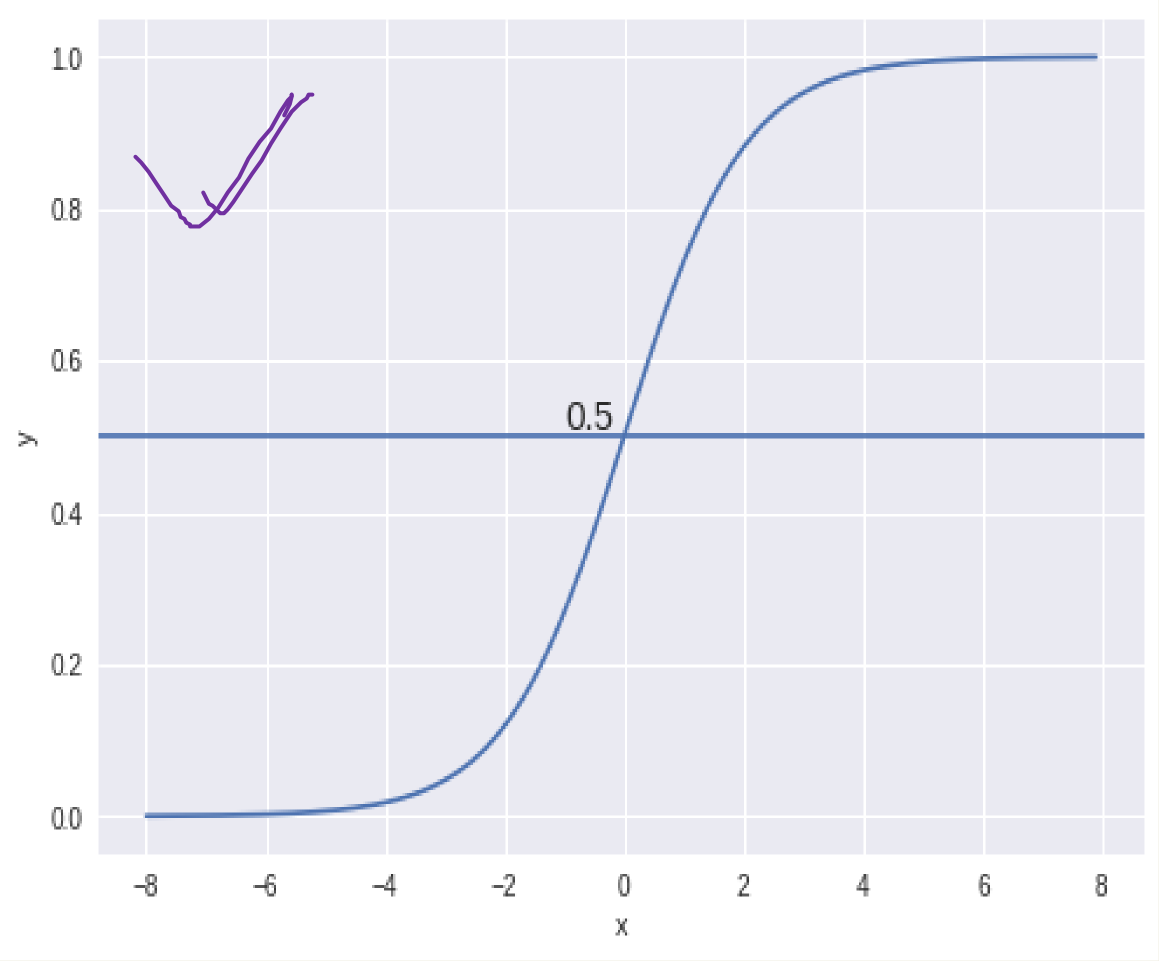

1
y
x
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
-0.1
-25
-20
-15
-10
-5
0
5
10
15
20
25
y = w
y = w
2


2
y
x
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
-0.1
-25
-20
-15
-10
-5
0
5
10
15
20
25
y =
x = -8

- This gives a value y that is extremely close to 0 if x is a large negative value and close to 1 if x is a large positive value. After the input value has been squeezed towards 0 or 1, the input can be run through a typical linear function, but the inputs can now be put into distinct categories.

Example of Sigmoid function





✓ $\boxed{1 - D}$ ✓

$\boxed{0 - C}$

Example

- We have 2 classes, let’s take them like cats and dogs(1 — dog , 0 — cats). We basically decide with a threshold value above which we classify values into Class 1 and of the value goes below the threshold then we classify it in Class 2.

$\boxed{0.7} > 0.5$

$x = 2.$

$y = \frac{1}{1+e^{-x}} = 0.7$

$\boxed{0.4}$

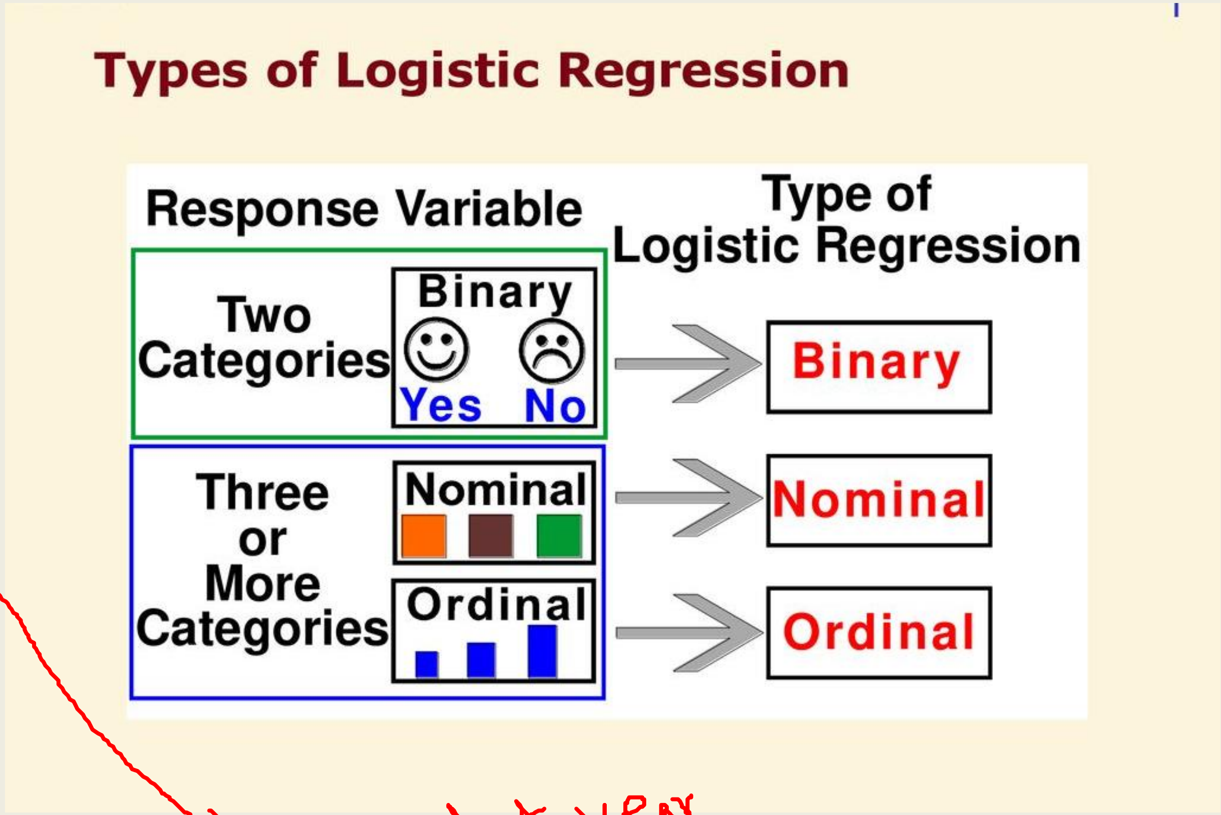
- Suppose we have chosen the threshold as 0.5, if the prediction function returned a value of 0.7 then we would classify this observation as Class 1(DOG). If our prediction returned a value of 0.2 then we would classify the observation as Class 2(CAT).

Types Of Logistic Regression

99%. *(0,1) T/F , Y/N

- Binary logistic regression – It has only two possible outcomes. Example- yes or no
- Multinomial logistic regression – It has three or more nominal categories. Example- cat, dog, elephant.
- Ordinal logistic regression- It has three or more ordinal categories, ordinal meaning that the categories will be in a order. Example- user ratings(1-5).

1, 2, 3, 4, 5,



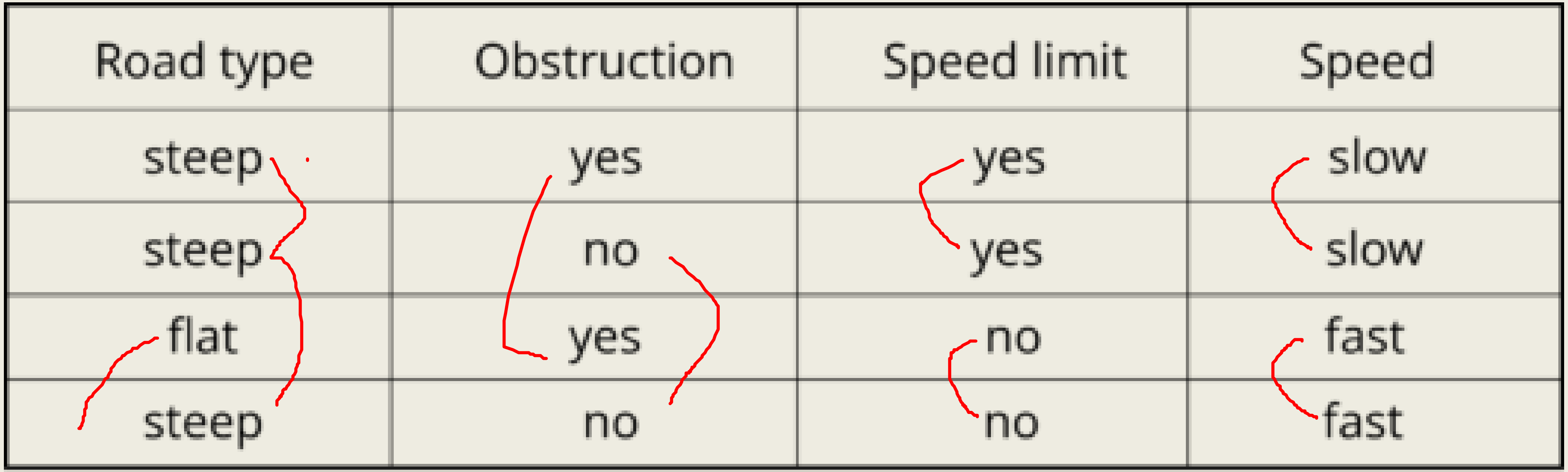
Types of Logistic Regression

Student year

I, II, III

Problem

→ Decision Tree with classification



Road type	Obstruction	Speed limit	Speed
steep	yes	yes	slow
steep	no	yes	slow
flat	yes	no	fast
steep	no	no	fast

- Your problem statement is to study this data set and create a Decision Tree that classifies the speed of a car (response variable) as either slow or fast, depending on the following predictor variables:

- Road type
- Obstruction
- Speed limit

- We’ll be building a Decision Tree using these variables in order to predict the speed of a car. Like I mentioned earlier we must first begin by deciding a variable that best splits the data set and assign that particular variable to the root node and repeat the same thing for the other nodes as well.

- At this point, you might be wondering how do you know which variable best separates the data? The answer is, the variable with the highest Information Gain best divides the data into the desired output classes.

- So, let’s begin by calculating the Entropy and Information Gain (IG) for each of the predictor variables, starting with ‘Road type’.

- In our data set, there are four observations in the ‘Road type’ column that correspond to four labels in the ‘Speed of car’ column. We shall begin by calculating the entropy of the parent node (Speed of car).

- Step one is to find out the fraction of the two classes present in the parent node. We know that there are a total of four values present in the parent node, out of which two samples belong to the 'slow' class and the other 2 belong to the 'fast' class, therefore:

- P(slow) -> fraction of 'slow' outcomes in the parent node
- P(fast) -> fraction of 'fast' outcomes in the parent node

- The formula to calculate P(slow) is:

- p(slow) = no. of ‘slow’ outcomes in the parent node / total number of outcomes

$$P_{\text{slow}} = \frac{2}{4} = 0.5$$

- Similarly, the formula to calculate P(fast) is:

- p(fast) = no. of ‘fast’ outcomes in the parent node / total number of outcomes

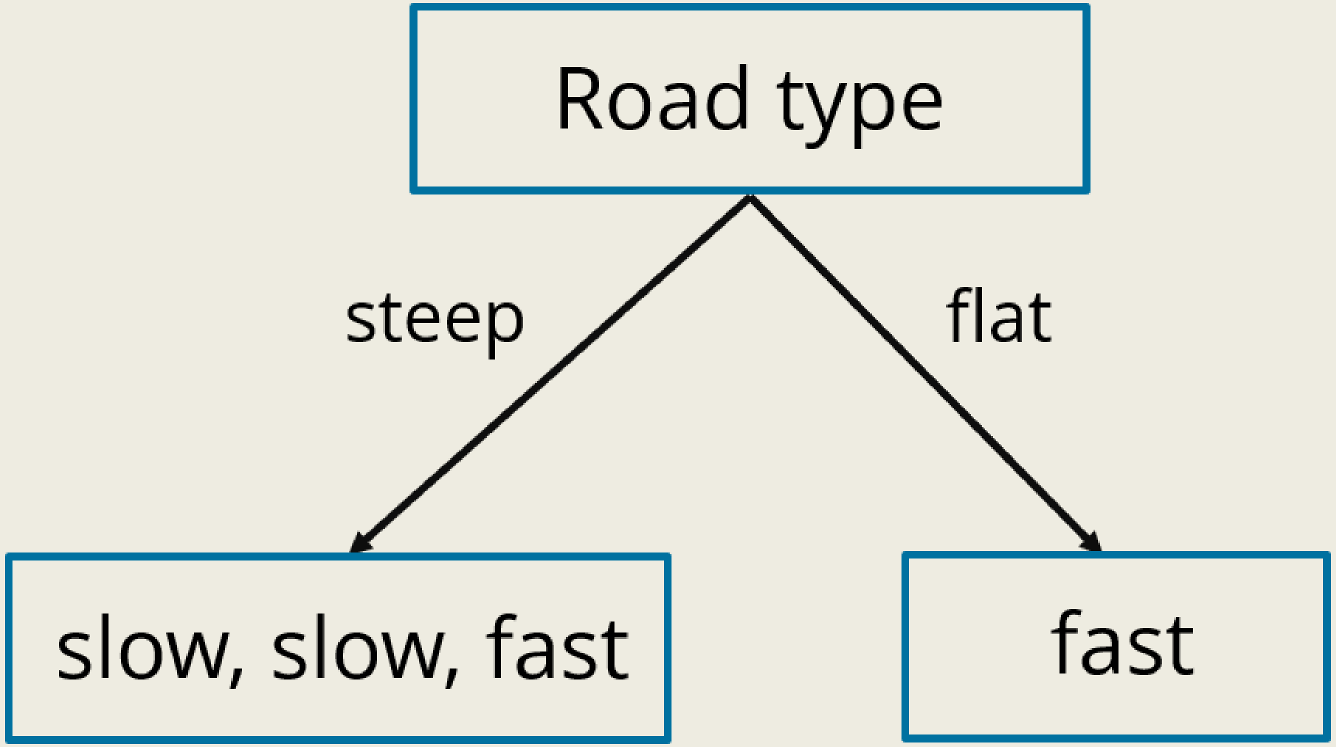
$$P_{\text{fast}} = \frac{2}{4} = 0.5$$

- Therefore, the entropy of the parent node is:

$$Entropy_{\text{parent}} = -\Sigma p_{slow} log_2(p_{slow}) + p_{fast} log_2(p_{fast})$$

- Entropy(parent) = – {0.5 log2(0.5) + 0.5 log2(0.5)} = – {-0.5 + (-0.5)} = 1

- Now that we know that the entropy of the parent node is 1, let’s see how to calculate the Information Gain for the ‘Road type’ variable. Remember that, if the Information gain of the ‘Road type’ variable is greater than the Information Gain of all the other predictor variables, only then the root node can be split by using the ‘Road type’ variable.


Road type
steep
flat
slow, slow, fast
fast

- we’ve split the parent node by using the ‘Road type’ variable, the child nodes denote the corresponding responses as shown in the data set. Now, we need to measure the entropy of the child nodes.

- The entropy of the right-hand side child node (fast) is 0 because all of the outcomes in this node belongs to one class (fast). In a similar manner, we must find the Entropy of the left-hand side node (slow, slow, fast).

- In this node there are two types of outcomes (fast and slow), therefore, we first need to calculate the fraction of slow and fast outcomes for this particular node.

- P(slow) = 2/3 = 0.667
- P(fast) = 1/3 = 0.334

- Therefore, entropy is:

- Entropy(left child node) = – {0.667 log2(0.667) + 0.334 log2(0.334)} = – {-0.38 + (-0.52)}
- = 0.9

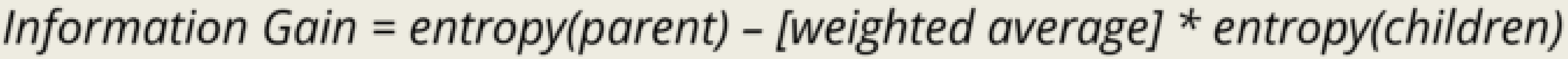
- Our next step is to calculate the Entropy(children) with weighted average:

- Total number of outcomes in parent node: 4
- Total number of outcomes in left child node: 3
- Total number of outcomes in right child node: 1

- Formula for Entropy(children) with weighted avg. :
- *[Weighted avg]Entropy(children) = (no. of outcomes in left child node) / (total no. of outcomes in parent node) * (entropy of left node) + (no. of outcomes in right child node)/ (total no. of outcomes in parent node) * (entropy of right node)*

- By using the above formula you'll find that the, Entropy(children) with weighted avg. is = 0.675

- Our final step is to substitute the above weighted average in the IG formula in order to calculate the final IG of the ‘Road type’ variable:



*Information Gain = entropy(parent) – [weighted average] * entropy(children)*

- Information gain(Road type) = 1 – 0.675 = 0.325

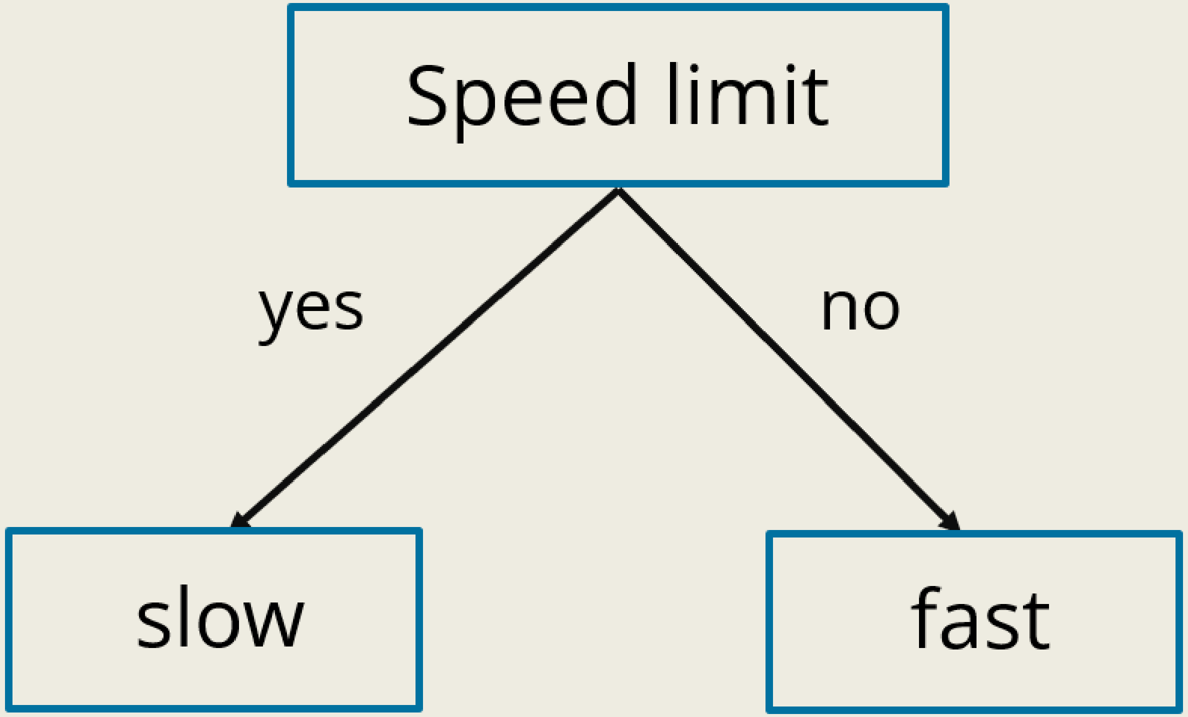
- Information gain of Road type feature is 0.325.

- Like I mentioned earlier, the Decision Tree Algorithm selects the variable with the highest Information Gain to split the Decision Tree. Therefore, by using the above method you need to calculate the Information Gain for all the predictor variables to check which variable has the highest IG.

- So by using the above methodology, you must get the following values for each predictor variable:

- Information gain(Road type) = 1 – 0.675 = 0.325
- Information gain(Obstruction) = 1 – 1 = 0
- Information gain(Speed limit) = 1 – 0 = 1

- So, here we can see that the ‘Speed limit’ variable has the highest Information Gain. Therefore, the final Decision Tree for this dataset is built using the ‘Speed limit’ variable.



Speed limit

yes

no

slow

fast