Module II

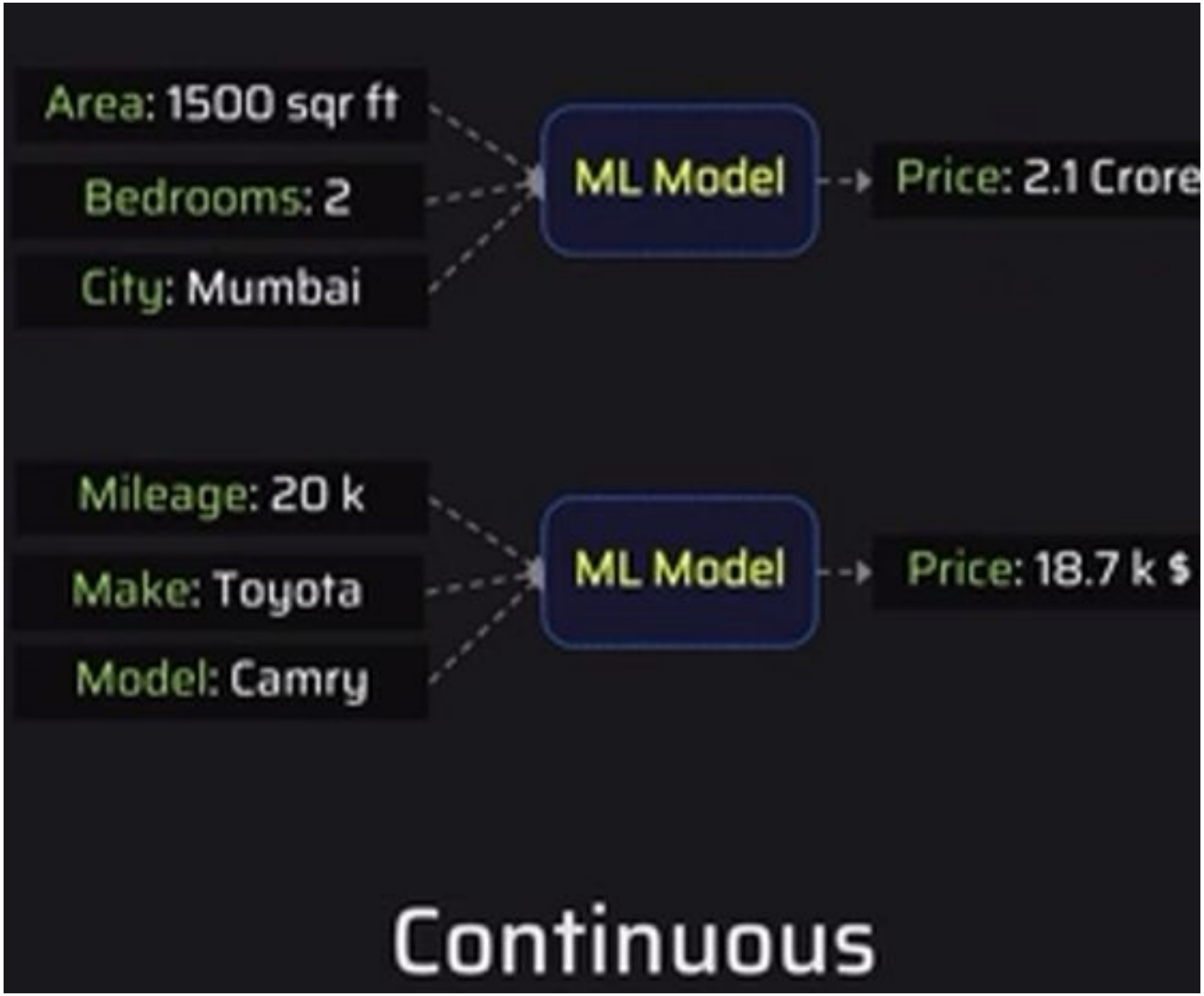
Part 2
CLASSIFICATION

Classification

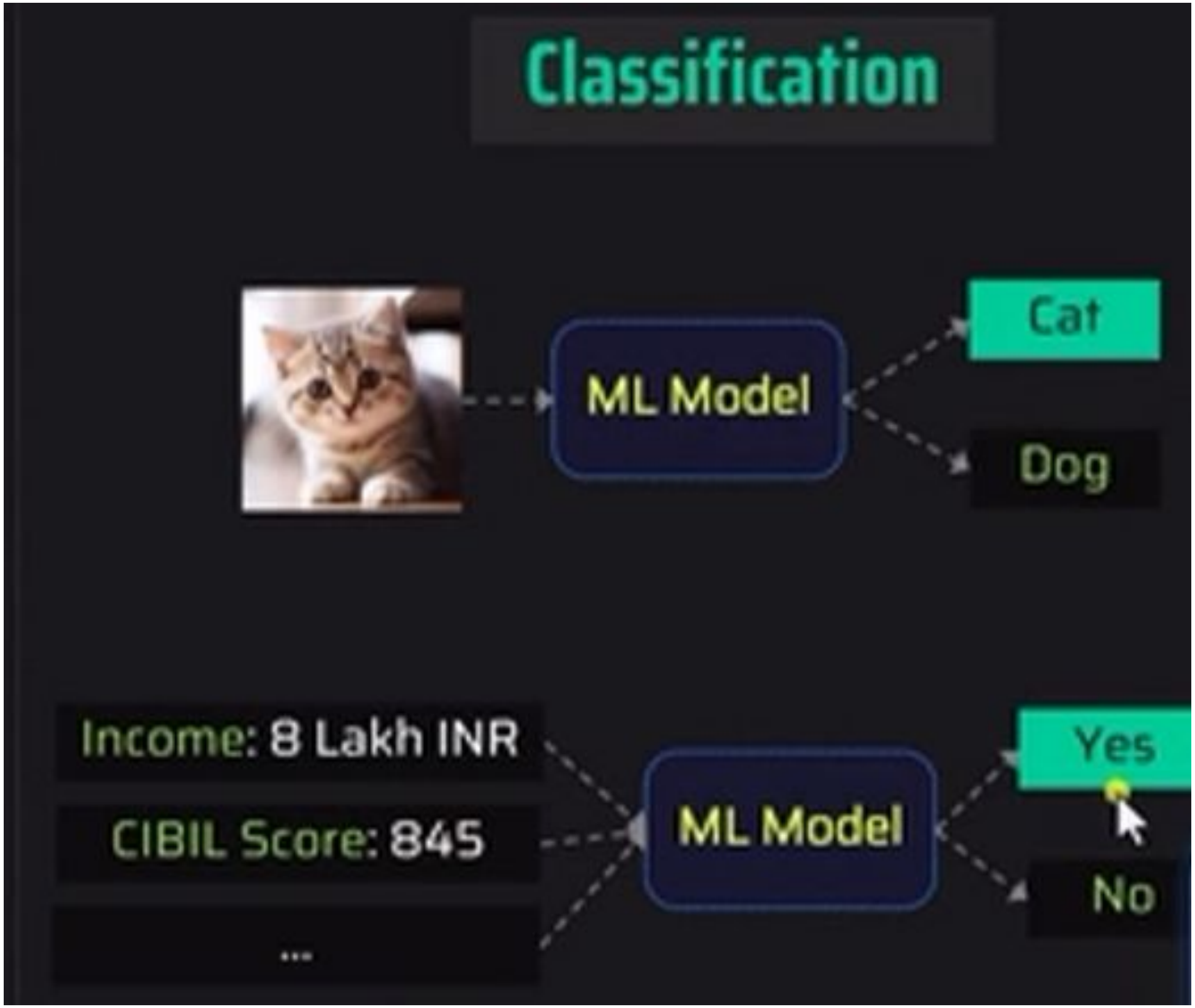
Classification in machine learning (ML) is a **supervised learning technique** where algorithms are trained on labeled data to predict the category of new, unseen data. The process involves:

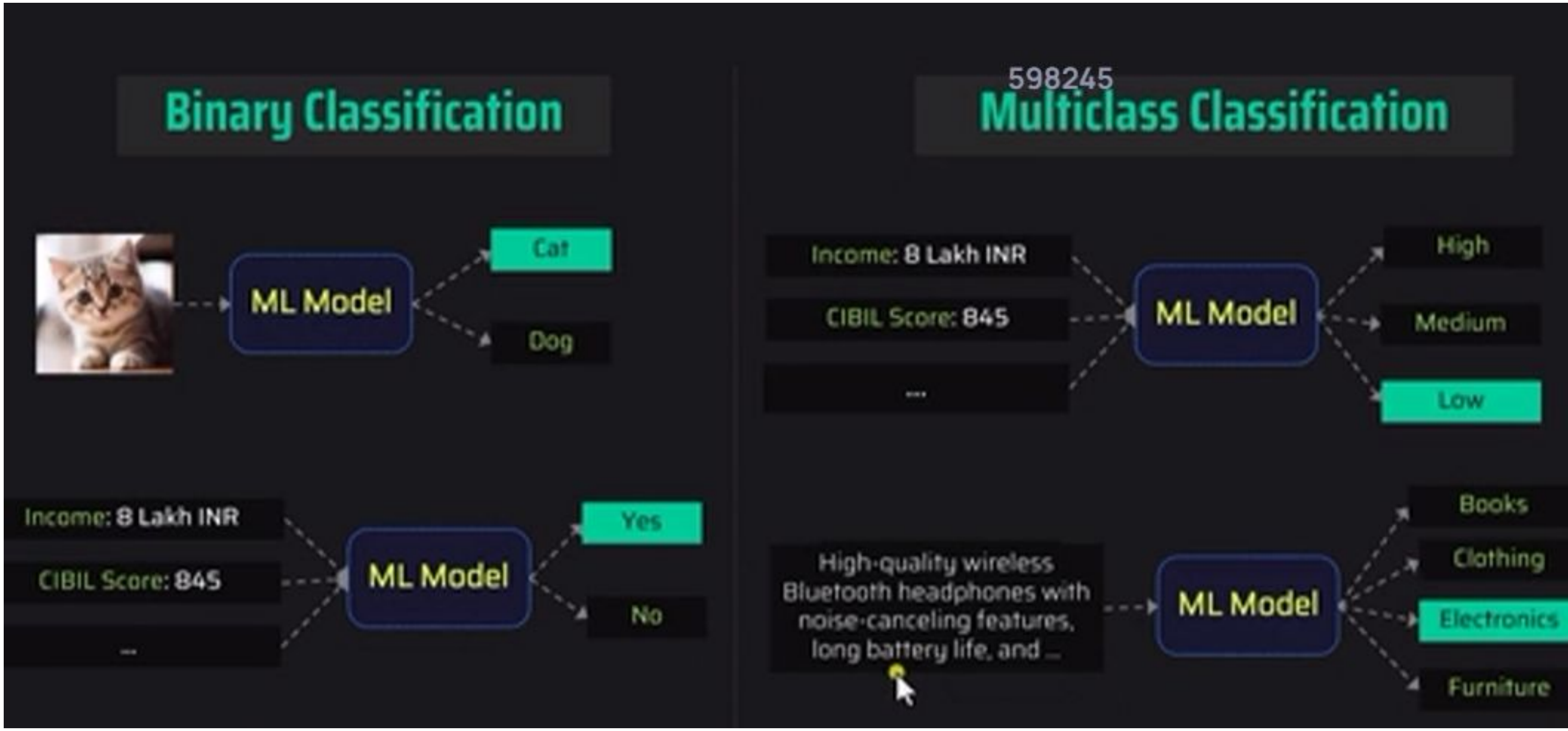
- **Training a model** using input data with known labels to learn the relationship between the input features and the output categories.
- **Predicting categorical outcomes** for new instances by assigning them to predefined classes based on the learned model.
- Utilizing various **classification algorithms** to approximate a mapping function from input variables to output variables.
- This method is widely used in applications such as spam detection, image recognition, and medical diagnosis.

Regression



Area: 1500 sqr ft
Bedrooms: 2
City: Mumbai
ML Model
Price: 2.1 Crore
Mileage: 20 k
Make: Toyota
Model: Camry
ML Model
Price: 18.7 k $
Continuous


Classification
ML Model
Cat
Dog
Income: 8 Lakh INR
CIBIL Score: 845
...
ML Model
Yes
No



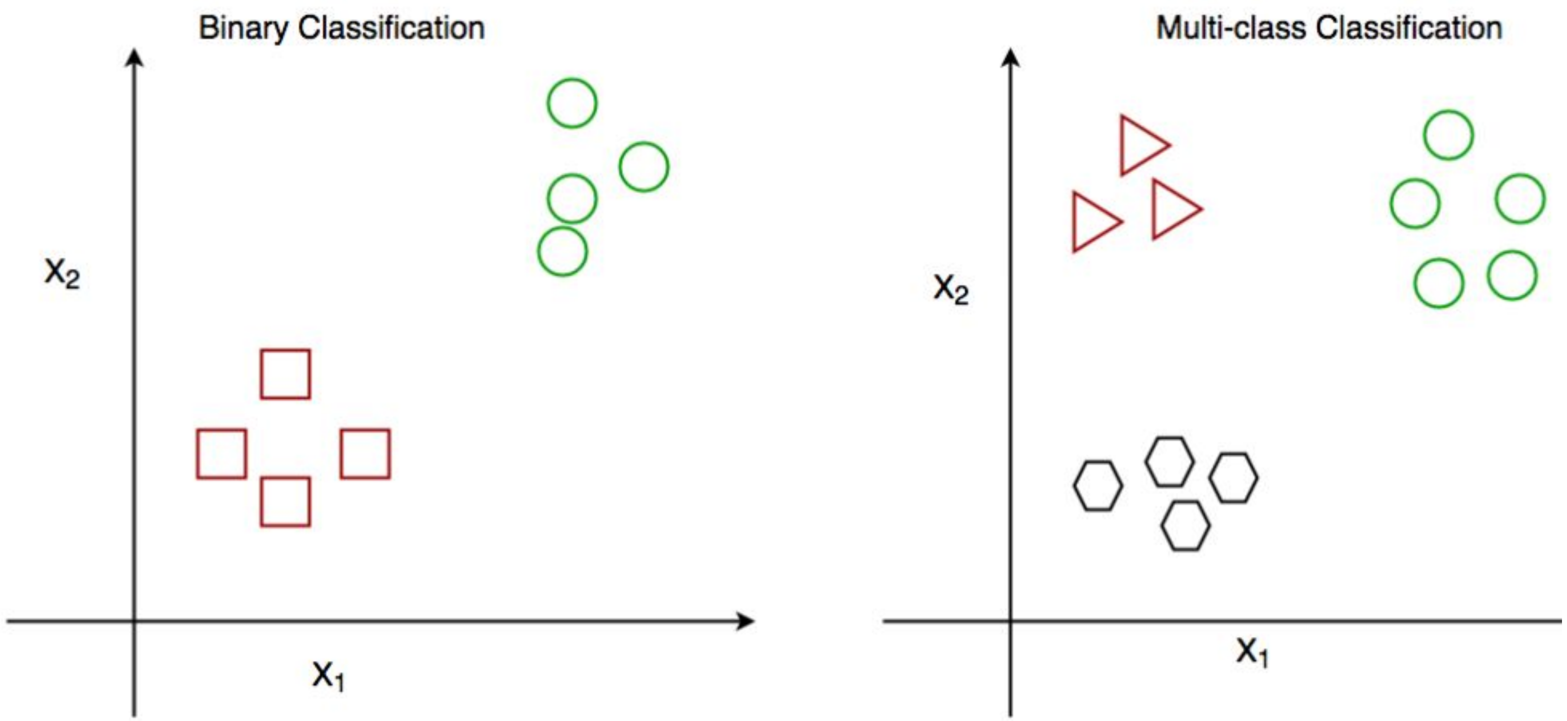
Binary Classification
ML Model
Cat
Dog
Income: 8 Lakh INR
CIBIL Score: 845
...
ML Model
Yes
No
598245
Multiclass Classification
Income: 8 Lakh INR
CIBIL Score: 845
...
ML Model
High
Medium
Low
High-quality wireless Bluetooth headphones with noise-canceling features, long battery life, and ...
ML Model
Books
Clothing
Electronics
Furniture

1. Binary Classification

- This is the simplest kind of classification.
- In binary classification, the goal is to sort the data into **two distinct categories**.
- Imagine a system that sorts emails into either **spam** or **not spam**.
- It works by looking at **different features of the email** like certain keywords or sender details, and decides whether it's spam or not. It only chooses between these two options.

2. Multiclass Classification

- Here, instead of just two categories, the data needs to be sorted into **more than two categories**.
- The model picks the one that best matches the input.
- An image recognition system that sorts pictures of animals into categories like **cat**, **dog**, and **bird**.
- Basically, machine looks at the features in the image (like shape, color, or texture) and chooses which animal the picture is most likely to be based on the training it received.



Binary classification vs Multi class classification

Classification works by training a model to **learn patterns** from labeled data, so it can predict the category or class of new, unseen data.

- **Data Collection**: You start with a dataset where each item is labeled with the correct class (for example, "cat" or "dog").
- **Feature Extraction**: The system identifies features (like color, shape, or texture) that help distinguish one class from another. These features are what the model uses to make predictions.
- **Model Training**: Classification - machine learning algorithm uses the labeled data to learn how to map the features to the correct class. It looks for patterns and relationships in the data.

- **Model Evaluation**: Once the model is trained, it's tested on new, unseen data to check how accurately it can classify the items.
- **Prediction**: After being trained and evaluated, the model can be used to predict the class of new data based on the features it has learned.
- **Model Evaluation**: Evaluating a classification model is a key step in machine learning. It helps us check how well the model performs and how good it is at handling new, unseen data. Depending on the problem and needs we can use different metrics to measure its performance.
- If the quality metric is not satisfactory, the ML algorithm or hyperparameters can be adjusted, and the model is retrained. This iterative process continues until a satisfactory performance is achieved.

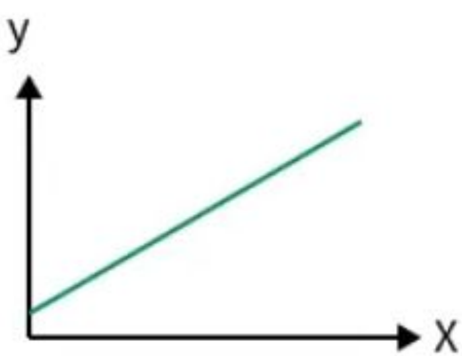
Logistic Regression

- Logistic Regression is a supervised machine learning algorithm used for classification problems.
- Unlike linear regression which predicts continuous values it predicts the probability that an input belongs to a specific class.
- It is used for binary classification where the output can be one of two possible categories such as Yes/No, True/False or 0/1.
- It uses sigmoid function to convert inputs into a probability value between 0 and 1.

Linear Regression vs Logistic Regression

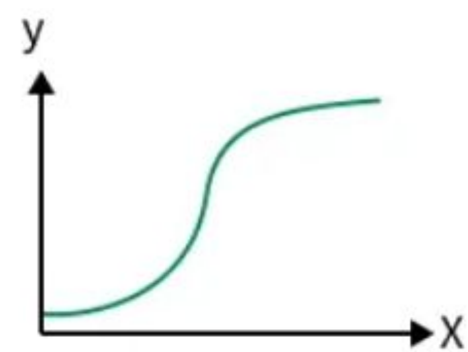
Linear Regression

- Predicts continuous values
- Uses best-fit line
- Solves regression problems



Logistic Regression

- Predicts categorical classes
- Uses sigmoid S-curve
- Solves classification problems



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

e = Euler's number ~ 2.71828

Sigmoid function converts input into range 0 to 1

Sigmoid function

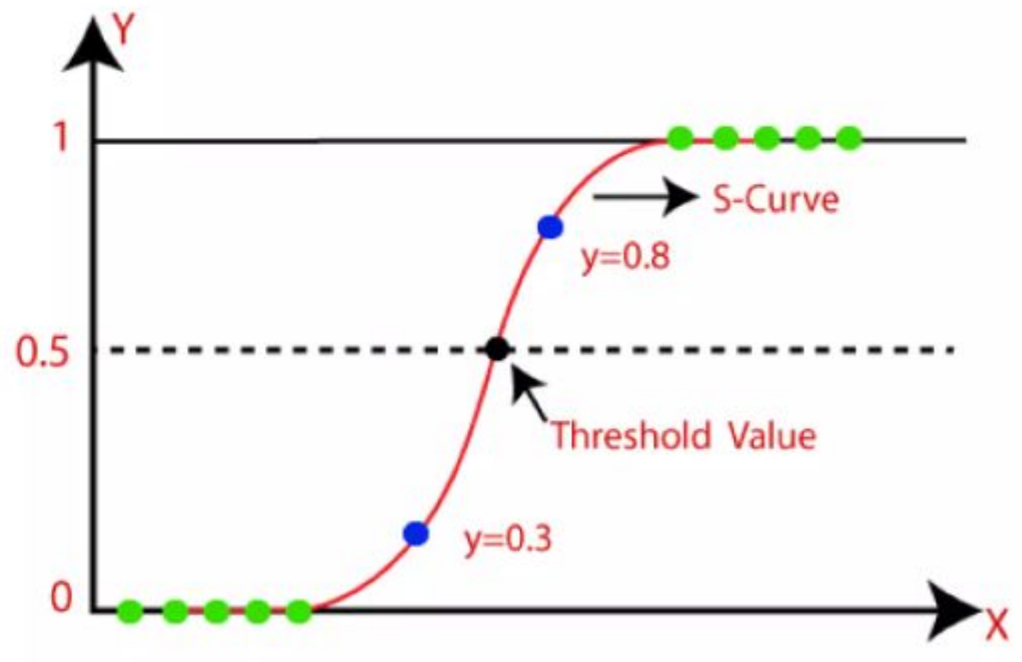
- The sigmoid function is a important part of logistic regression which is used to convert the raw output of the model into a probability value between 0 and 1.
- This function takes any real number and maps it into the range 0 to 1 forming an "S" shaped curve called the sigmoid curve or logistic curve. Because probabilities must lie between 0 and 1, the sigmoid function is perfect for this purpose.
- In logistic regression, we use a threshold value usually 0.5 to decide the class label.
- If the sigmoid output is same or above the threshold, the input is classified as Class 1.
- If it is below the threshold, the input is classified as Class 0.
- This approach helps to transform continuous input values into meaningful class predictions.

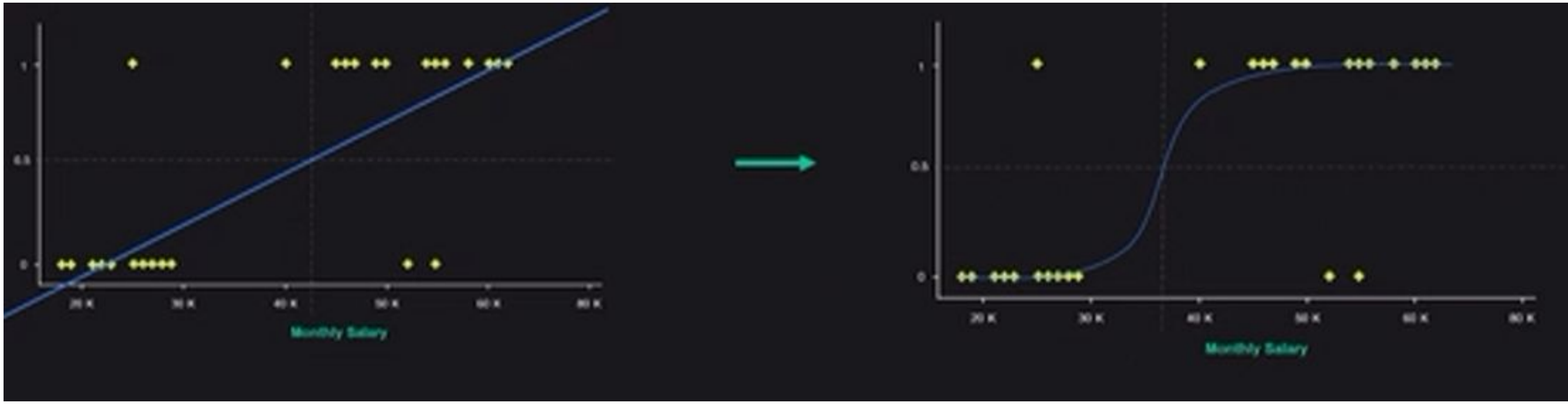
Decision boundary

Threshold classifier output y' *or* $h_\theta(x)$ at 0.5:

If $\boldsymbol{h_\theta(x) \geq 0.5}$, predict "**y = 1**"

If $\boldsymbol{h_\theta(x) < 0.5}$, predict "**y = 0**"





$$z = \beta_0 + \beta_1 * \text{salary}$$

$$z = \beta_0 + \beta_1 x_1$$

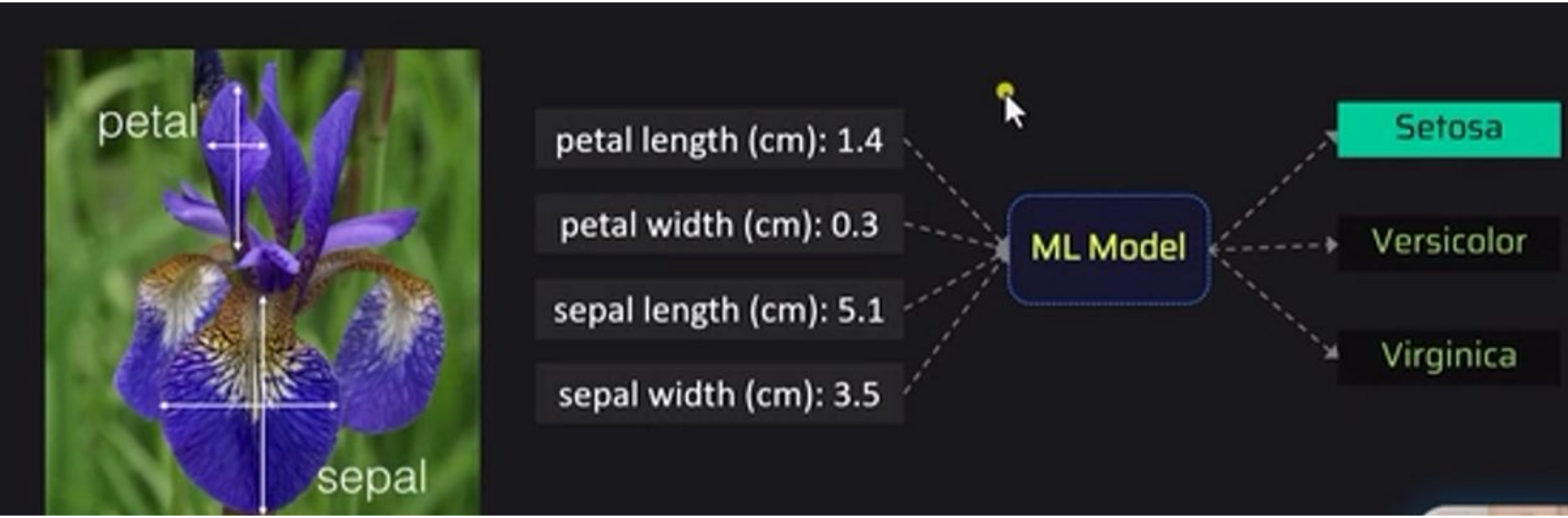
$$y = \frac{1}{1 + e^{-z}}$$

$$y = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1)}}$$

$$P(y = 1 \mid x)$$

- Probability of y to be 1 , when salary= x

Multiclass classification


petal
sepal
petal length (cm): 1.4
petal width (cm): 0.3
sepal length (cm): 5.1
sepal width (cm): 3.5
ML Model
Setosa
Versicolor
Virginica

SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
6.8	3.2	5.9	2.3	Iris-virginica
6.9	3.1	5.1	2.3	Iris-virginica
4.9	3.0	1.4	0.2	Iris-setosa
5.6	3.0	4.5	1.5	Iris-versicolor
4.8	3.1	1.6	0.2	Iris-setosa
5.8	2.8	5.1	2.4	Iris-virginica
7.2	3.6	6.1	2.5	Iris-virginica
5.1	3.5	1.4	0.3	Iris-setosa
4.7	3.2	1.6	0.2	Iris-setosa
6.6	3.0	4.4	1.4	Iris-versicolor

Fig.1: Iris dataset having three categories

Softmax function

- This is a mathematical function that transforms a vector of values into a probability distribution.
- It is commonly used in machine learning, particularly in classification tasks, to convert raw scores (logits) into probabilities.
- In a multi-class classification problem, the softmax function can be used to determine the probability of each class given the input data.

K: no. of possible classes

Z:ith raw score of the model(no. of logits)

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

- Softmax Classifier plays a crucial role in transforming raw model outputs into probabilities.
- It is commonly used in multi-class classification problems where the goal is to assign an input into one of many classes.
- The Softmax function is a mathematical function that converts a vector of real numbers into a probability distribution.
- Each element in the output is between 0 and 1, and the sum of all elements equals 1.
- This property makes it perfect for classification tasks, where we want to know the probability that a given input belongs to a certain class.

1. When we use this?

- When you have a classification problem with more than two classes (say k classes), you can't directly use sigmoid because sigmoid squashes the output into a range (0,1) for a single probability.
  We need a function that:
- Converts multiple raw scores (logits) into probabilities
- Makes sure all probabilities are non-negative and sum to 1
- Allows us to "associate" an input with one of multiple possible classes

2. The Softmax Function

- Given a vector of scores z=[z1,z2,...,zk] for k classes, the **softmax** is:

$$P(y = i \mid \mathbf{z}) = \frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_j}}$$

- Where:
- z_i = raw output (logit) from the model for class i
- e^{z_i} = exponential transformation to make scores positive and preserve ranking
- The denominator is the sum of exponentials for all classes → normalizes the results into probabilities.

3. Why "association"?

- Softmax turns raw outputs into a **probability distribution** over all classes.
  The class with the highest probability is the one the model "associates" the input with.
- For example:
  Suppose the raw logits from a neural network for an image are:

Class	Logit (z)
Cat	2.0
Dog	1.0
Rabbit	0.1

Step 1: Apply exponential

$$e^z = [e^{2.0}, e^{1.0}, e^{0.1}] \approx [7.389, 2.718, 1.105]$$

Step 2: Sum exponentials

$$\text{Sum} \approx 7.389 + 2.718 + 1.105 = 11.212$$

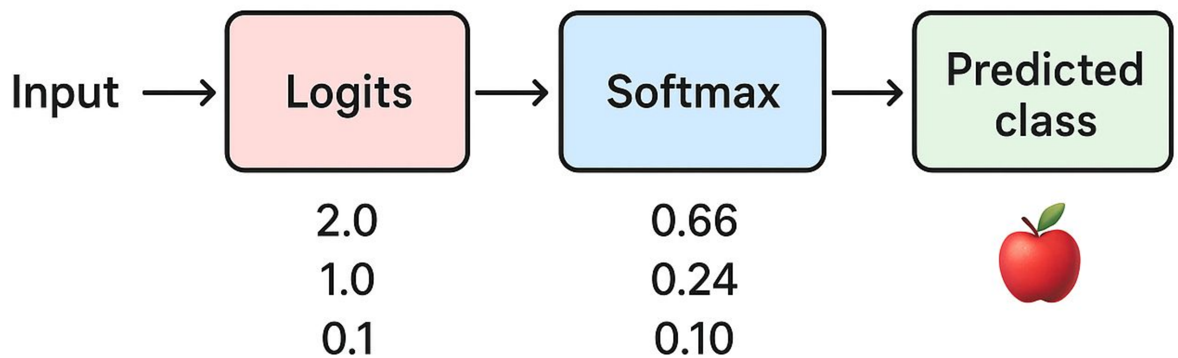
Step 3: Divide each by sum

- Cat: $7.389/11.212 \approx 0.659$
- Dog: $2.718/11.212 \approx 0.242$
- Rabbit: $1.105/11.212 \approx 0.099$

Now we have:

Class	Probability
Cat	65.9%
Dog	24.2%
Rabbit	9.9%

The model **associates** this input with **Cat**, because it has the highest probability.


Input
Logits
Softmax
Predicted class
2.0
1.0
0.1
0.66
0.24
0.10

Algorithms that handle multiclass

- Logistic regression (multinomial)
- Neural networks
- Decision trees / Random forests
- Naïve Bayes
- SVM