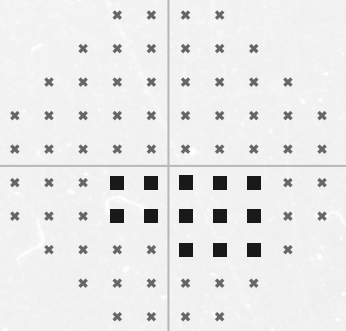
Logistic Regression



01

What is Classification?

What is Classification?

- Classification is a fundamental technique in machine learning and statistics, utilized for categorizing data into distinct classes or groups. (Binary Classification / Multi-class Classification)

- Unlike regression, which predicts continuous outcomes, classification deals with discrete outcomes and focuses on assigning each data point to a specific category based on its features.

- The main objective of classification is to accurately predict the class or category of new or unseen data, using a model trained on a dataset where the categories are known.

02

Why not use linear regression?

Linear Regression

- We can model classification problems by predicting the probability that a data example belongs to a certain class.
- In multi-class classification, we may encounter several problems.
- Imagine encoding animal classes in a scenario where we need to determine, based on animal size, whether the animal is a cat, dog, or elephant.

Dataset: (cat, 5), (cat, 4) , (cat, 6) , (dog, 20), (dog, 30), (dog,25), (elephant, 5400), (elephant, 6000), (elephant, 5000)

$$y = \begin{cases} 1 & Dog \\ 2 & Cat \\ 3 & Elephant \end{cases}$$

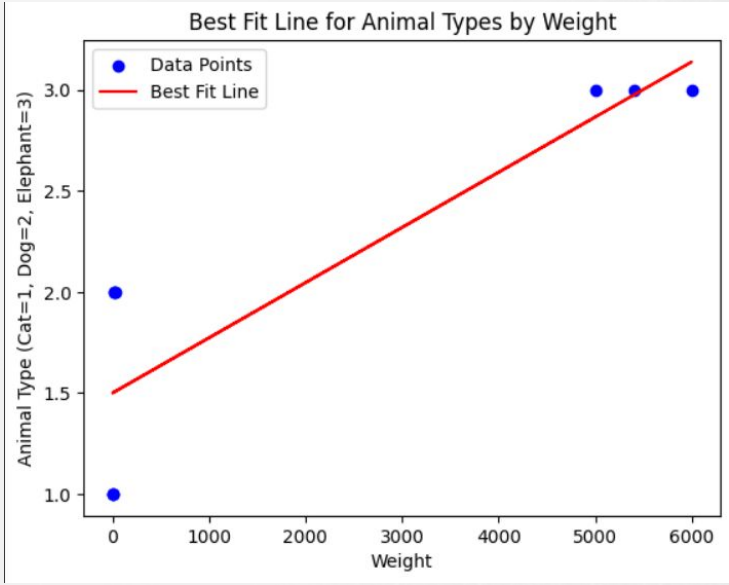
$$y = \begin{cases} 1 & Elephant \\ 2 & Dog \\ 3 & Cat \end{cases}$$

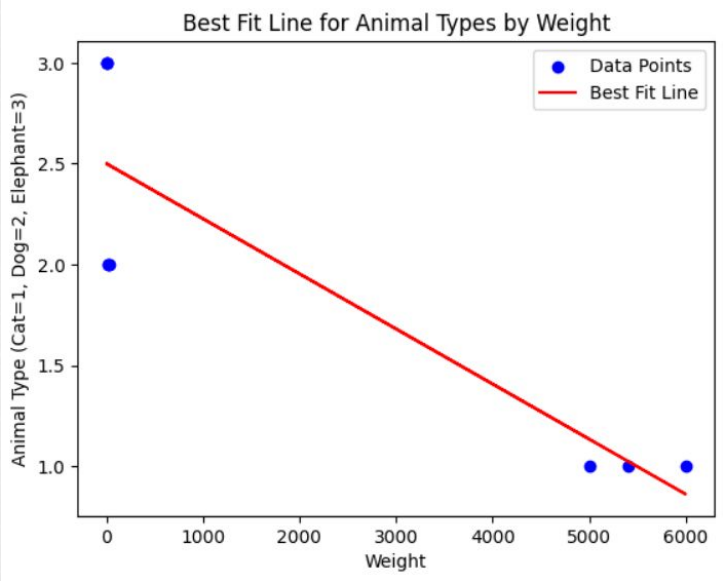
Linear Regression

$$y = \begin{cases} 1 & Cat \\ 2 & Dog \\ 3 & Elephant \end{cases} \qquad y = \begin{cases} 1 & Elephant \\ 2 & Dog \\ 3 & Cat \end{cases}$$

- Problems arise due to two key factors:
  - Different encodings lead to different linear regression models
    - Different order on the axis.
    - Notice different lines in next slide.
  - This approach assumes equal differences between classes.
    - For encoding 1 we assume the same difference between a cat and a dog is the same as the difference between a dog and an elephant
    - In the next slide, note that each class has three data points. However, due to the elephant's large size, the scale is skewed. The dog and cat are similar in size, but the elephant is significantly larger.

Different encodings





$$y = \begin{cases} 1 & Cat \\ 2 & Dog \\ 3 & Elephant \end{cases}$$

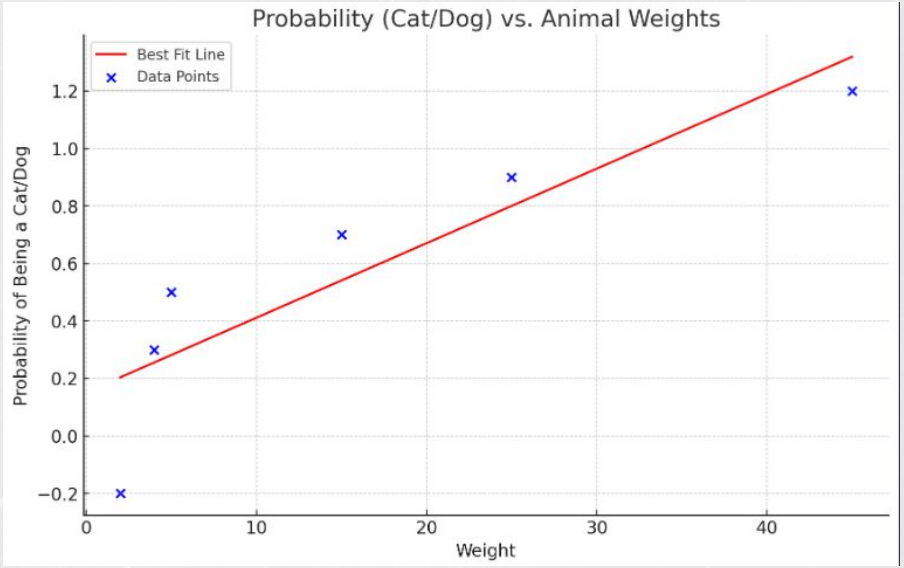
$$y = \begin{cases} 1 & Elephant \\ 2 & Dog \\ 3 & Cat \end{cases}$$

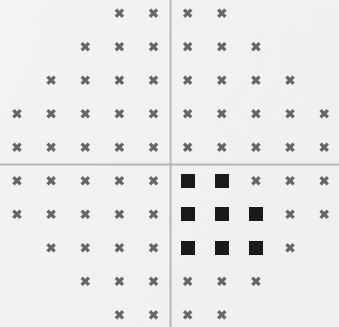
Linear Regression

- n Binary classification, these two restrictions don't apply. We can model it by considering the probability of one class (e.g., a dog) and deducing the probability of the other class (e.g., a cat) as 1 minus the probability of the first class.

- A more subtle problem will appear, because of the linear nature of the model we can have probabilities that either is less than 0 or more than 1

- Observe in the figure that we have -ve probabilities and probabilities greater than 1 !!!



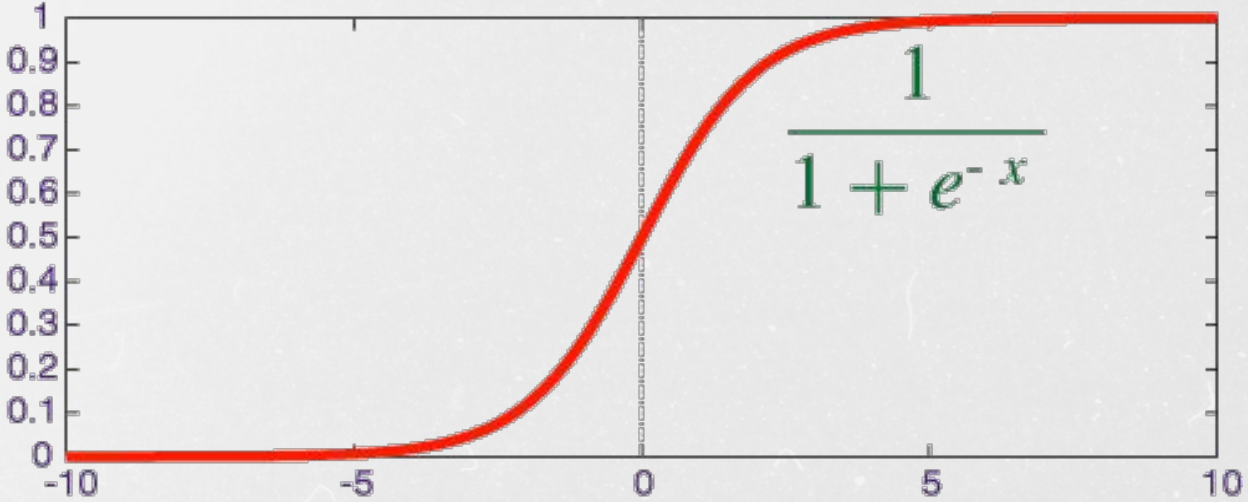


03

Sigmoid

Sigmoid Function

- To avoid the problem of probabilities being -ve and too large we will introduce a function that gives outputs only between 0 and 1
- This function is called the Sigmoid or the Logistic function
- The choice of this function is based on specific reasons, which are beyond the scope of this discussion

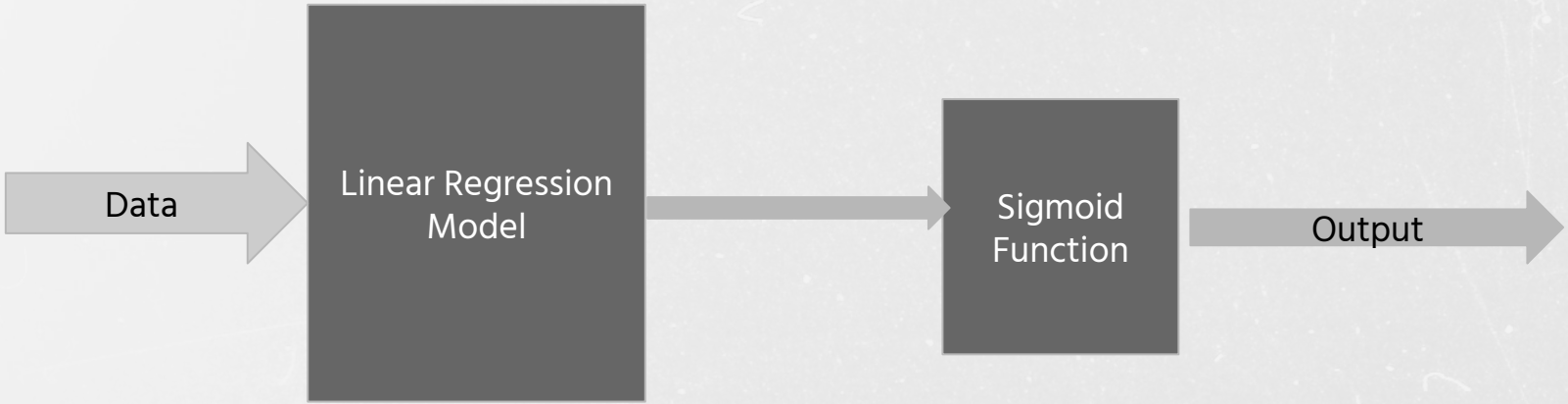


04

Binary Classification with logistic regression

Binary Classification with logistic regression

- Although named logistic regression it is used for classification
- So for binary classification let's connect the things we know about.
- We will predict the probability of 1 class only and if this probability is low then we know that it is the other class.
- A class will have the label of 0 and the other class will have the label of 1



Binary Classification with logistic regression

$$P(Y = Class1 \mid Data) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_1 + ... + \beta_p X_P)}}$$

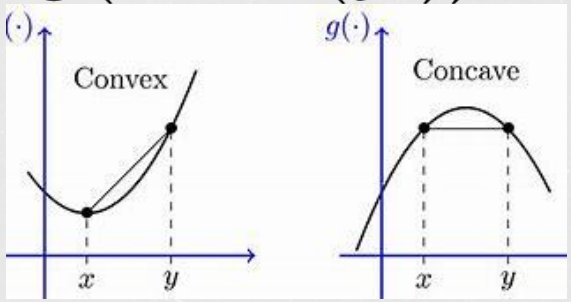
- So notice that
  - We are just passing the linear regression model to our sigmoid function to get the data between 0 & 1
  - Our output can be interpreted as the probability of the label being of a certain class given the data we are training on.
- So know how will we optimize our model?

Optimizing Logistic Regression

- We need to get the W that will maximize the performance of our model.
- But first let's determine our cost function:
  - We cannot use MSE like in linear regression
  - We need to introduce a new cost function.
- Binary Cross Entropy:

$$-\frac{1}{n}\sum_{n} y_n \log\left(P(\hat{y}_n)\right) + (1 - y_n)\log\left(1 - P(\hat{y_n})\right)$$

This function is convex!!!



Binary Cross Entropy

$$-\frac{1}{n}\sum_{n} y_n \log\left(P(\hat{y}_n)\right) + (1 - y_n)\log\left(1 - P(\hat{y_n})\right)$$

Let's Break it down: "remember that log(1) = 0 and log(number < 0) is -ve"

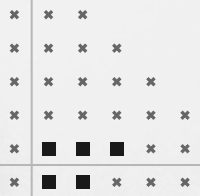
- For the -ve: without the -ve the function is concave so the -ve is just to apply gradient descent and also to cancel the fact that the log in this equation will produce 0 and -ve numbers.
- For the 1 / n : it is just a constant to average the losses
- For the yn and (1 - yn): to handle 2 cases:
  - If the true value is 1 then the (1 - yn) will cancel the second part
  - If the true value is 0 then the yn will cancel the first part
- For the logs:
  - If the true value is 1 so when the probability increases near 1 the loss will be near 0
  - If the true value is 0 so when the probability decrease the 1 - P(yn) will be near 1 so the loss will be near 0

Binary Classification

- So after learning about the Binary Cross Entropy what should we do?

- We need to run our faithful gradient descent to get the global minima of our function

- Is there a direct solution like in linear regression?
  - No, due to the fact our function is not linear so we cannot isolate the w to minimize. “Try the Algebra yourself!!”
  - So we can only use an iterative approach like Gradient descent or Newton's method.



05

Multi-Class Classification with logistic regression

Multi-Class Classification

- This is the generalized form of binary classification

- There are 2 changes
  1) We change the sigmoid function
  2) We change the binary cross entropy

- It is used to predict more than 2 classes “back to our cat, dog, elephant example”

Softmax Function

- For more than 2 classes we use this function instead of the sigmoid function.
- It can be viewed as the generalization of the sigmoid function.

$$P(Y = class \mid data) = \frac{e^{wX}}{\sum_{j=0}^{k} e^{w_j X}}$$

- K is the number of classes
- If we use 2 classes with labels 1 and 0 we will get the sigmoid function!
- It can be interpreted as the probability of the data being a certain class.
- If we sum the probability of all classes the result will be equal 1.
- The class with the highest probability is the one we predict.

Categorical Cross Entropy

- For more than 2 classes we use this cost function instead of binary cross entropy.
- It can be viewed as the generalization of the binary cross entropy.
- For 1 training example:

$$-\sum_{i=0}^{k} y_i \log\left(P(\hat{y}_i)\right)$$

  - It breaks down like the binary cross entropy as yi is 1 only in 1 term of the summation and the log(P) measures how far is the probability from being 1
  - The above is just for 1 example “Loss function” to get the cost function for the whole data set we will sum over the whole data set and divide by N ”the size of the data set”

Sources

- https://www.statlearning.com/
- https://deepai.org/machine-learning-glossary-and-terms/softmax-layer

THANK YOU!