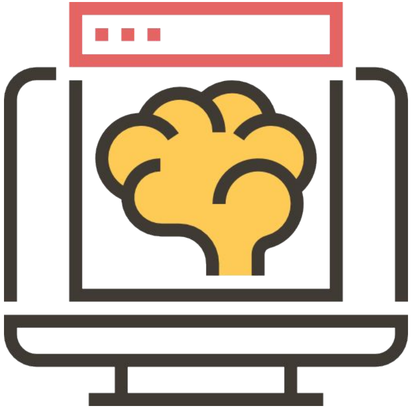
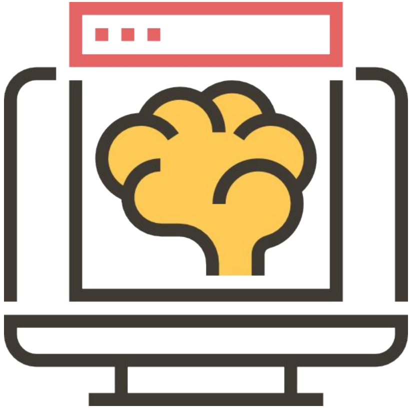


Supervised Learning Classification



Logistic Regression

Agenda

- Introduction to Classification
- Introduction to Logistic Regression
- Difference between Regression and Classification
- Assumption of Logistic Regression
- Types of Logistic Regression
- Implementation of Logistic Regression / Working
- Model Evaluation in Classification

Introduction to Classification

Classification is technique where we categorize data into given number of classes.

The main goal of classification problem is identify the category / class to which a new data will fall under.

1) **Binary classification :-** task with two possible outcome.(T / F)

2) **Multi-class classification :-** more than two classes.

3) **Ordinal Logistic Regression:-**It deals with target variables with ordered categories. For example, a test score can be categorized as: Very poor, Poor, Good, Very good. Here, each category can be given a score like 0, 1, 2, 3.

Introduction To Logistic Regression

- It is a Statistical Machine learning algorithm that comes under Supervised Learning techniques.
- Logistic regression is used to predict the categorical dependent variable with the help of independent variables.
- The output of Logistic Regression problem can be only between the 0 and 1.

Introduction To Logistic Regression

- Logistic regression can be used where the probabilities between two classes is required. Such as whether it will rain today or not, either 0 or 1, true or false etc.
- If estimate probability is greater than 50% then the model predict that the instance belong to that class. (labeled ‘1’ or called the ‘+ve’ class) or else predict that the is does not (i.e. labeled ‘0’ or called the ‘-ve’ class).

Introduction To Logistic Regression

- In logistic regression, we pass the weighted sum of inputs through an activation function that can map values in between 0 and 1. Such activation function is known as **sigmoid function** and the curve obtained is called as sigmoid curve or S-curve.

Difference between Classification & Regressions

CLASSIFICATION	REGRESSION
Is the task of predicting a discrete class label.	Is the task of predicting a continuous quantity
Prediction can be evaluated using accuracy.	Prediction can not be evaluated using accuracy. MSE, RMSE, etc.
Can have real-values or discrete value.	Real value such as integer or floating value.
More than two classes is often called multi-class classification problem.	A problem with multiple i/p variables is often called a multivariate regression problem.

Linear Regression vs Logistic Regression

LINEAR REGRESSION	LOGISTIC REGRESSION
Continuous variable	Categorical or discrete variable
Solve regression problem	Solve classification problem.
Straight line.	S-curve

Logistic Regression

Sigmoid activation

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.

Math

$$S(z)=1/1+e^{\wedge}-z$$

Logistic Regression

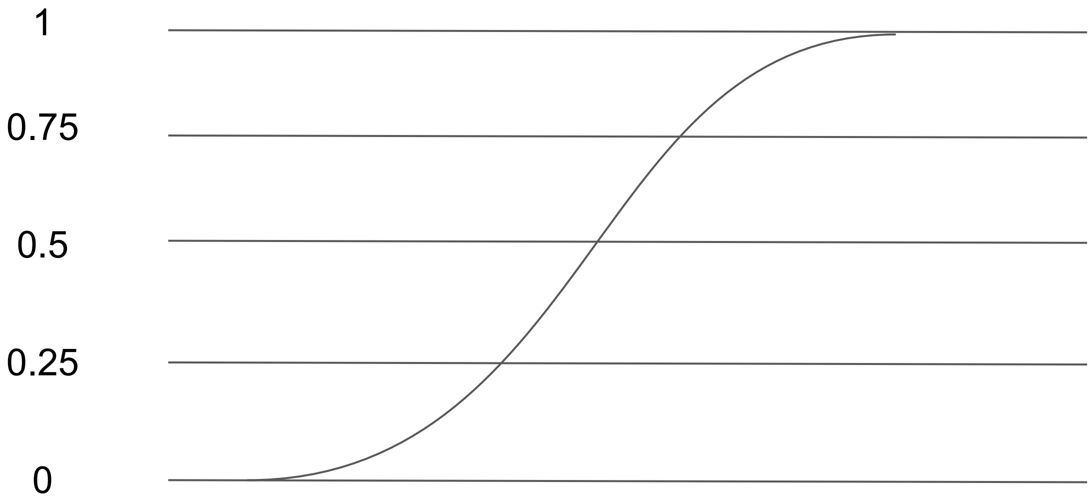
Note :

s(z) = output between 0 and 1 (probability estimate)

z = input to the function (your algorithm's prediction e.g. mx + b)

e = base of natural log

Logit Function



Logistic Regression

Decision boundary

Our current prediction function returns a probability score between 0 and 1.

In order to map this to a discrete class (true/false, cat/dog), we select a threshold value or tipping point above which we will classify values into class 1 and below which we classify values into class 2.

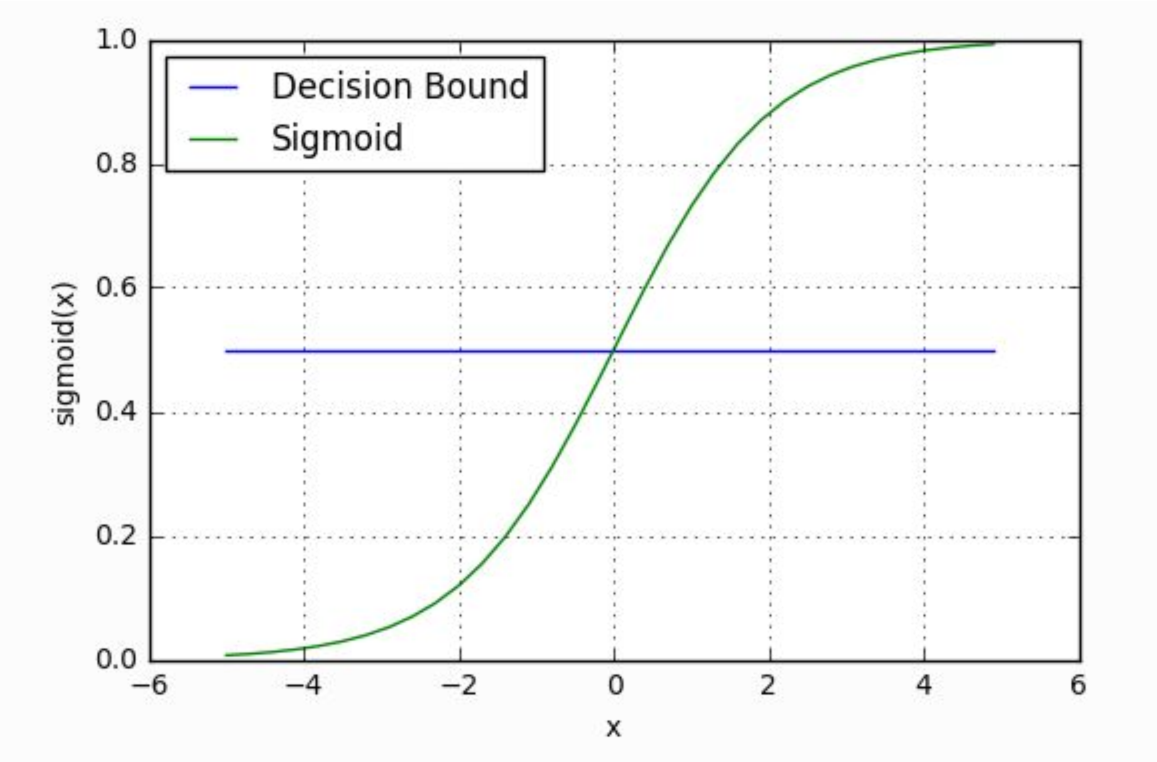
$$p \geq 0.5, class = 1$$

$$p < 0.5, class = 0$$

Logistic Regression

- For example, if our threshold was 0.5 and our prediction function returned 0.7, we would classify this observation as positive.
- If our prediction was 0.2 we would classify the observation as negative.
- For logistic regression with multiple classes we could select the class with the highest predicted probability.

Logistic Regression



Logistic Regression

Making predictions

- Using our knowledge of sigmoid functions and decision boundaries, we can now write a prediction function.
- A prediction function in logistic regression returns the probability of our observation being positive, True, or “Yes”. We call this class 1 and its notation is P(class=1)
- P(class=1). As the probability gets closer to 1, our model is more confident that the observation is in class 1.

Logistic Regression

Math

- Let’s use the same multiple linear regression equation from our linear regression tutorial.

$$Z = W0 + W1Studied + W2Slept$$

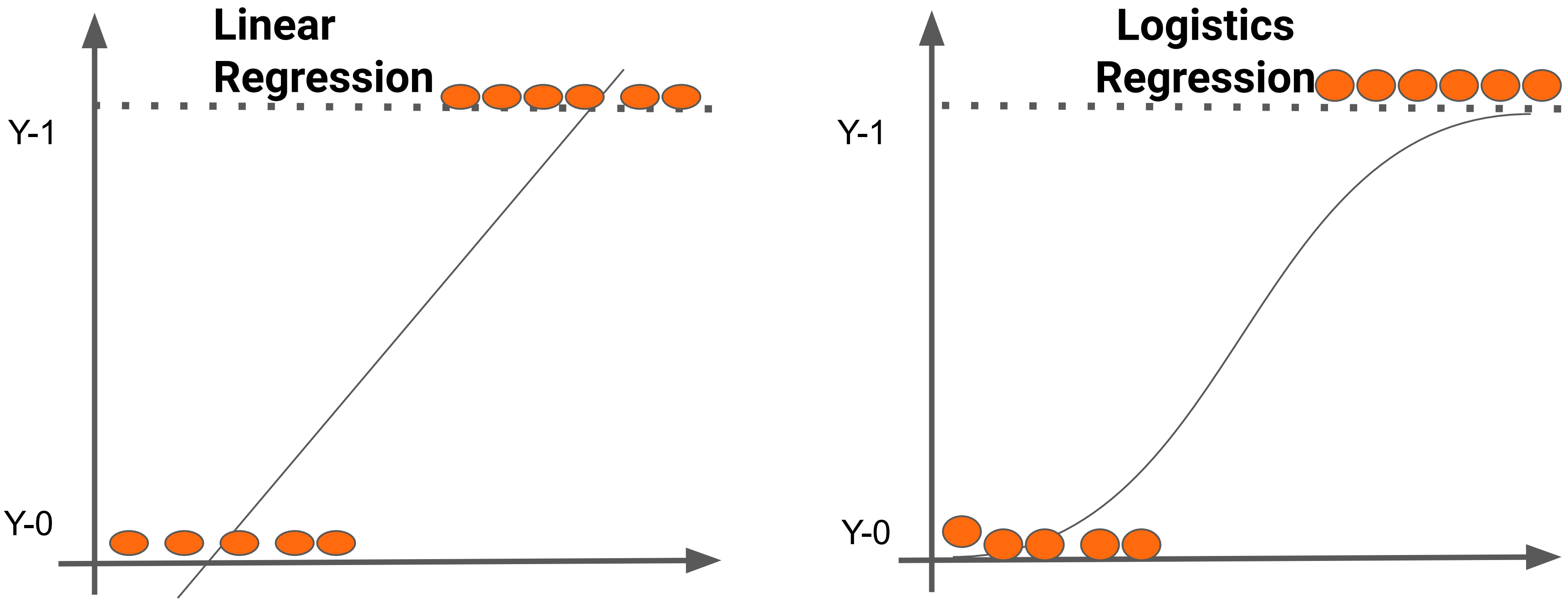
- This time however we will transform the output using the sigmoid function to return a probability value between 0 and 1.

Logistic Regression

P(class=1)=**1/1+e^−z**

- If the model returns 0.4 it believes there is only a 40% chance of passing. If our decision boundary was 0.5, we would categorize this observation as “Fail.”

Comparison of Linear and Logistic Regression



Assumptions of Logistic Regression

- **Binary output variable :-** In a binary logistic regression, the dependent variable must be binary
- **Remove Noise :-** Logistic regression assumes no error in the o/p variable (Y).
- **Gaussian Distribution :-** Logistic Regression is a linear regression. It does assume a linear relationship between the i/p variables with the o/p.

Assumptions of Logistic Regression

- **Remove correlated i/p :-** The model can overfit if you have multiple highly-correlated i/p. Consider calculating the pairwise correlation between all i/p & removing highly correlated i/p.
- Only meaningful variables should be included
- The independent variables are linearly related to the log odds
- Logistic regression requires quite large sample sizes

Model Evaluation in Classification

- Classification accuracy is the ratio of corrected predictions to total predictions mode.
- **Classification accuracy = correct prediction / total predictions**
- It is often presented as a percentage by multiplying 100 But, classification accuracy is that it hides the detail you need to better understand the performance of your classification model

Model Evaluation in Classification

- A confusion matrix is a summary of prediction results on a classification problem.
- The number of correct and incorrect predictions are summarized with count values and broken down by each class.

Model Evaluation in Classification

<u>Below is the process for calculating a confusion matrix.</u>

1. You need a test dataset.
2. Make a prediction for each row in your test dataset.
3. From the expected outcomes and prediction count:
   - The number of correct predictions for each class
   - The number of incorrect predictions for each class, organized by class that was predicted.

Model Evaluation in Classification

- These number are then organized into a table, or a matrix:
  - Each row of the matrix corresponds to a predicted class.
  - Each column of the matrix corresponds to an actual class.

Actual Values

Predicted Values

Predicted Values	TP	FP
	FN	TN

Model Evaluation in Classification

Expected	Predicted
man	woman
man	man
woman	woman
man	man
woman	man
woman	woman
woman	woman
man	man
man	woman
woman	woman

Model Evaluation in Classification

- Total number of correct predictions for a class go into the expected row for that class value and the predicted column for that class value.
- accuracy = total correct predictions / total predictions made * 100

accuracy = 7 / 10 * 100

- Let’s turn our results into a confusion matrix.
- First,we must calculate the number of correct predictions for each class.
    - man classified as man: 3
    - woman classified as woman : 4

Model Evaluation in Classification

- Now, we calculate the number of incorrect predictions for each class, organized by the predicted value.
    - man classified as woman: 2
    - woman classified as man : 1

Now, arrange these values into the 2-class confusion matrix

	man	woman
man	3	1
woman	2	4

- The correct values are organized in a diagonal line (3+4)

Model Evaluation in Classification

- In this way, we can assign the event row as “positive” and the no-event row as “negative”.
- We can then assign the event predictions as ‘T’ and the no-event as ‘F’.
- Now, arrange these values into the 2-class confusion matrix

	event	no-event
event	TP	FP
no-event	FN	TN

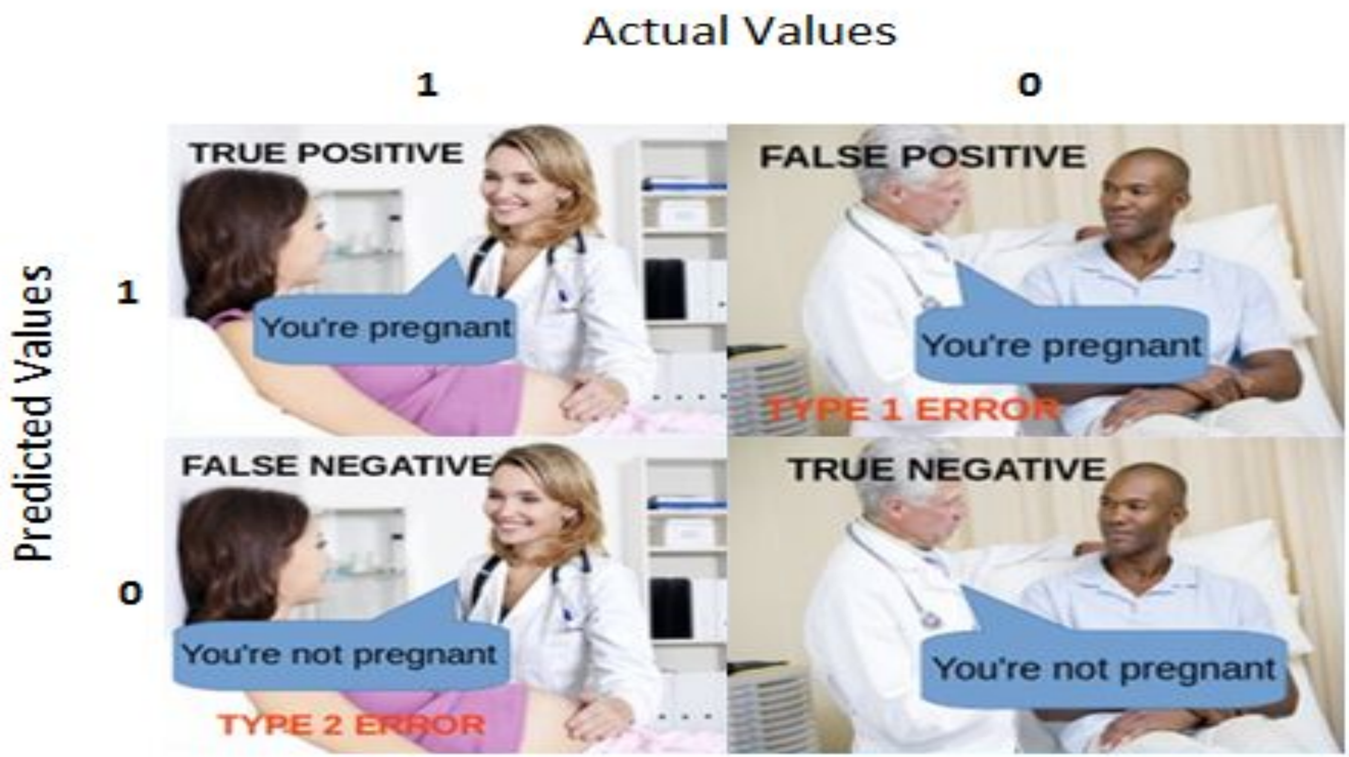
Model Evaluation in Classification

This gives us:

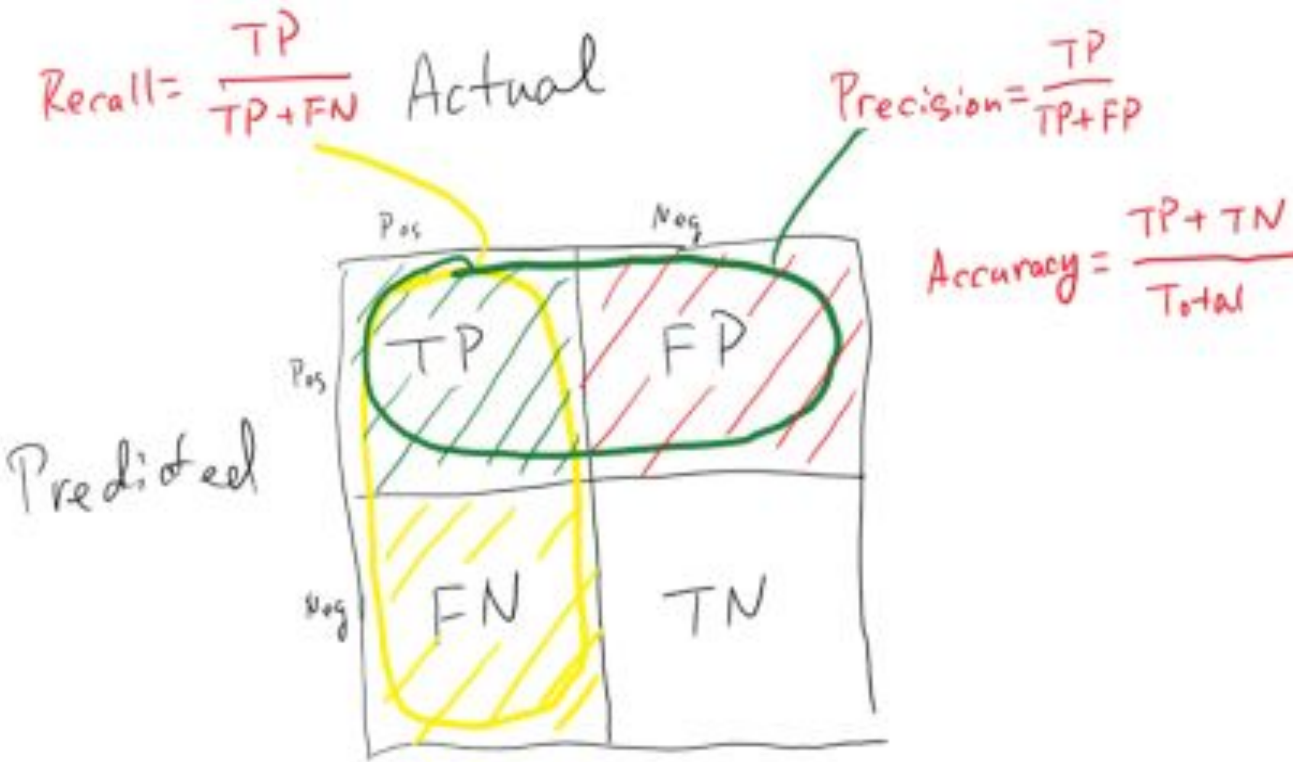
- **‘True Positive’**: for correctly predicted event values.
- **‘False Positive’**: for incorrectly predicted event values.
- **‘True Negative’**: for correctly predicted no-event values.
- **‘False Negative’**: for incorrectly predicted no-event values.

It is extremely useful for measuring Recall, Precision, Specificity, Accuracy, and most useful AUC-ROC curve.

Model Evaluation in Classification



Model Evaluation in Classification



Picture Source :- Google

Model Evaluation in Classification

F-Measure provides a way to combine both precision and recall into a single measure that captures both properties.

F - measure = 2 * (Recall * Precision) / (Recall + Precision)

Model Evaluation in Classification

Sensitivity / True Positive Rate / Recall

$$Sensitivity = \frac{TP}{TP + FN}$$

- Sensitivity tells us what proportion of the positive class got correctly classified.

Model Evaluation in Classification

False Negative Rate

$$FNR = \frac{FN}{TP + FN}$$

- False Negative Rate (FNR) tells us what proportion of the positive class got incorrectly classified by the classifier.

Model Evaluation in Classification

Specificity / True Negative Rate

$$Specificity = \frac{TN}{TN + FP}$$

- Specificity tells us what proportion of the negative class got correctly classified.

Model Evaluation in Classification

False Positive Rate

$$FPR = \frac{FP}{TN + FP} = 1 - Specificity$$

FPR tells us what proportion of the negative class got incorrectly classified by the classifier.

The End