Logistic Regression

Linear Reg. → Predict quantitative variable.

Sometimes, we may need to predict a discrete variable. For eg., a model can predict whether a person is male or female based on their weight.

This type of prediction where we are bothered about the output label and not the exact value is called classification problem.

Logistic Regression is a binary classification algorithm used when the response variable is dichotomous (1 or 0).

The output variable y_i is thus a realization of a random variable Y_i that can take the values 1 and 0 with probabilities p_i and $1 - p_i$, respectively.

Examples of binary classification model -

- Spam detection
- Credit card fraud detection
- Cancer detection

In logistic regression, we get a probability score that reflects the probability of occurrence of the event, in contrast to linear regression which gives the actual predicted output.

$$\log\left(\frac{p}{1-p}\right) = a + b_1 x_1 + b_2 x_2 + \cdots + b_n x_n$$

Odds →

prob. → way to quantify the chances that an event will occur.

But, there are other ways also to represent the chances of the event occurring. one of which is **Odds**.

The odds of an event occurring is the ratio of the expected no. of times that that the event will occur to the expected no of times it will not occur.

$$O = \frac{p}{1-p} = \frac{\text{prob. of event}}{\text{prob. of no event.}}$$

Building logistic Regression Model → (Logit function)

Transforming prob. to odds remove the upper bound. If we then take the log of odds, we also remove the lower bound.

Thus, logistic model frame -

$$\log\left[\frac{p_i}{1-p_i}\right] = \alpha + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_K x_{iK} \quad \text{--- (1)}$$

The expression $\log\left[\frac{p_i}{1-p_i}\right]$ is called logit func".

we can solve logit eq" for p_i to obtain expression

$$p_i = \frac{\exp(\alpha + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_K x_{iK})}{1 + \exp(\alpha + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_K x_{iK})} \quad (2)$$

Eq^n (2) can be simplified by dividing both numerator & denominator by numerator itself.

$$p_i = \frac{1}{1 + \exp(-\alpha - \beta_1 x_{i1} - \beta_2 x_{i2} - \cdots - \beta_K x_{iK})} \quad (3)$$

This func^n is the logistic regression func^n. It is a nonlinear func^n as shown.

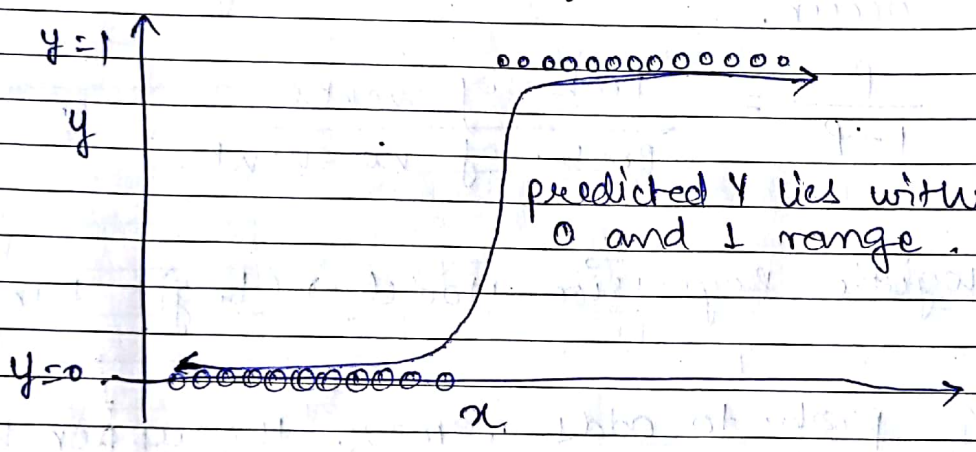

Fig: Logistic regression func^n used in classification.

Maximum Likelihood Estimation →

In Linear Regression → we used method of least square to estimate regression coefficients.

In Logistic Regression → we use maximum likelihood estimation.

Maximum likelihood estimate of a parameter is the value that maximize the prob. of the observed data.

Example of logistic Regression →

Predict whether a customer is a loan non-defaulter or not based on the amount of yearly savings.

Amount in savings (Lakh)	Loan Non-Defaulter	fitted Value	Prediction
0.5	0	0.034710025	0
0.75	0	0.049771971	0
1.00	0	0.070889852	0
1.25	0	0.100024715	0
1.5	0	0.139337907	0
1.75	0	0.190826302	0
1.75	1	0.190826302	0
2.00	0	0.255688447	0
2.25	1	0.333510508	0
2.5	0	0.421602115	0
2.75	1	0.514983013	1
3.00	0	0.607329347	1
3.25	1	0.692588758	1
3.50	0	0.766454783	1
4.00	1	0.874429026	1
4.25	1	0.910262967	1
4.5	1	0.936612324	1
4.75	1	0.955602124	1
5.00	1	0.969090667	1
5.5	1	0.985190994	1

logistic regression analysis using maximum likelihood estimate gives the following o/p.

$\beta_0 = -4.07778$

$\beta_1 = 1.5046$

These coefficients are entered into the logistic regression eq^n to estimate the probability of being a loan - non defaulter!

$$\text{prob. of being non-defaulter} = \frac{1}{1 + \exp(-\beta_0 - \beta_1 x)}$$

$$= \frac{1}{1 + \exp(4.0777 - 1.5046 * \text{saving})}$$

If fitted value > 0.5, then o/p = 1 else o/p = 0.

Confusion Matrix -

	Predicted 0	Predicted 1
Actual 0	True Negative (TN)	False Positive (FP)
Actual 1	False Negative (FN)	True Position (TP)

$$\text{Accuracy} = \frac{TP + TN}{N}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$ { what % of tuples that the classifier labled as positive are actually positive }

(positive predictive value)

21

$$\text{Negative Predictive Value} = \frac{TN}{TN + FN}$$

$$\text{Recall / Sensitivity (True Positive Recognition Rate)} = \frac{TP}{TP + FN}$$

Specifivity = True negative recog. rate

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

for our Eq:

confusion Matrix:

	Predicted 0	Predicted 1
Actual 0	8 (TN)	2 (FP)
Actual 1	2 (FN)	8 (TP)

$$\text{Accuracy} = \frac{TP + TN}{N} = \frac{8+8}{20} = \frac{16}{20} = 0.8 = 80\%$$

$$\text{Precision} = \frac{TP}{TP + FP} = \frac{8}{8+2} = \frac{8}{10} = 0.8$$

$$\text{Neg. Predictive Rate} = \frac{TN}{TN + FN} = \frac{8}{10} = 0.8$$

$$\text{Sensitivity} = \frac{TP}{TP + FN} = \frac{8}{10} = 0.8$$

$$\text{Specifinity} = \frac{TN}{TN + FP} = \frac{8}{10} = 0.8$$