Logit model

- The binary output for new samples can now be easily predicted using the following

$$p(x) = \frac{e^{(\beta_0+\beta_1 X)}}{1 + e^{(\beta_0+\beta_1 X)}}$$

- If $\beta_0 + \beta_1 X$ is non-negative then we get p>0.5 and Y=1 otherwise we get p<0.5 and Y=0
- Decision boundary is the equation $\beta_0 + \beta_1 X$

Example 1

x_1	x_2
1	1
2	1
3	1
4	1
5	1
1	2
2	2
3	2
4	2
5	2

Class 0

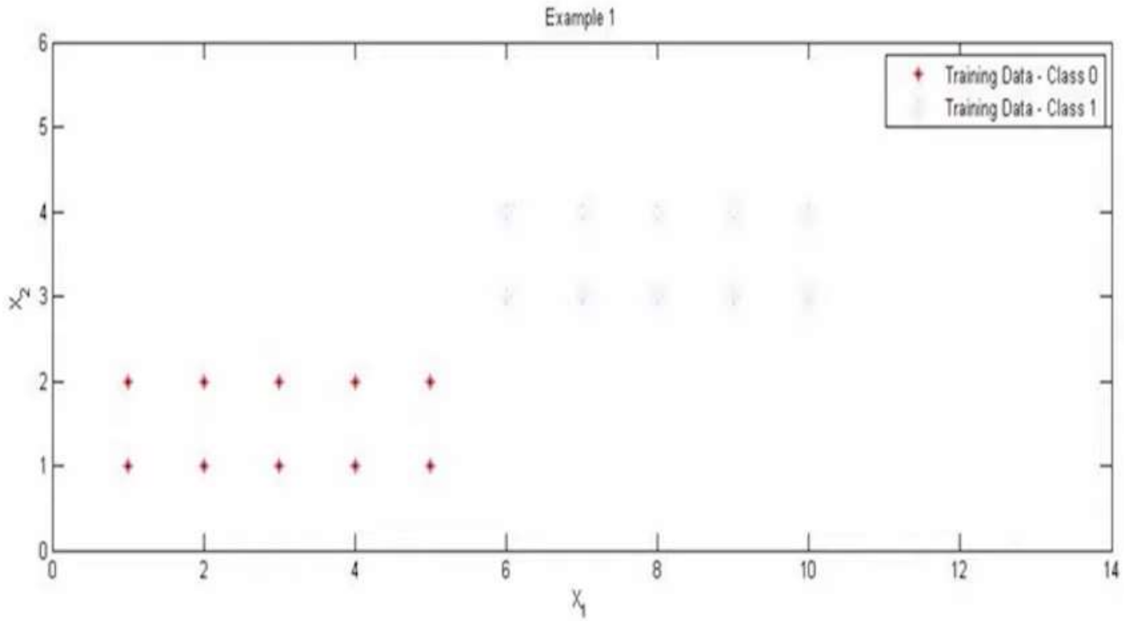
x_1	x_2
6	3
7	3
8	3
9	3
10	3
6	4
7	4
8	4
9	4
10	4

Class 1

x_1	x_2	Class ?
1	3	?
2	3	?
4	4	?
5	4	?
3	3	?
6	2	?
9	2	?
8	1	?
7	2	?
10	1	?

Test Data

Example 1 continued



Results

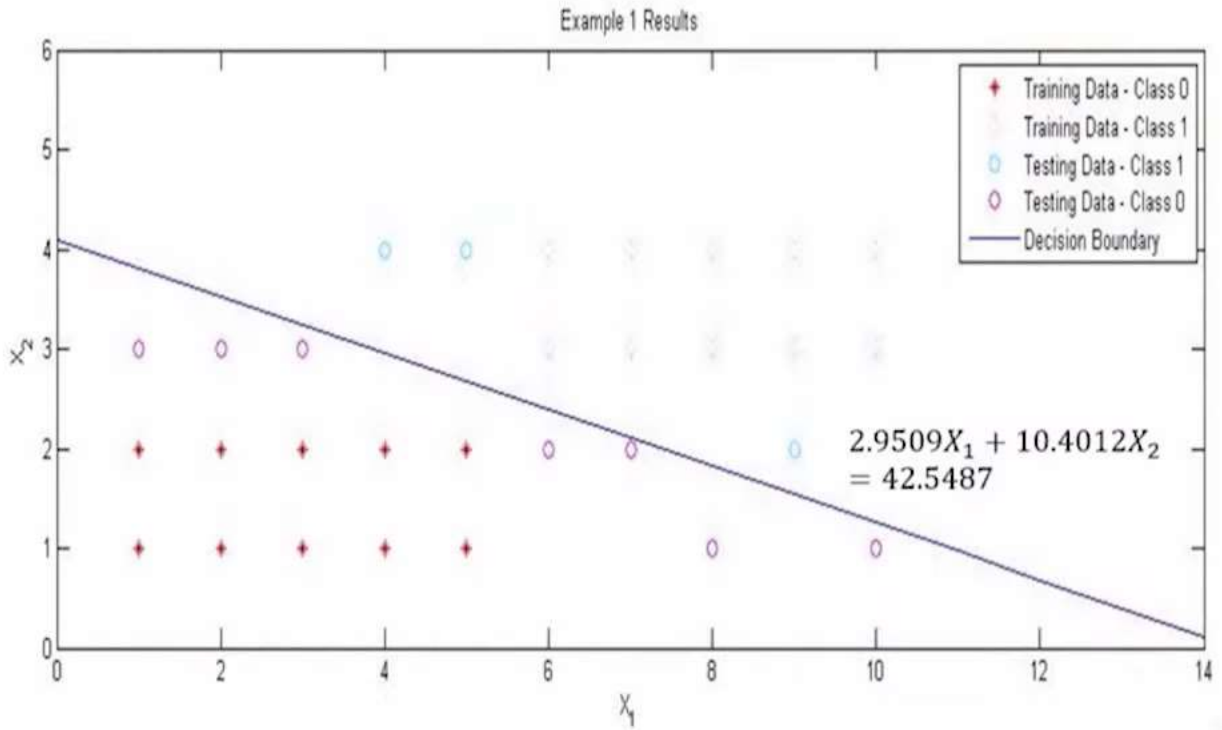
- Input Features : X_1, X_2
- Classes : 0, 1
- Parameters:

$$\beta_0 = -42.5487$$
$$\beta_{11} = 2.9509$$
$$\beta_{12} = 10.4012$$

Test Results

X_1	X_2	Prob	Class
1	3	0.0002	0
2	3	0.004	0
4	4	0.999	1
5	4	0.999	1
3	3	0.076	0
6	2	0.0172	0
9	2	0.991	1
8	1	0.0002	0
7	2	0.251	0
10	1	0.0667	0

Example 1 solution



Regularization

- General objective
  - $\min_{\theta} -L(\theta)$
  - where $L(\theta) = \left(\sum_{i=1}^{n} y_i \log\big(p(x_i)\big) + (1 - y_i) \log\big(1 - p(x_i)\big)\right)$
- When large number of independent variables are present, logistic regression tends to over-fit
- To prevent over-fitting, we need to penalize the coefficients
- This is known as regularization

Regularization

- Regularization helps in building non-complex models that avoids capturing noise in model due to over-fitting
- The objective now becomes
  - $\min\limits_{\theta} -L(\theta) + \lambda * h(\theta)$ where λ is regularization parameter and $h(\theta)$ is regularization function
- Depending on $h(\theta)$, the regularization can be classified as L_1 or L_2 type
- $h(\theta) = \theta^T\theta$ for L_2 type regularization
- Larger the value of λ, more is the regularization strength
- Regularization helps the model work better on test data due to the fact that over-fitting is minimized on training data