09/06/2020

Logistic Regression [IBM]

1. What is Logistic Regression?

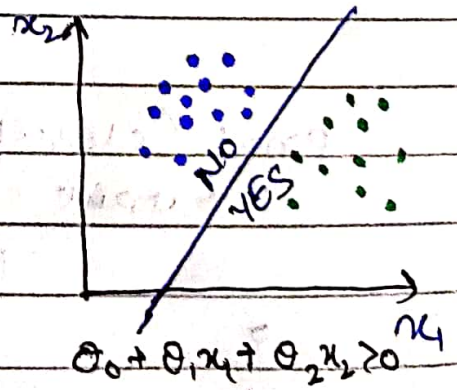
- Logistic Regression is a classification algorithm for categorical variables.

Logistic Regression Applications

- Predicting the probability of a person having heart attack.
- Predicting the mortality in injured patients.
- Predicting a customer's propensity to purchase a product or halt a subscription.
- Predicting the probability of failure of a given process or product.
- Predicting the likelihood of a homeowner defaulting on a mortgage.

When is Logistic Regression suitable?

- If your data is binary
  → 0/1, YES/NO, TRUE/FALSE
- If you need Probabilistic results
- When you need a linear Decision Boundary
- If you need to understand the impact of a feature.

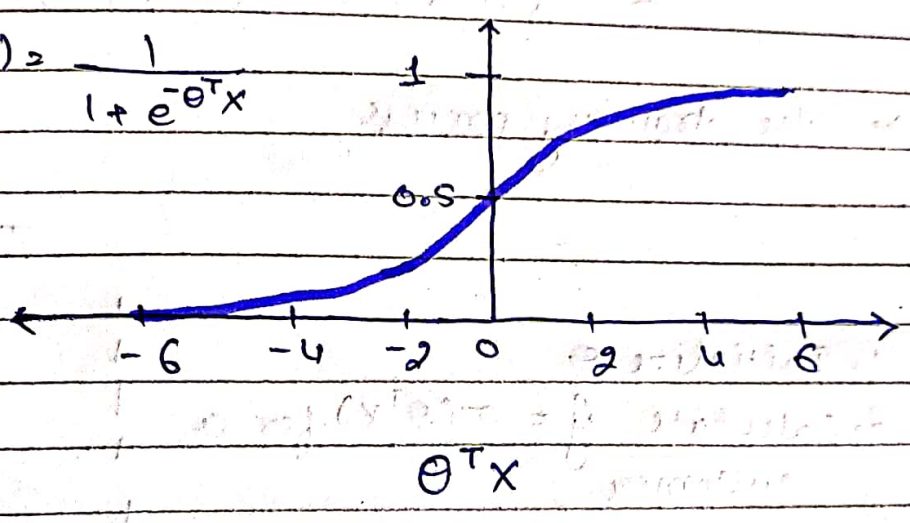


- Independent feature, $X \in R^{m \times n}$
  where, $m \rightarrow$ features
  $n \rightarrow$ records
- Dependent variable, $y \in \{0, 1\}$
- $\hat{y} = P(y=1|x)$

Sigmoid function in Logistic Regression

- Logistic function

$$\sigma(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}}$$



- If $\theta^T x$ is bigger than $\sigma(\theta^T x)$ will be equivalent to 1.
- If $\theta^T x$ is smaller than $\sigma(\theta^T x)$ will be equivalent to 0.
- So, the value of sigmoid func^n will be in b/w $[0, 1]$

Classification of the customer churn model

- what is the o/p of our model?
  - → $P(Y=1|x)$
  - → $P(Y=0|x) = 1 - P(Y=1|x)$

→ $P(\text{churn}=1 | \text{income, age}) = 0.8$
→ $P(\text{churn}=0 | \text{income, age}) = 1-0.8 = 0.2$

$\sigma(\theta^T x) \rightarrow P(Y=1|x)$

$1 - \sigma(\theta^T x) \rightarrow P(Y=0|x)$

The training process

$\sigma(\theta^T x) \rightarrow P(Y=1|x)$

eg

1. Initialize θ	$\theta = [-1, 2]$
2. Calculate $\hat{y} = \sigma(\theta^T x)$ for a customer	$\hat{y} = \sigma([-1,2] \times [2,5])$ $= 0.7$
3. Compare the output of $\hat{y}$ with the actual o/p of customer, y, and record it as error.	Error $= 1 - 0.7 = 0.3$
4. Calculate the Error for all customers.	Cost $= J(\theta)$
5. change the θ to reduce the cost	θ_{new}
6. Go back to Step 2.	

Logistic Regression Training

General Cost function

- $\sigma(\theta^T x) \rightarrow P(Y=1|x)$
- change the weight → Reduce the cost
- Cost function

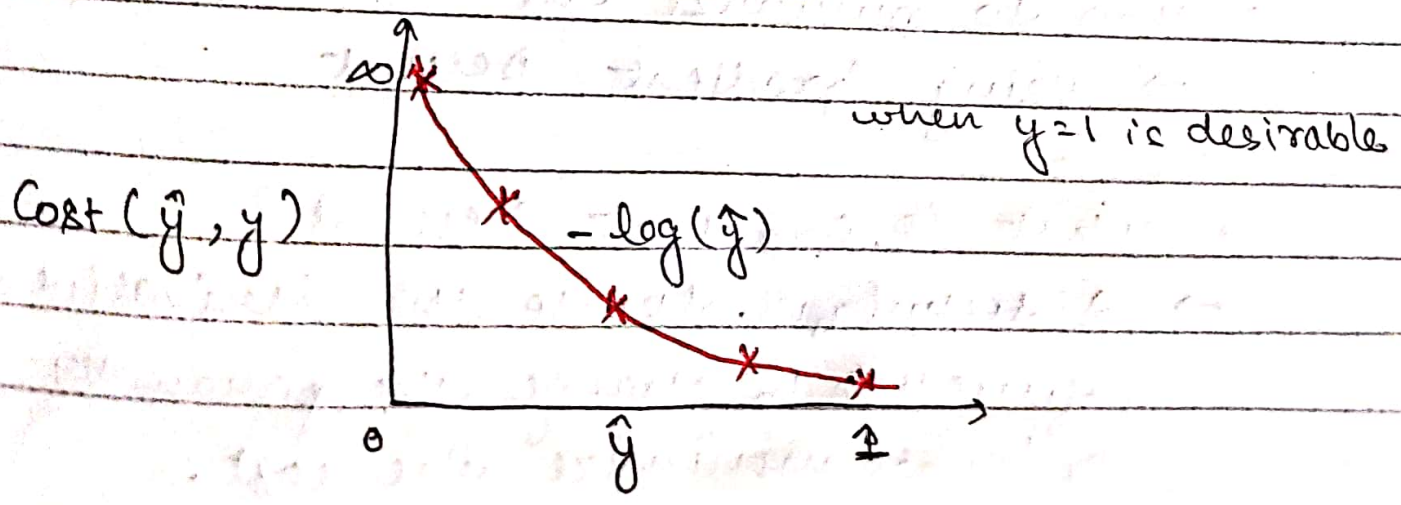
$$cost(\hat{y}, y) = \frac{1}{2}(\sigma(\theta^T x) - y)^2$$

($\hat{y}$: Predicted value of the target; y: Actual value of the target)

$$J(\theta) = \frac{1}{m} \sum_{i=1}^{n} cost(\hat{y}, y)$$

Plotting the cost function of the model

- Model $\hat{y}$
- Actual value $y=1$ or 0
- If $y=1$, & $\hat{y}=1 \rightarrow cost=0$
- If $y=1$, & $\hat{y}=0 \rightarrow cost=large$



Logistic Regression Cost function

- So, we will replace cost function with:

$$\rightarrow \text{cost}(\hat{y}, y) = \frac{1}{2}(\sigma(\theta^T x) - y)^2$$

$$\Downarrow$$

$$\text{cost}(\hat{y}, y) = \begin{cases} -\log(\hat{y}) & \text{if } y=1 \\ -\log(1-\hat{y}) & \text{if } y=0 \end{cases}$$

$$\rightarrow J(\theta) = \frac{1}{m} \sum_{i=1}^{m} \text{cost}(\hat{y}, y)$$

$$\Downarrow$$

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1-y^i) \log(1-\hat{y}^i)$$

Minimizing the cost function of the model:

- How to minimize the best parameters of our model?
  → Minimize the cost function

- How to minimize the cost function?
  → Using Gradient Descent

- what is Gradient Descent?
  → A technique to use the derivative of a cost function to change the parameter values, in order to minimize the cost.

Logistic Regression and the ROC curve [DATACAMP]

Logistic Regression for binary classification

- Logistic Regression outputs probabilities
- If the probability 'p' is greater than 0.5:
  → The data is labelled '1'
- If the probability 'p' is less than 0.5:
  → The data is labelled '0'

- Logistic Regression helps to create Linear Decision Boundary

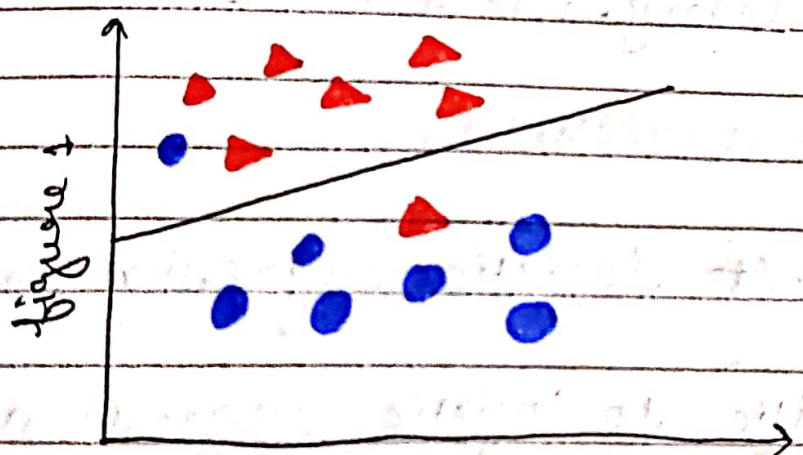


figure 0

Logistic Regression

Logistic Regression in Scikit-learn

```
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split

logreg = LogisticRegression()

X_train, X_test, y_train, y_test =
    train_test_split(X, y, test_size=0.4,
                     random_state=42)

logreg.fit(X_train, y_train)
y_pred = logreg.predict(X_test)
```

Probability thresholds

- By default logistic regression threshold = 0.5
- Not specific to logistic regression
  → KNN classifier also have thresholds

→ What is ROC curve? (TOWARDS DATA SCIENCE)

- Reciever Operating Characteristics curve
- ROC curve is a performance measurement for classification problem at various threshold settings.
- ROC is a probability curve.

- The ROC curve is plotted with TPR against the FPR where TPR is on y-axis & FPR is on x-axis.

where,

↳ (TPR : TRUE POSITIVE RATE) / RECALL / SENSITIVITY

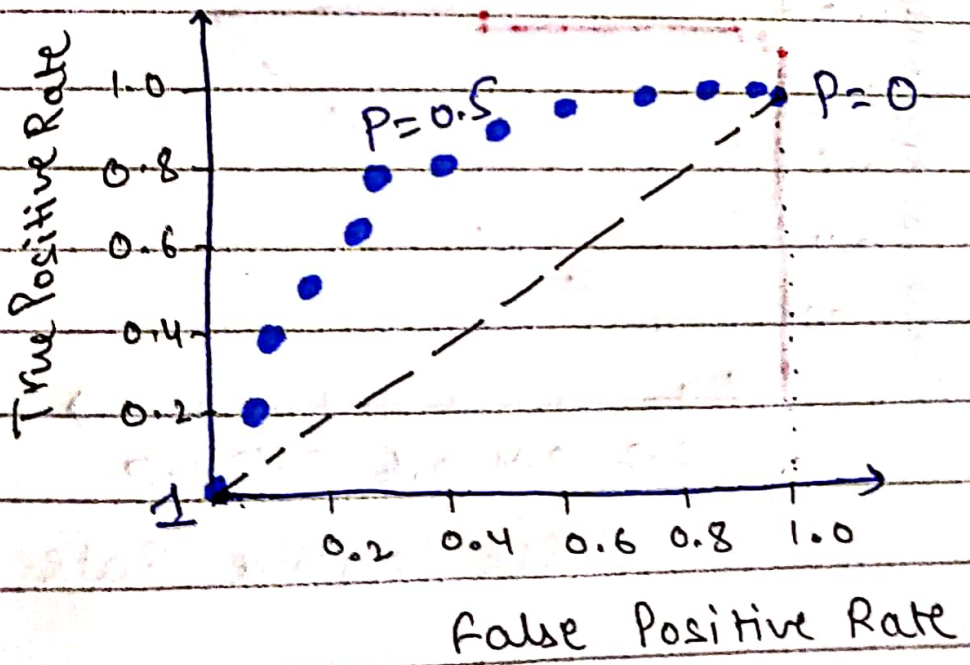
$$= \frac{TP}{TP + FN}$$

↳ (FPR : False Positive Rate) = 1 - Specifity

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

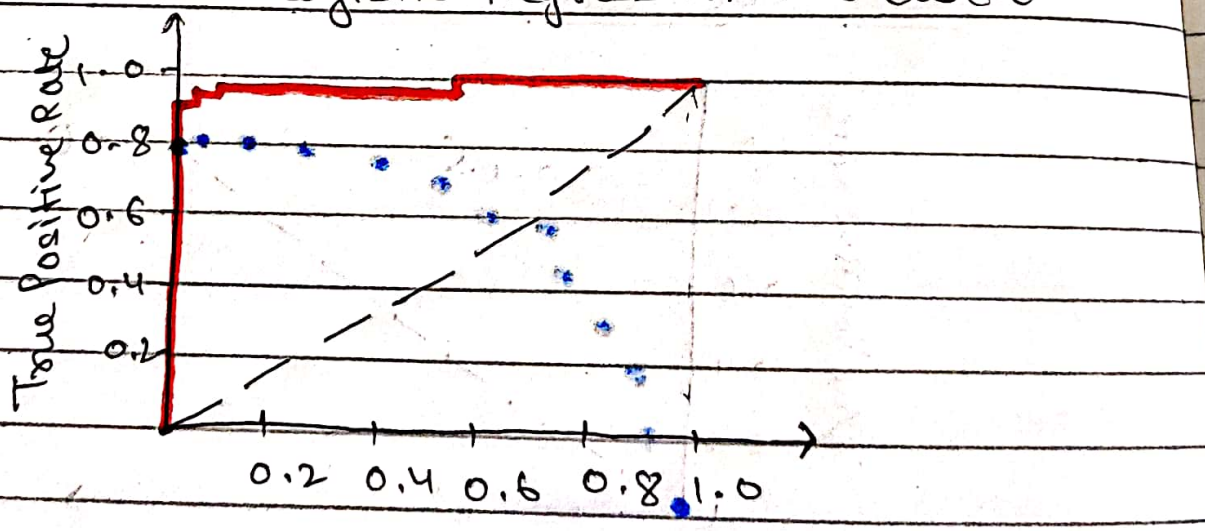
↳ Specifity = $\frac{FP}{TN + FP}$

The ROC curve [DATACAMP]



Plotting the ROC curve

- ```
  from sklearn.metrics import roc_curve
  y_pred_prob = logreg.predict_proba(X_test)[:,1]
  fpr, tpr, thresholds = roc_curve(y_test,
                                   y_pred_prob)
  plt.plot([0,1], [0,1], 'k--')
  plt.plot(fpr, tpr, label='Logistic Regression')
  plt.xlabel('False Positive Rate')
  plt.ylabel('True Positive Rate')
  plt.title('Logistic Regression ROC curve')
  plt.show()
  ```


  ```