Machine Learning

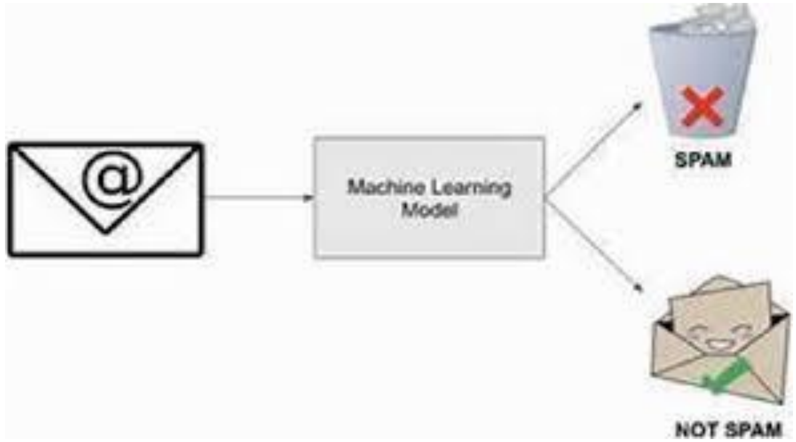
Logistic Regression

Overview

- Logistic Regression
- Type of Logistic Regression
- How Logistic Regression Works
- Decision Boundary
- Confusion Metrics
- Application
- Linear Regression vs Logistic Regression

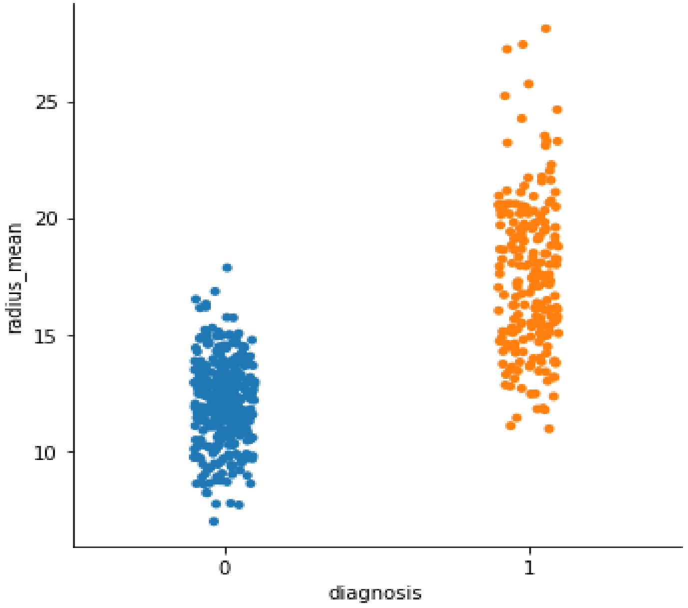
Logistic regression:

- Logistic regression is one of the most popular Supervised Machine Learning algorithm
- It is used for predicting the categorical dependent variable using a given set of independent variables.
- Logistic regression is used for solving the classification problems
- it gives the probabilistic values which lie between 0 and 1.



Example:

Classify emails as "Spam (1)" or "Not Spam (0)"



Types of Logistic Regression:

Binary Logistic Regression:

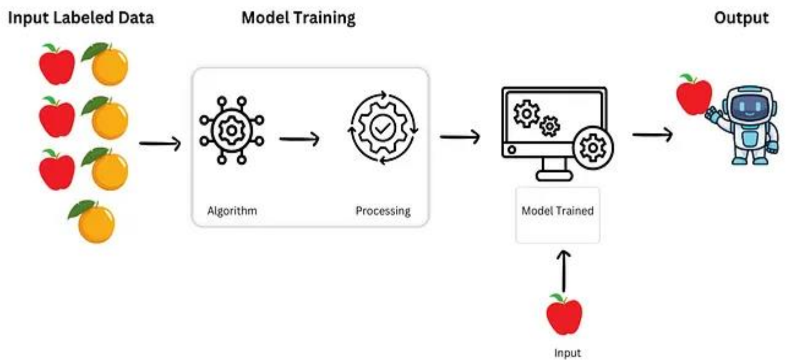
- Binary logistic regression is used to predict the outcome of a binary dependent variable based on one or more independent variables.

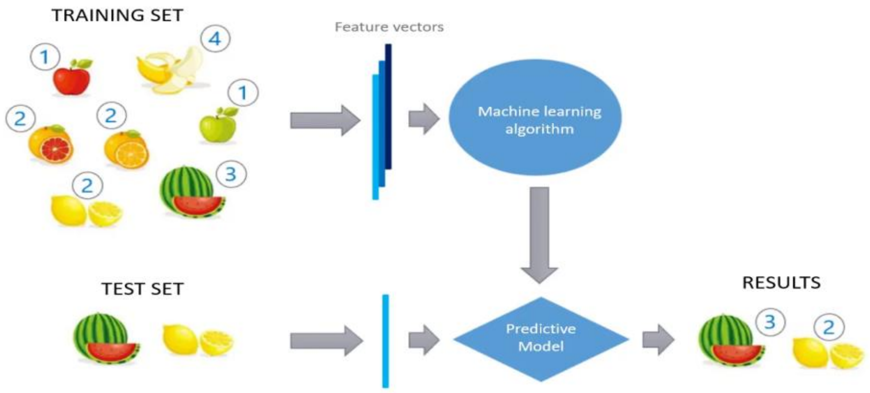
Multinomial Logistic Regression:

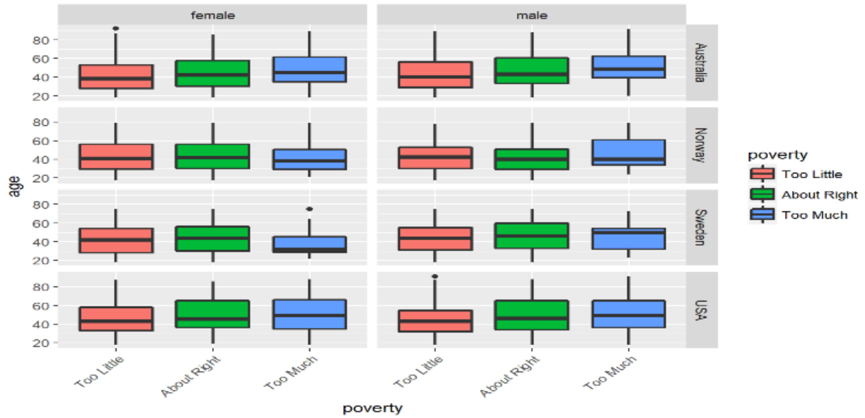
- Used when the outcome has more than two classes (multiclass classification).

Ordinal Logistic Regression:

- Used when the outcome has more than two ordered categories





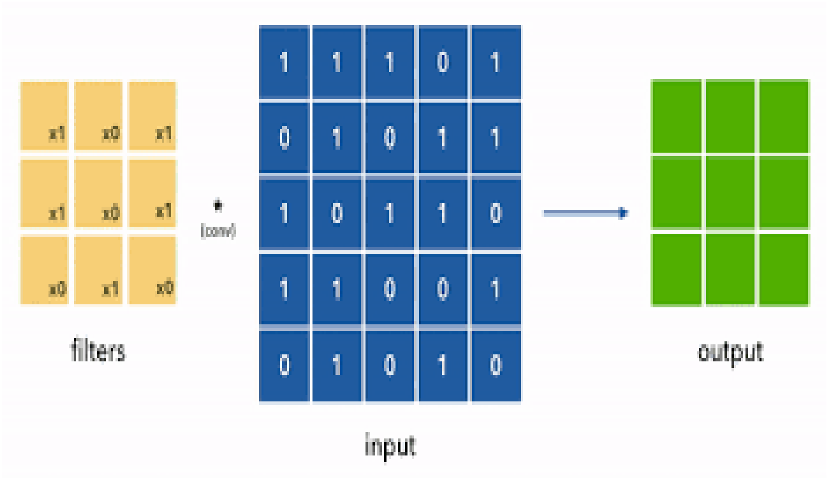


How Logistic Regression Works:

Input Features :

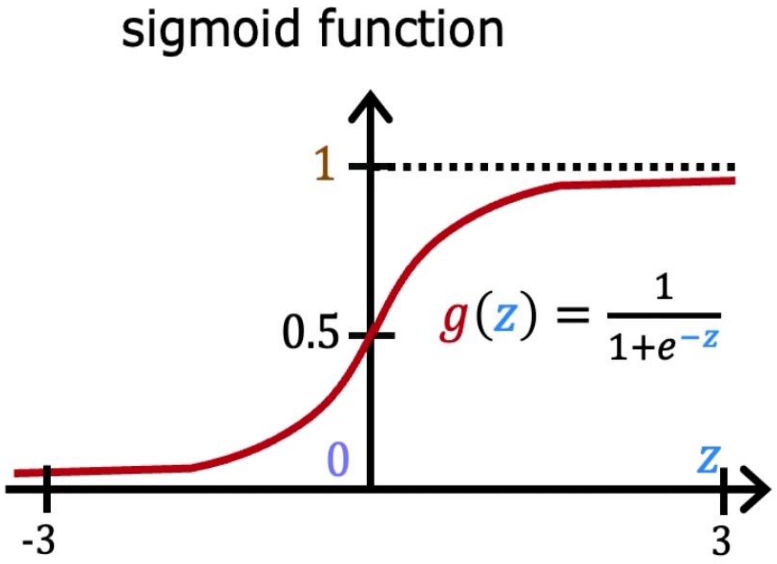
- Logistic regression uses input features (X), which can be continuous, categorical, or both, to make predictions.

- Logistic regression computes a linear combination of input features using this formula **(z=mx+b)** where z is the result, m represents the weights (slope), and b is the bias (intercept).



Sigmoid Function:

Logistic regression uses the sigmoid function to transform the linear combination zzz into a probability between 0 and 1.



Cost function: Logistic regression uses a special **cost function** called **log-loss** or **binary cross-entropy** to measure how well the model is performing.

$$LogLoss = -\frac{1}{N}\sum_{i=1}^{N}(y_i \cdot \log(p_i) + (1-y_i)\cdot \log(1-p_i))$$

Gradient Descent: Gradient descent adjusts weights to minimize the loss function, aiming for optimal values by converging to the minimum point.

cost

$$J(\vec{w},b) = -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}\log\left(f_{\vec{w},b}(\vec{x}^{(i)})\right) + (1-y^{(i)})\log\left(1-f_{\vec{w},b}(\vec{x}^{(i)})\right)\right]$$

repeat {

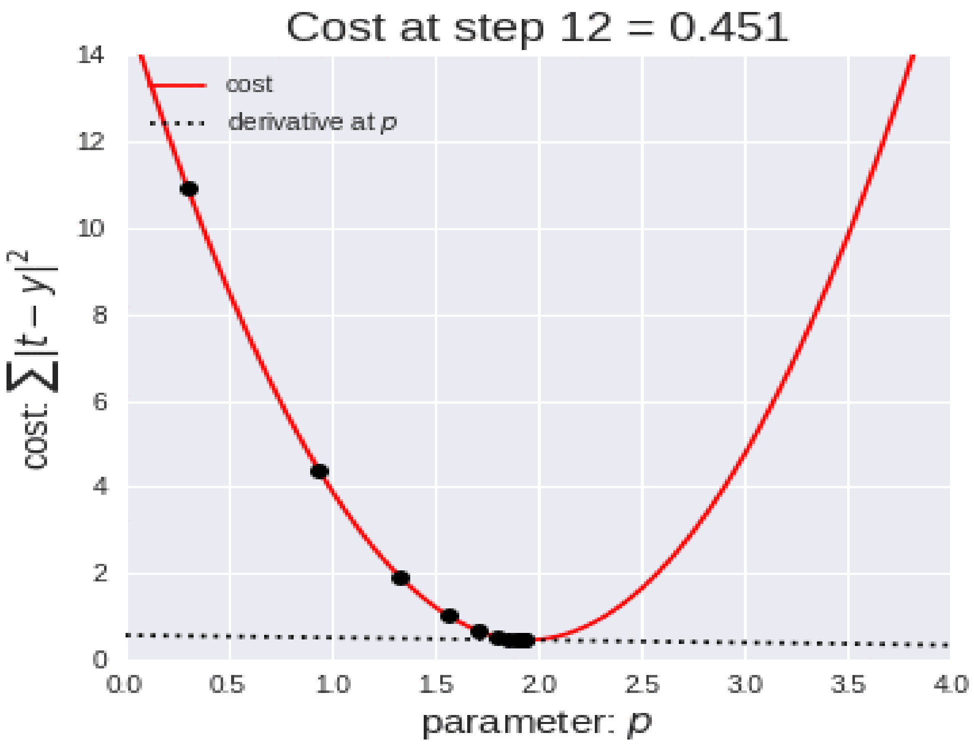
$j=1\ldots n$

$$w_j = w_j - \alpha\frac{\partial}{\partial w_j}J(\vec{w},b) \qquad \frac{\partial}{\partial w_j}J(\vec{w},b) = \frac{1}{m}\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)x_j^{(i)}$$

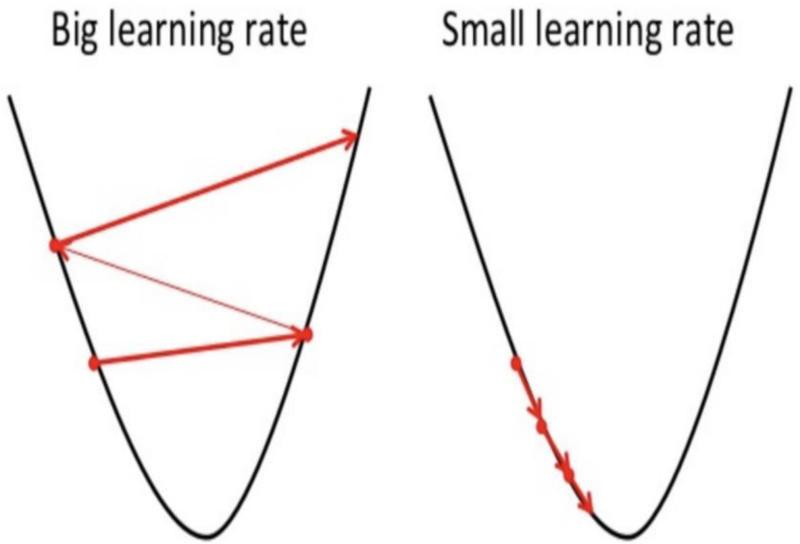
$$b = b - \alpha\frac{\partial}{\partial b}J(\vec{w},b) \qquad \frac{\partial}{\partial b}J(\vec{w},b) = \frac{1}{m}\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)$$

} simultaneous updates

The graph shows gradient descent minimizing a cost function by updating the parameter ppp at each step.
The cost decreases over 12 steps, approaching the minimum value of the curve.



- **Big learning rate**: Moves quickly but can overshoot the minimum or even diverge.
- **Small learning rate**: Moves slowly and steadily but takes longer to reach the minimum.



Decision Boundary:

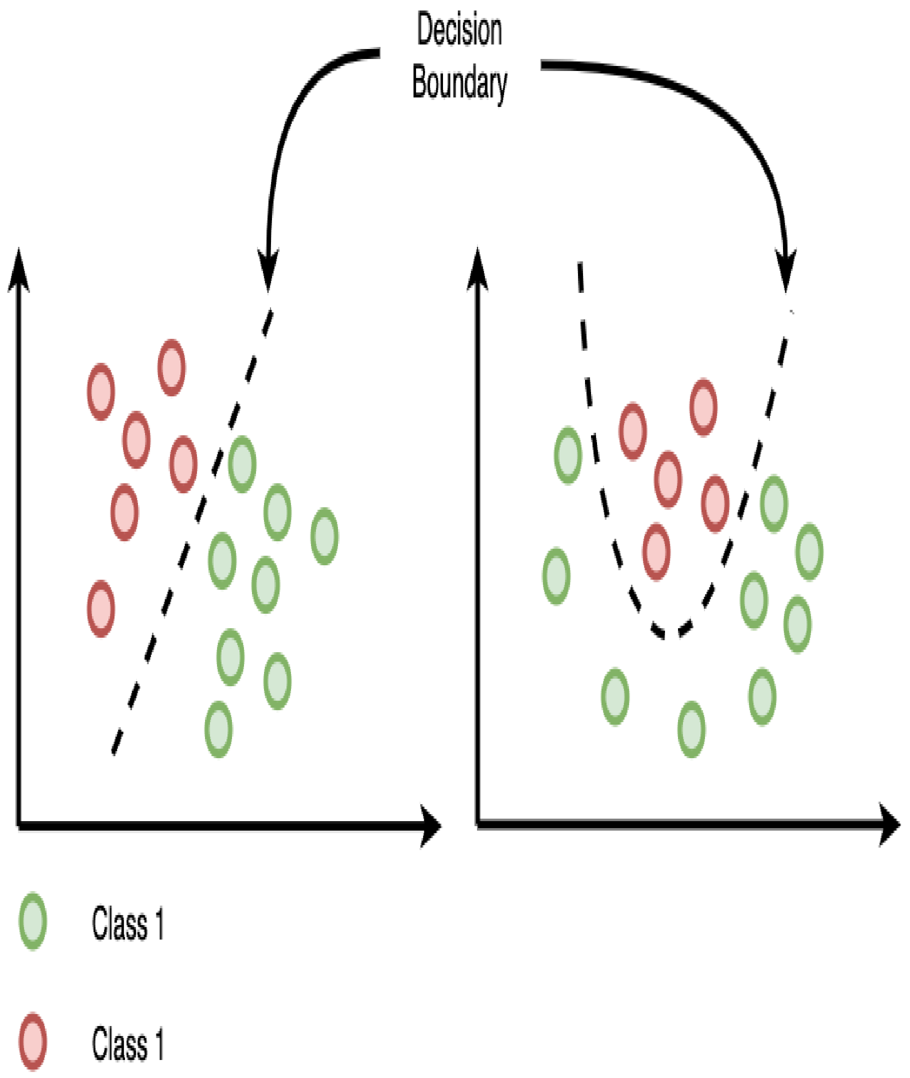
The decision boundary in logistic regression is a boundary that separates the instances of one class from the instances of the other class

It is a line (in two dimensions) or a hyperplane (in higher dimensions)

In logistic regression, the decision boundary is determined by the weights assigned to the input features

The weights control the slope and position of the decision boundary

If the weights are such that the positive class instances have a higher probability



Confusion Metrix:

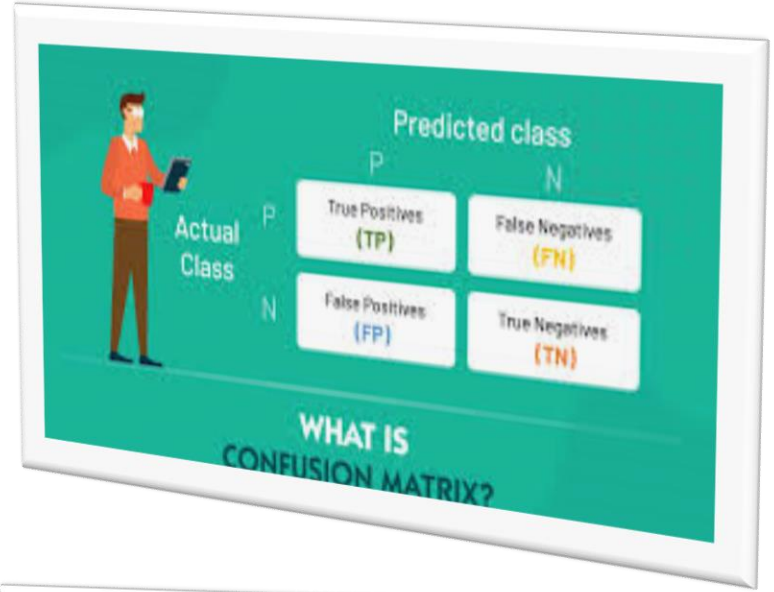
A confusion matrix summarizes a model's performance by displaying the counts of correct and incorrect predictions on test data.

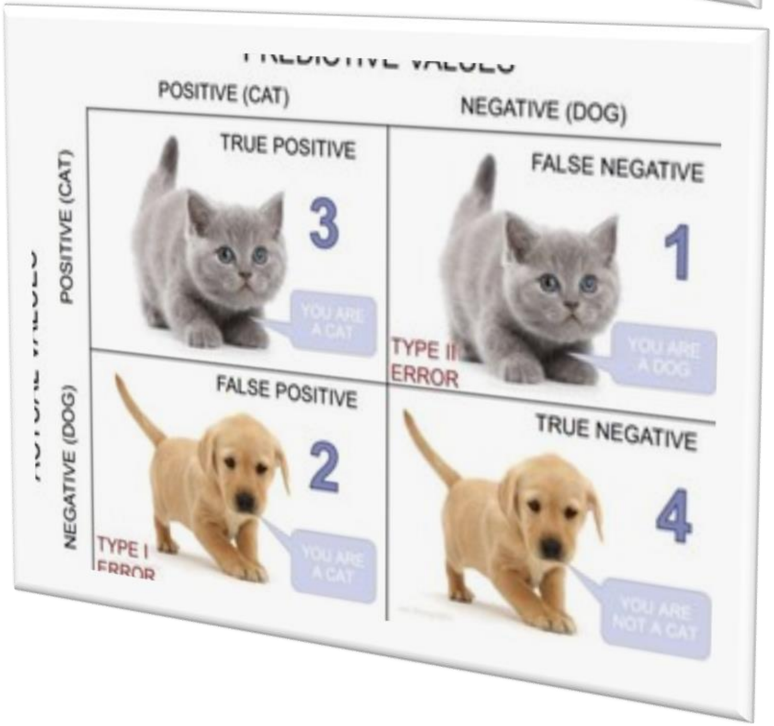
True Positive (TP): The model correctly predicted a positive outcome when the actual outcome is positive).

True Negative (TN): The model correctly predicted a negative outcome when (the actual outcome is negative).

False Positive (FP): The model incorrectly predicted a positive outcome when (the actual outcome is negative).

False Negative (FN): The model incorrectly predicted a negative outcome when (the actual outcome is positive).





Con…

Accuracy: it measures how often a machine learning model correctly predicts the outcome. You can calculate accuracy by dividing the number of correct predictions by the total number of predictions.

Precision: Precision tells us how many of the correctly predicted cases actually turned out to be positive.

Recall: Recall tells us how many of the actual positive cases we were able to predict correctly with our model.

F1 Score: It is the harmonic mean of precision and recall values. It is maximum when precision is equal to recall

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

$$F_1 = 2 * \frac{Precision * Recall}{Precision + Recall}$$

Application:

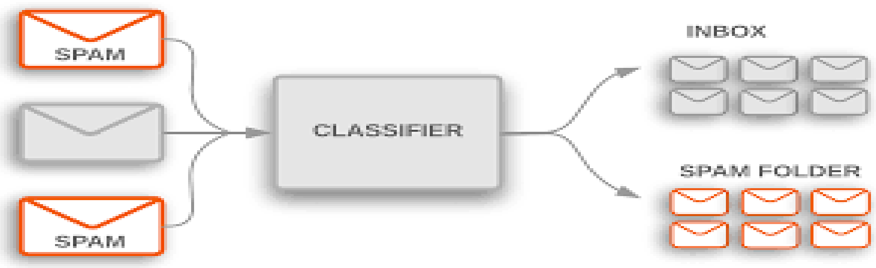
Medical Diagnosis: Logistic regression is widely used in medicine to predict disease presence or absence.

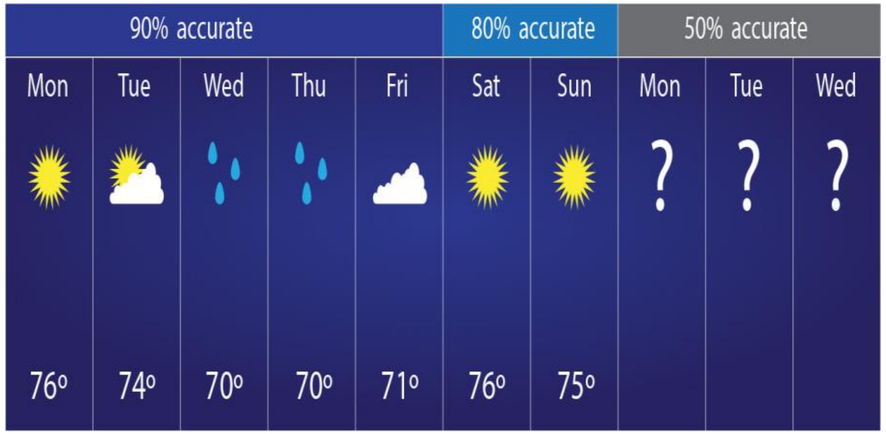
Spam Detection: Logistic regression classifies emails as spam or not spam.

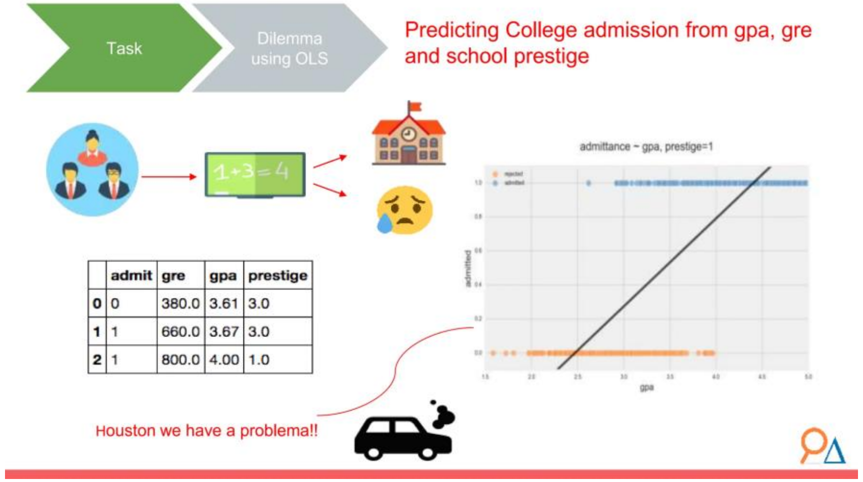
Weather Prediction: Predicting binary weather outcomes like rain or no rain.

Student Admission: Logistic regression predicts a student's likelihood of college admission.

OCR: Logistic regression classifies whether a set of features represents a specific character in OCR.







LINEAR REGRESSION	LOGISTIC REGRESSION
Supervised regression model	Supervised classification model
Predicted output is continuous	Predicted output is discrete
No activation function used	Activation function is used (Sigmoid)
Model measured by RMSE	Model measured by Confusion matrix
Predicts values like price, age, etc.	Predicts values like 0 or 1, Yes or No, etc

Thank You

Farooq shehzad