LOGISTIC REGRESSION

simplilearn

© Simplilearn. All rights reserved.

Surviving the Titanic

Suppose, you have to build a model to predict how many people survived the Titanic shipwreck

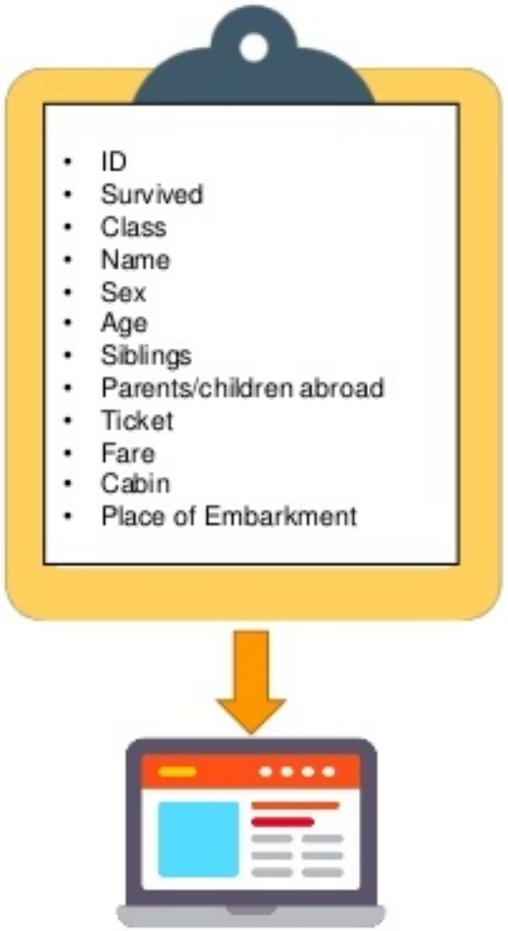
©Simplilearn. All rights reserved.

Surviving the Titanic

Suppose, you have to build a model to predict how many people survived the Titanic shipwreck

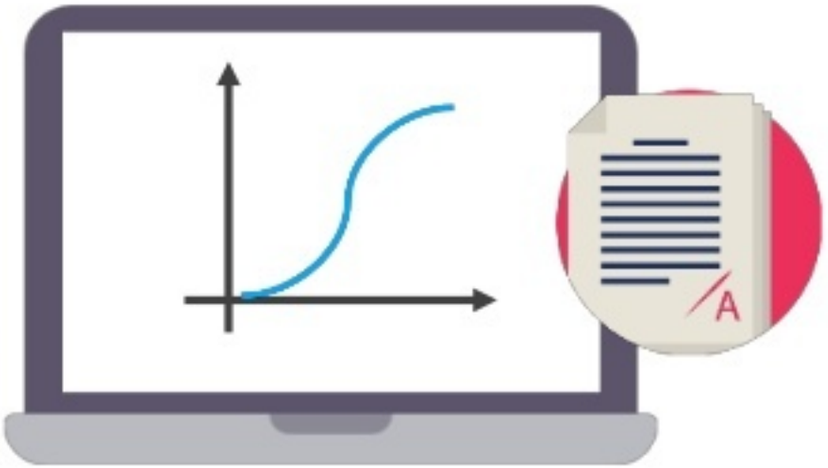
©Simplilearn. All rights reserved.

Surviving the Titanic



Teaching the model with the passenger dataset

Dropping the non-essential components of the dataset



Determining the survival of passengers and evaluating the model

©Simplilearn. All rights reserved.

Agenda

- What is Supervised Learning?
- What is Classification? What are some of its solutions?
- What is Logistic Regression?
- Comparing Linear and Logistic Regression
- Logistic Regression applications
- Use Case – Predicting the number in an image

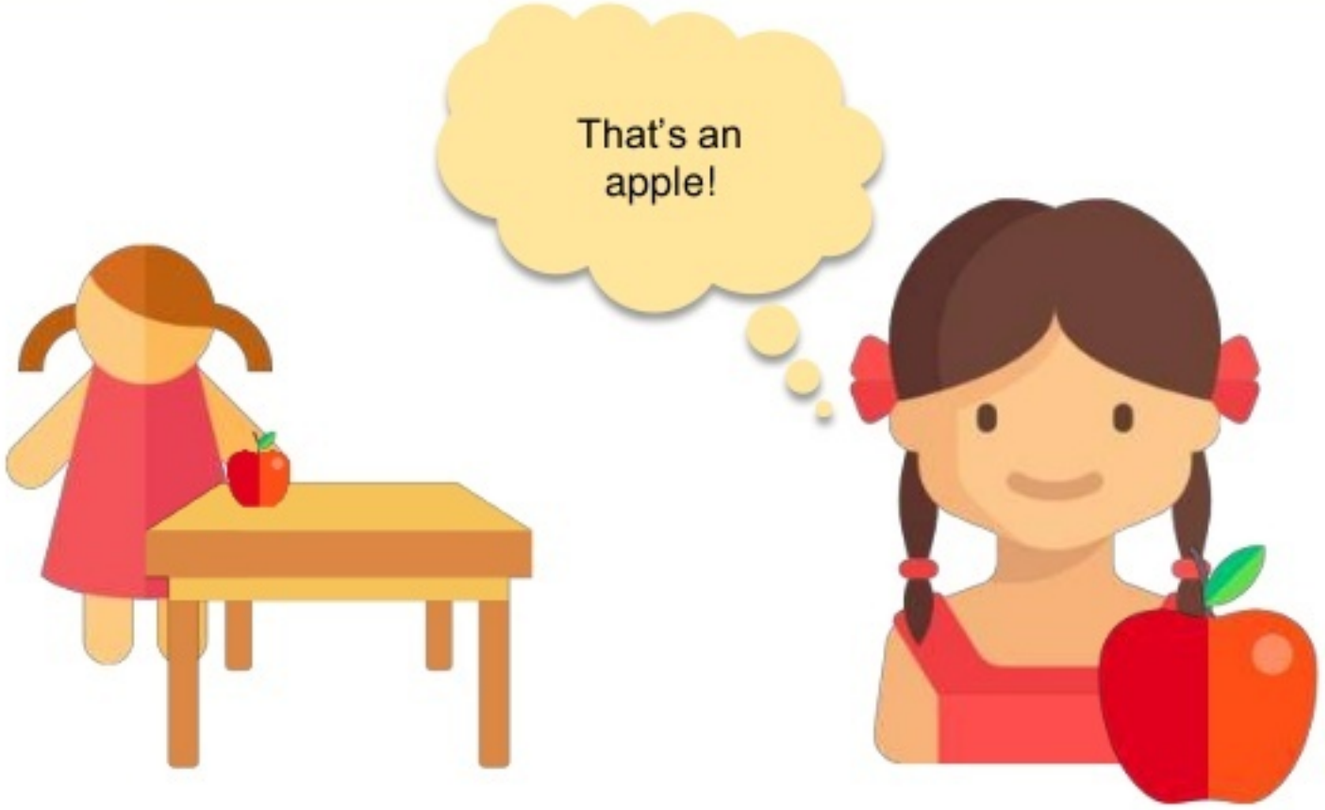
©Simplilearn. All rights reserved.

What is Supervised Learning?

©Simplilearn. All rights reserved.

What is Supervised Learning?

Teacher teaches child

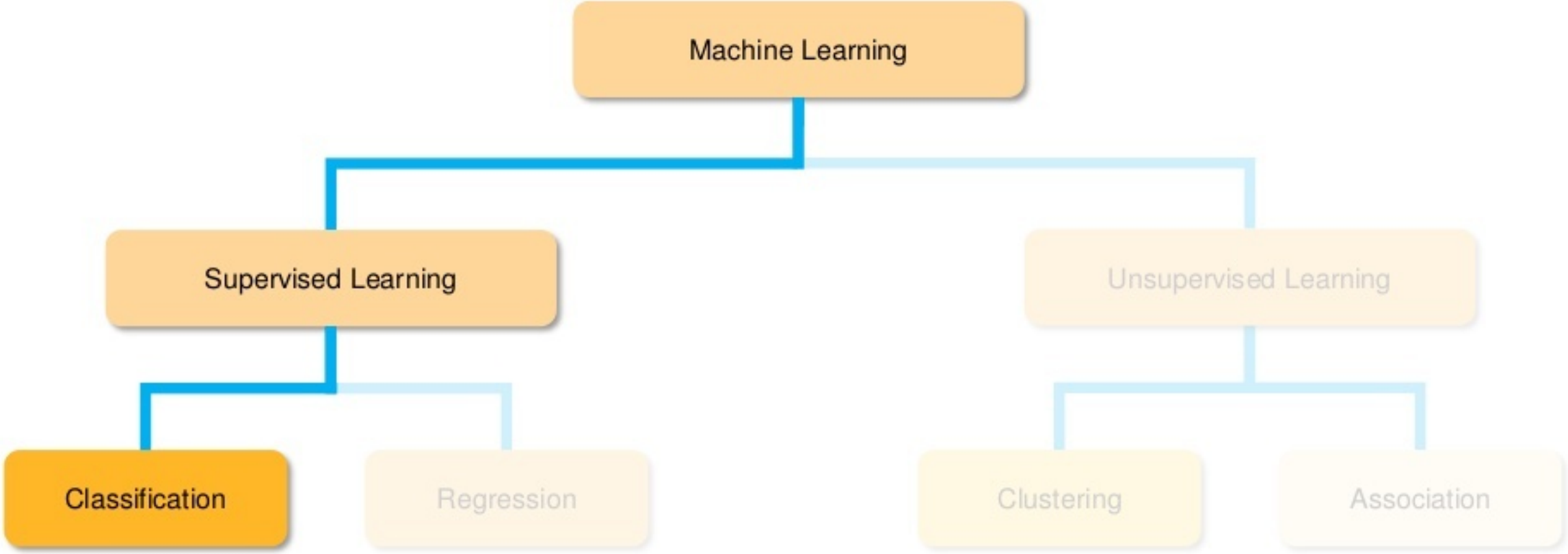


Child recognizes an apple when she sees it again

A model is able to make predictions based on past data

©Simplilearn. All rights reserved.

Where does Logistic Regression fit it?



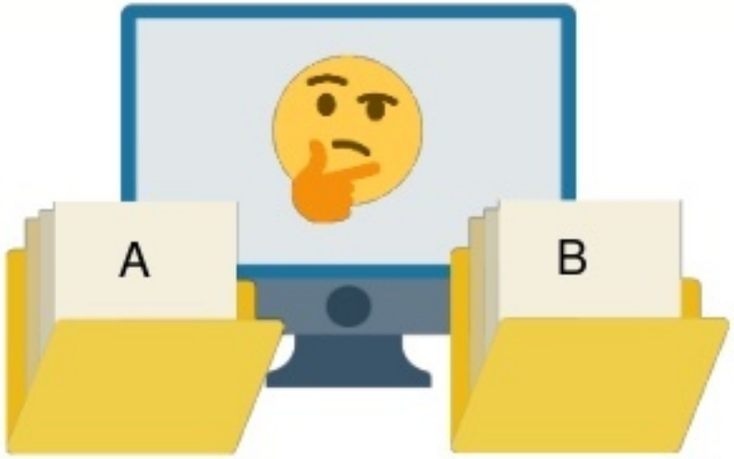
The systems predicts future outcomes based on training from past input

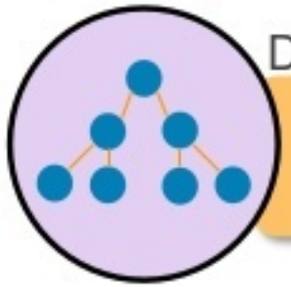
©Simplilearn. All rights reserved.

Solutions to Classification

©Simplilearn. All rights reserved.

A few Classification Solutions



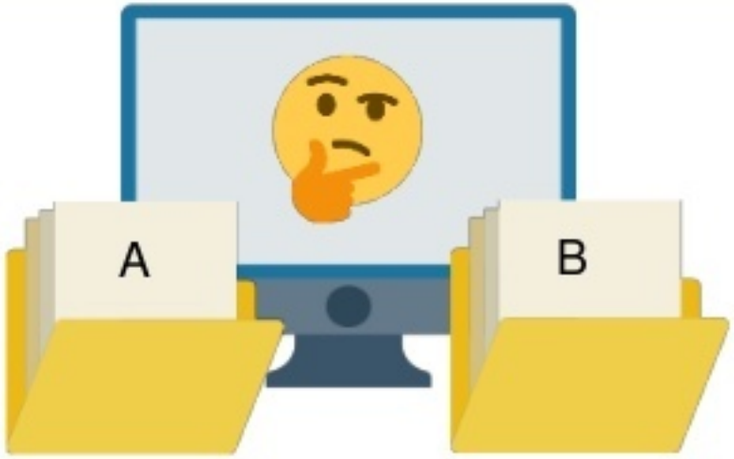


Decision Trees

We take decisions using a tree structure. Each branch node represents a choice, and leaf node represents a decision

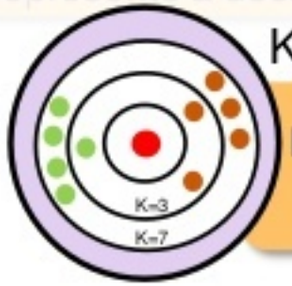
©Simplilearn. All rights reserved.

A few Classification Solutions



Decision Trees

We take decisions using a tree structure. Each branch node represents a choice, and leaf node represents a decision

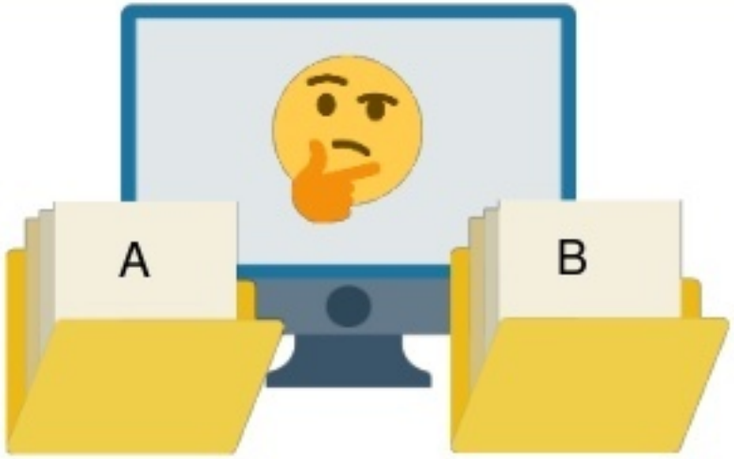


K-Nearest Neighbor

It helps determine what the given object is, based on its similarity to the objects it is compared to

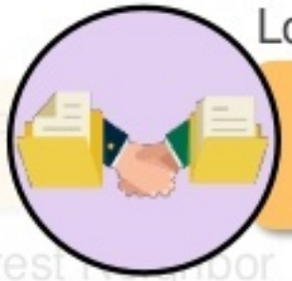
©Simplilearn. All rights reserved.

A few Classification Solutions



Decision Trees

We determine the probability of an event occurring with the help of a tree structure



Logistic Regression

A dataset with one or more independent variables is used to determine binary output of the dependent variable

K-Nearest Neighbor

It helps determine what the given object is, based on its similarity to the objects it is compared to

©Simplilearn. All rights reserved.

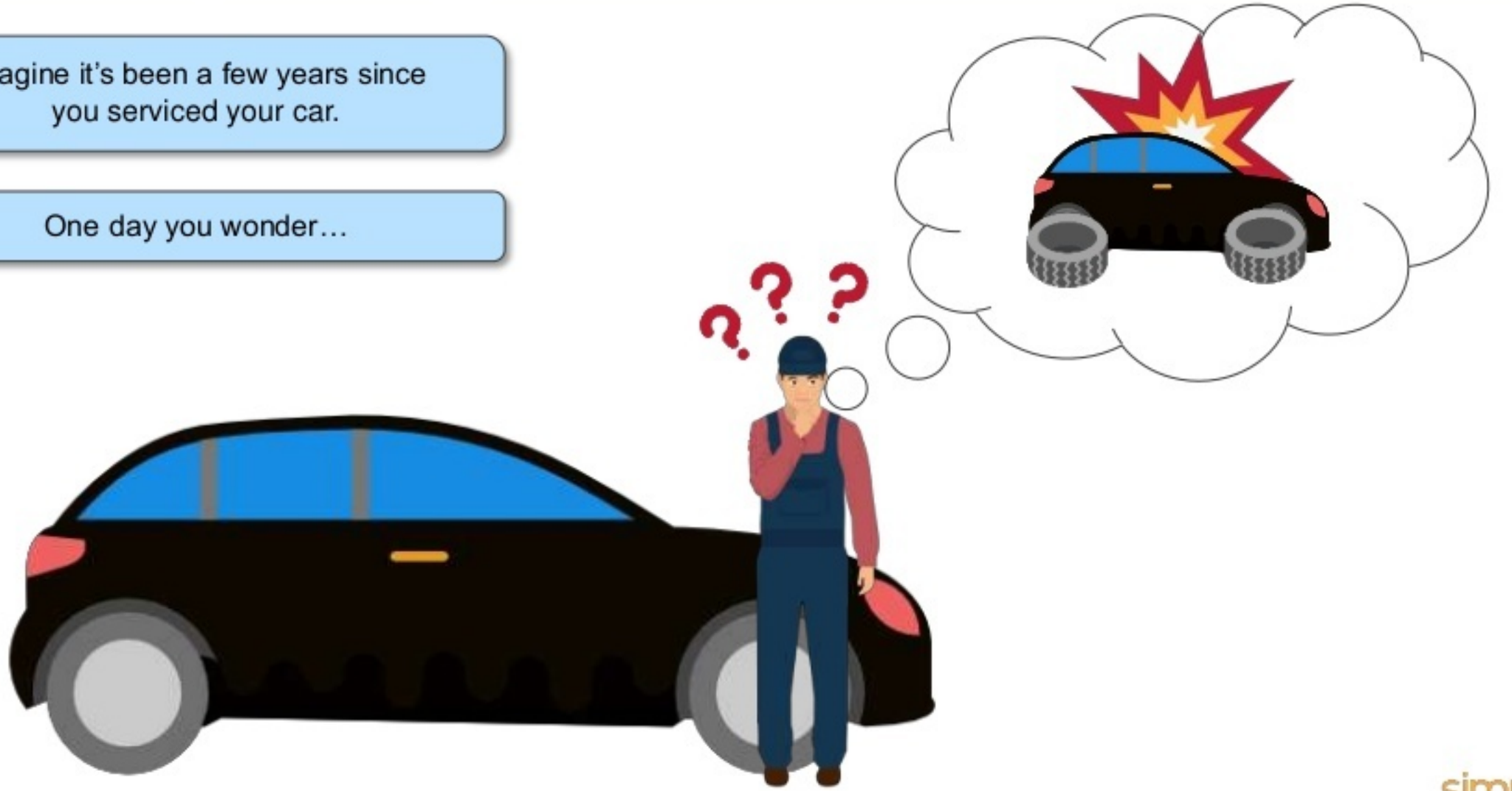
What is Logistic Regression?

©Simplilearn. All rights reserved.

What is Logistic Regression?

Imagine it's been a few years since you serviced your car.

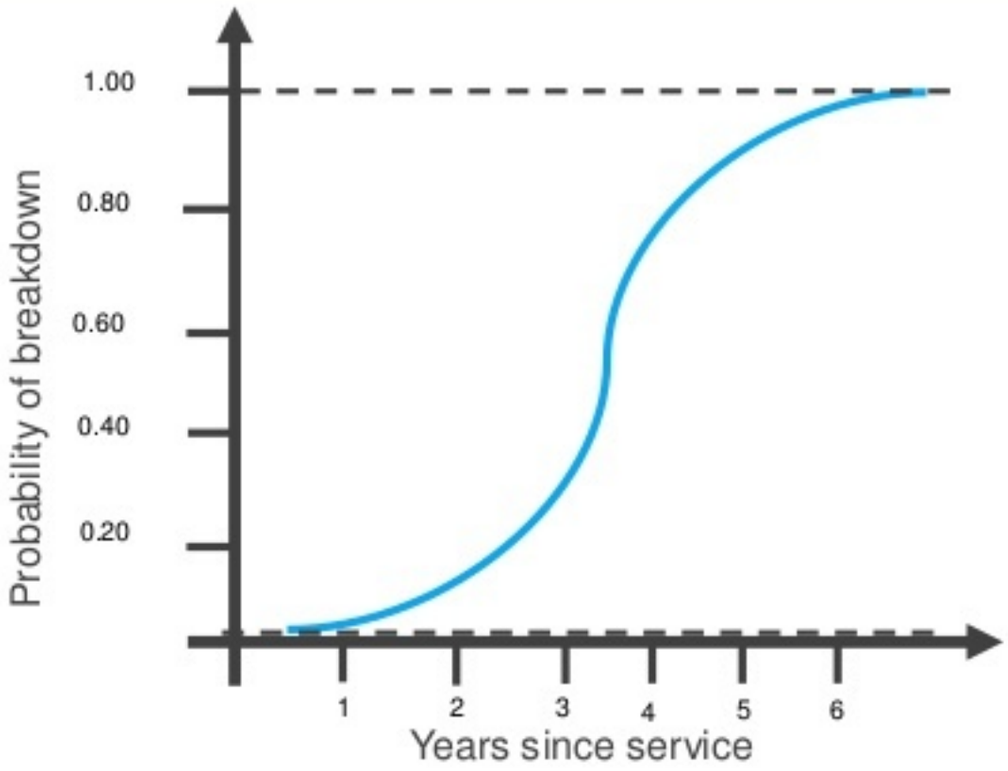
One day you wonder…



©Simplilearn. All rights reserved.

What is Logistic Regression?

You provide years since last service

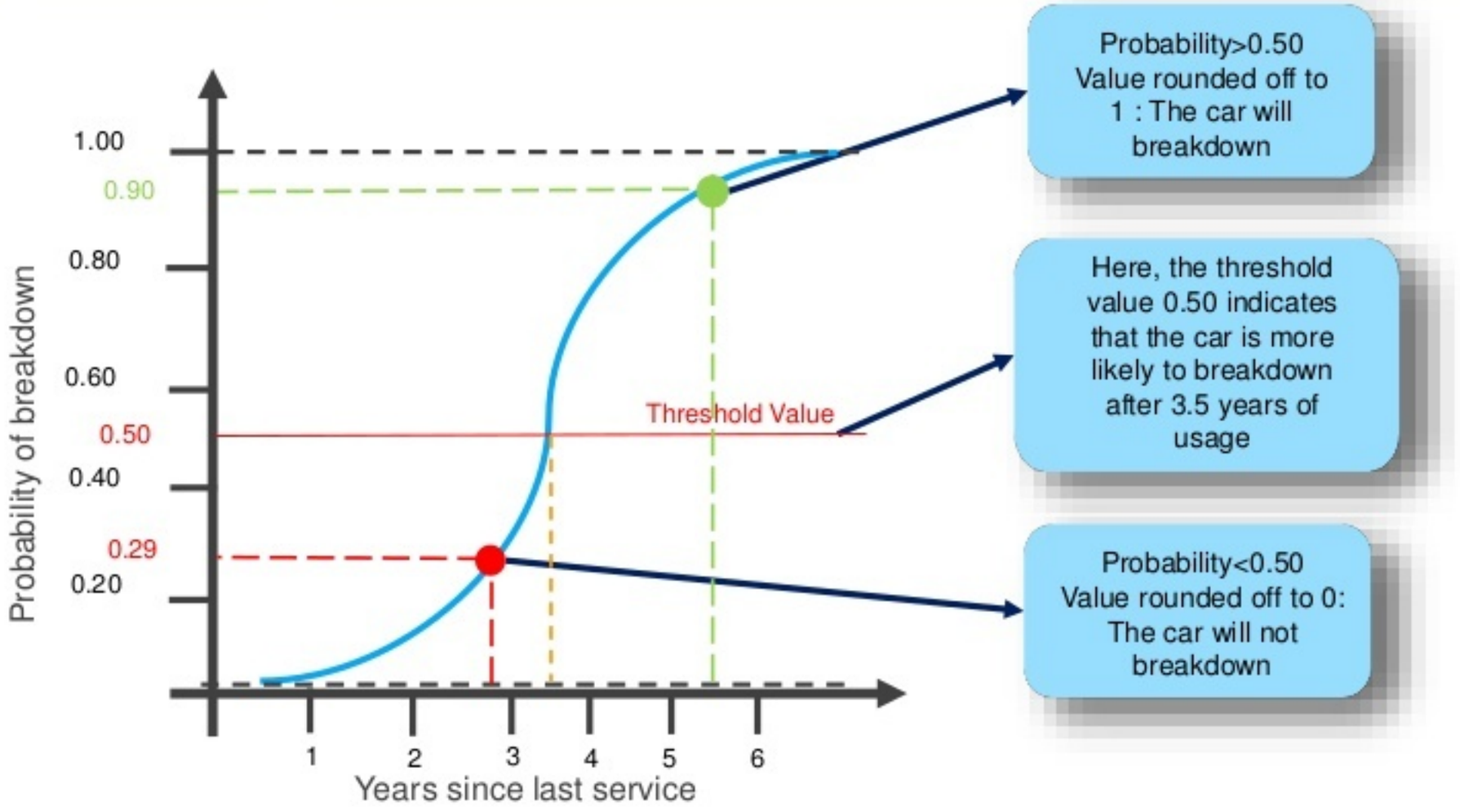


Regression model created based on other users' experience

It is a classification algorithm, used to predict binary outcomes for a given set of independent variables. The dependent variable's outcome is discrete.

©Simplilearn. All rights reserved.

What is Logistic Regression?



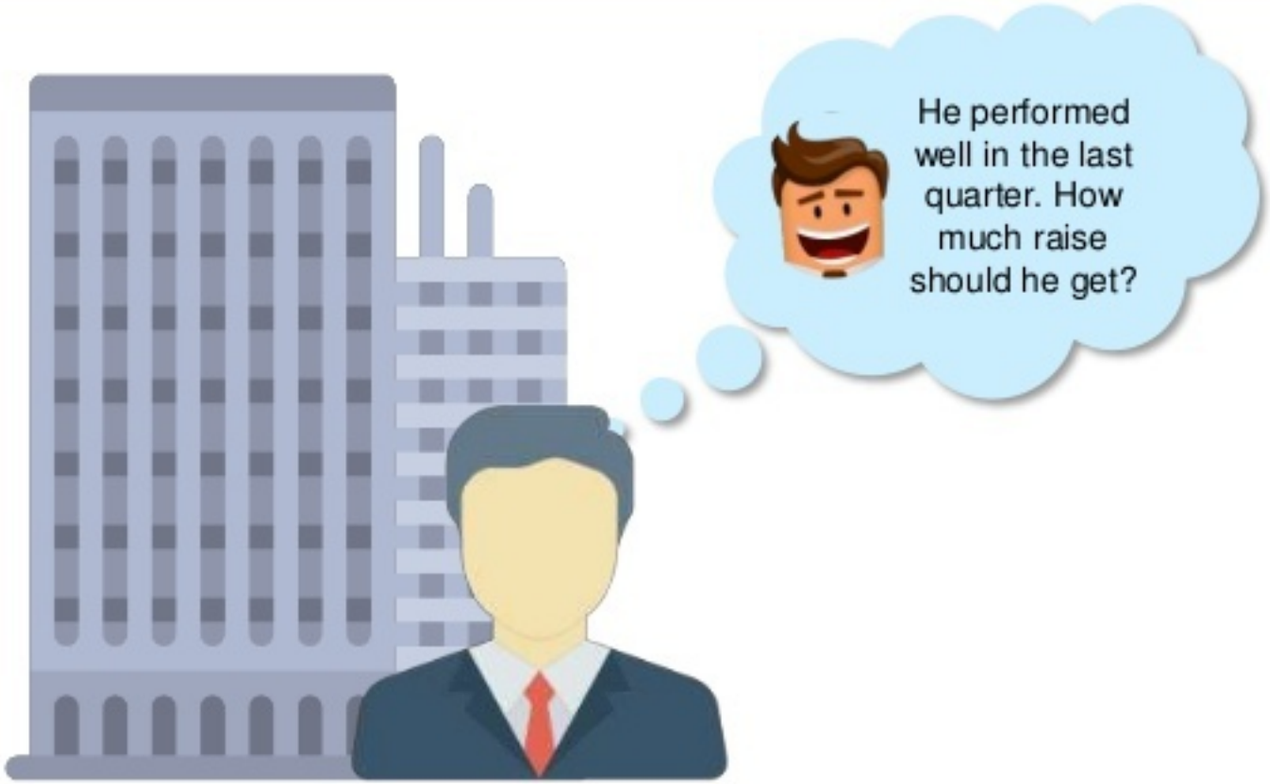
Model makes predictions

©Simplilearn. All rights reserved.

What is Linear Regression?

©Simplilearn. All rights reserved.

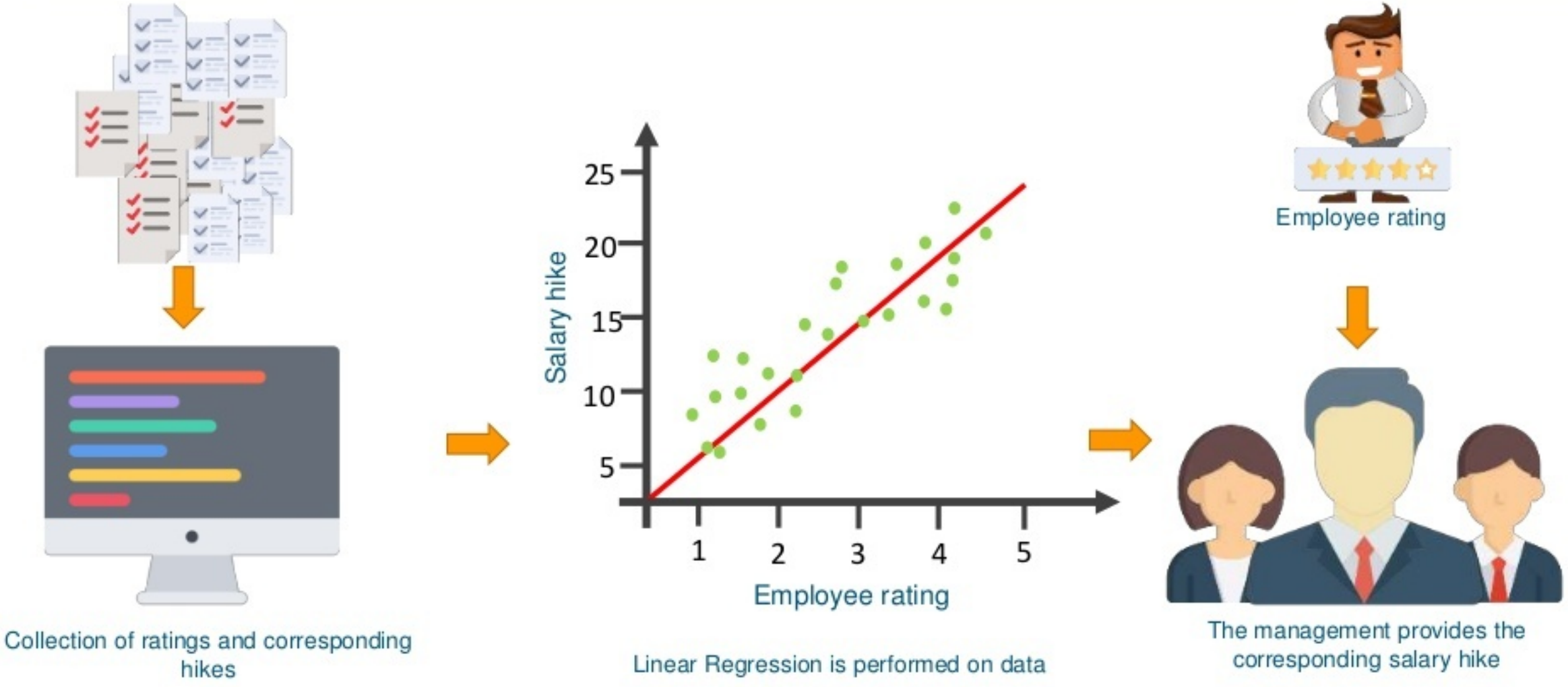
Linear Regression



It is a statistical method that helps find the relationship between an independent and dependent variable, both of which are continuous

©Simplilearn. All rights reserved.

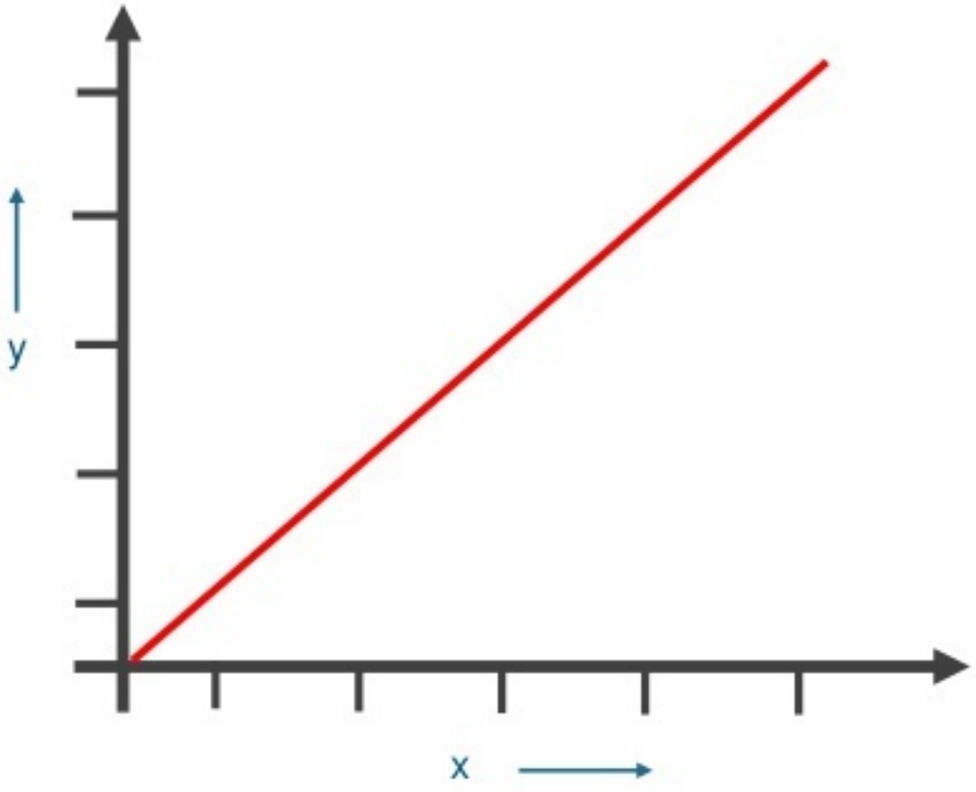
Linear Regression



©Simplilearn. All rights reserved.

Linear and Logistic Regression

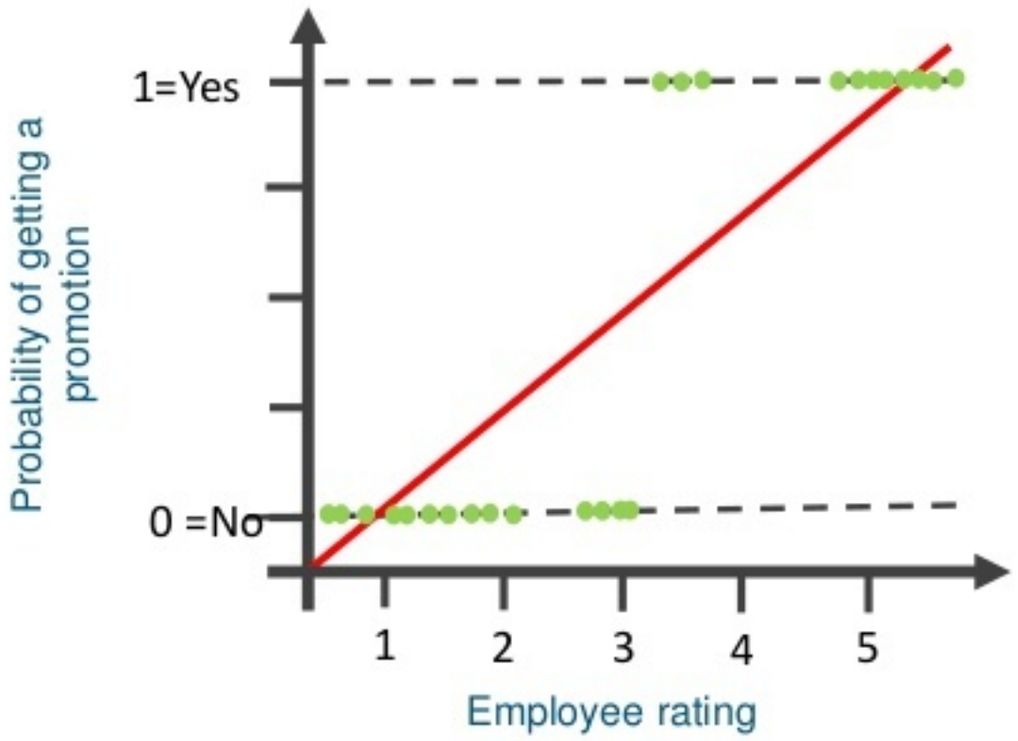
Here's the graph of how linear regression would be, for a given scenario



©Simplilearn. All rights reserved.

Linear and Logistic Regression

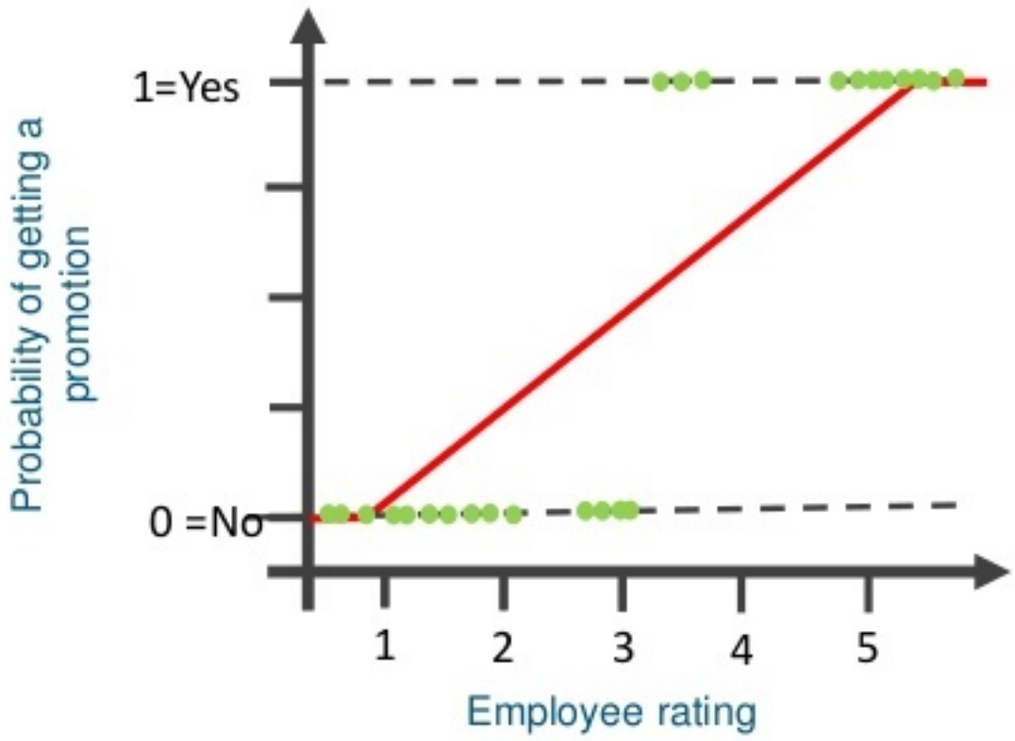
What if you wanted to know whether the employee would get a promotion or not based on their rating



©Simplilearn. All rights reserved.

Linear and Logistic Regression

This graph would not be able to make such a prediction. So we clip the line at 0 and 1.



©Simplilearn. All rights reserved.

Linear and Logistic Regression

So, how did this…

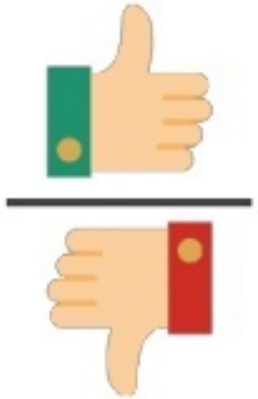
…become this?

©Simplilearn. All rights reserved.

The Math behind Logistic Regression

To understand Logistic Regression, let's talk about the odds of success

Odds (θ) =



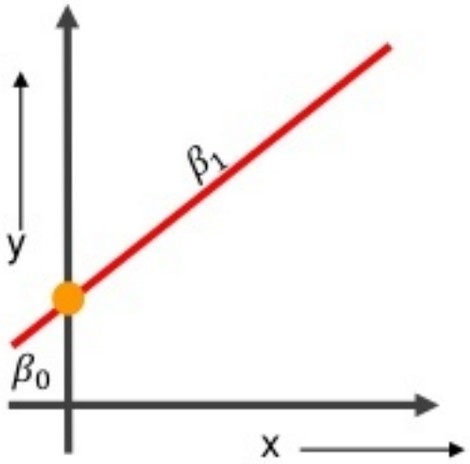
$$\frac{\text{Probability of an event happening}}{\text{Probability of an event not happening}} \quad or \quad \theta = \frac{p}{1-p}$$

The values of odds range from 0 to ∞
The values of probability change from 0 to 1

©Simplilearn. All rights reserved.

The Math behind Logistic Regression

Take the equation of the straight line



Here, β_0 is the y-intercept
β_1 is the slope of the line
x is the value of the x co-ordinate
y is the value of the prediction

The equation would be: $y = \beta_0 + \beta_1 x$

©Simplilearn. All rights reserved.

The Math behind Logistic Regression

Now, we predict the odds of success

$$\log\left(\frac{p(x)}{1-P(x)}\right) = \beta_0 + \beta_1 x$$

Exponentiating both sides:

$$e^{ln}\left(\frac{p(x)}{1-p(x)}\right) = e^{\beta_0+\beta_1 x}$$

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

Let Y = $e^{\beta_0+\beta_1 x}$

Then $\frac{p(x)}{1-p(x)} = Y$

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

The equation of a sigmoid function:

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

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

simplilearn

©Simplilearn. All rights reserved.

The Math behind Logistic Regression

A sigmoid curve is obtained!

©Simplilearn. All rights reserved.

Comparing Linear and Logistic Regression

©Simplilearn. All rights reserved.

How is Linear and Logistic Regression different?

Linear Regression

- Used to solve Regression Problems

Logistic Regression

©Simplilearn. All rights reserved.

How is Linear and Logistic Regression different?

Linear Regression

- Used to solve Regression Problems

Logistic Regression

- Used to solve Classification Problems

©Simplilearn. All rights reserved.

How is Linear and Logistic Regression different?

Linear Regression

- Used to solve Regression Problems
- The response variables are continuous in nature

Logistic Regression

- Used to solve Classification Problems

©Simplilearn. All rights reserved.

How is Linear and Logistic Regression different?

Linear Regression

- Used to solve Regression Problems
- The response variables are continuous in nature

Logistic Regression

- Used to solve Classification Problems
- The response variable is categorical in nature

©Simplilearn. All rights reserved.

How is Linear and Logistic Regression different?

Linear Regression

- Used to solve Regression Problems
- The response variables are continuous in nature
- It helps estimate the dependent variable when there is a change in the independent variable.

Logistic Regression

- Used to solve Classification Problems
- The response variable is categorical in nature

©Simplilearn. All rights reserved.

How is Linear and Logistic Regression different?

Linear Regression

- Used to solve Regression Problems
- The response variables are continuous in nature
- It helps estimate the dependent variable when there is a change in the independent variable.

Logistic Regression

- Used to solve Classification Problems
- The response variable is categorical in nature
- It helps calculate the possibility of a particular event taking place.

©Simplilearn. All rights reserved.

How is Linear and Logistic Regression different?

Linear Regression

- Used to solve Regression Problems
- The response variables are continuous in nature
- It helps estimate the dependent variable when there is a change in the independent variable.
- Is a straight line.

Logistic Regression

- Used to solve Classification Problems
- The response variable is categorical in nature
- It helps calculate the possibility of a particular event taking place.

©Simplilearn. All rights reserved.

How is Linear and Logistic Regression different?

Linear Regression	Logistic Regression
• Used to solve Regression Problems • The response variables are continuous in nature • It helps estimate the dependent variable when there is a change in the independent variable.	• Used to solve Classification Problems • The response variable is categorical in nature • It helps calculate the possibility of a particular event taking place.
• Is a straight line.	• An S-curve. (S = Sigmoid)

©Simplilearn. All rights reserved.

Logistic Regression Applications

Weather Prediction

Helps determine the kind of weather that can be expected

©Simplilearn. All rights reserved.

Logistic Regression Applications

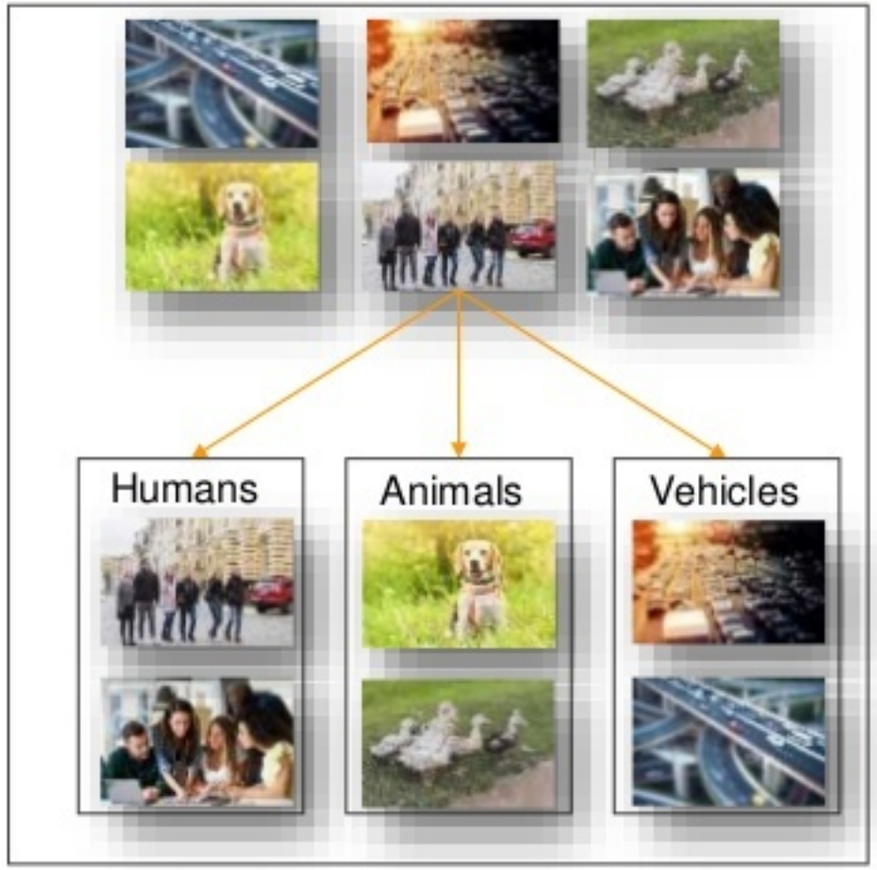
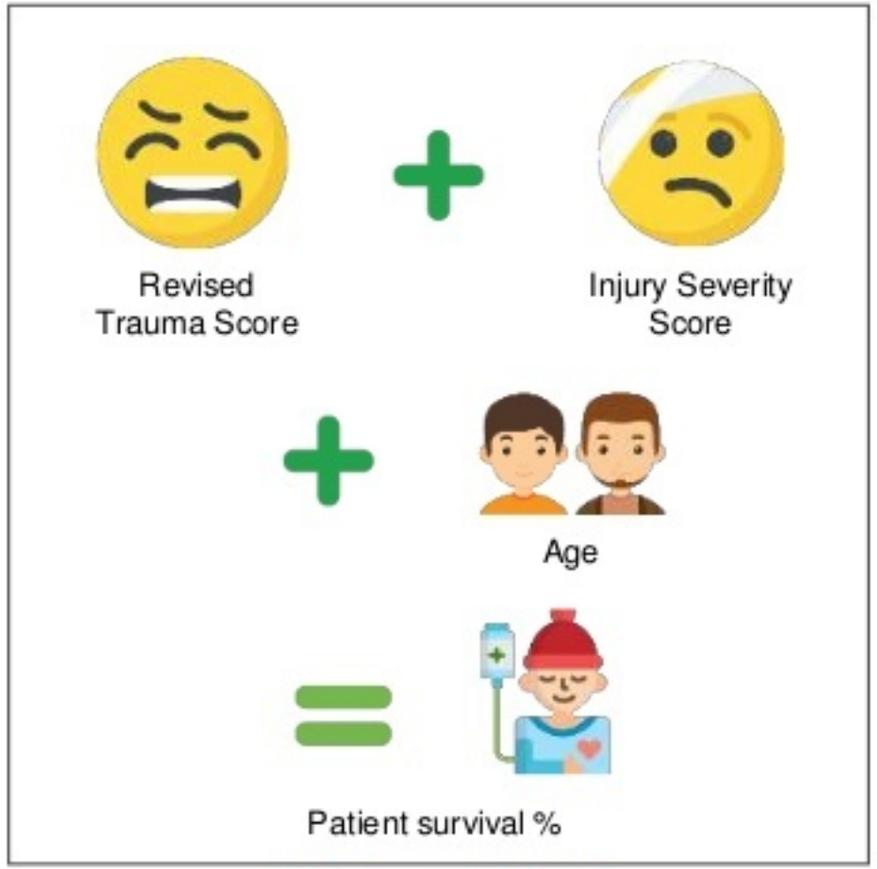


Image Categorization

Identifies the different components that are present in the image, and helps categorize them

©Simplilearn. All rights reserved.

Logistic Regression Applications



Healthcare (TRISS)

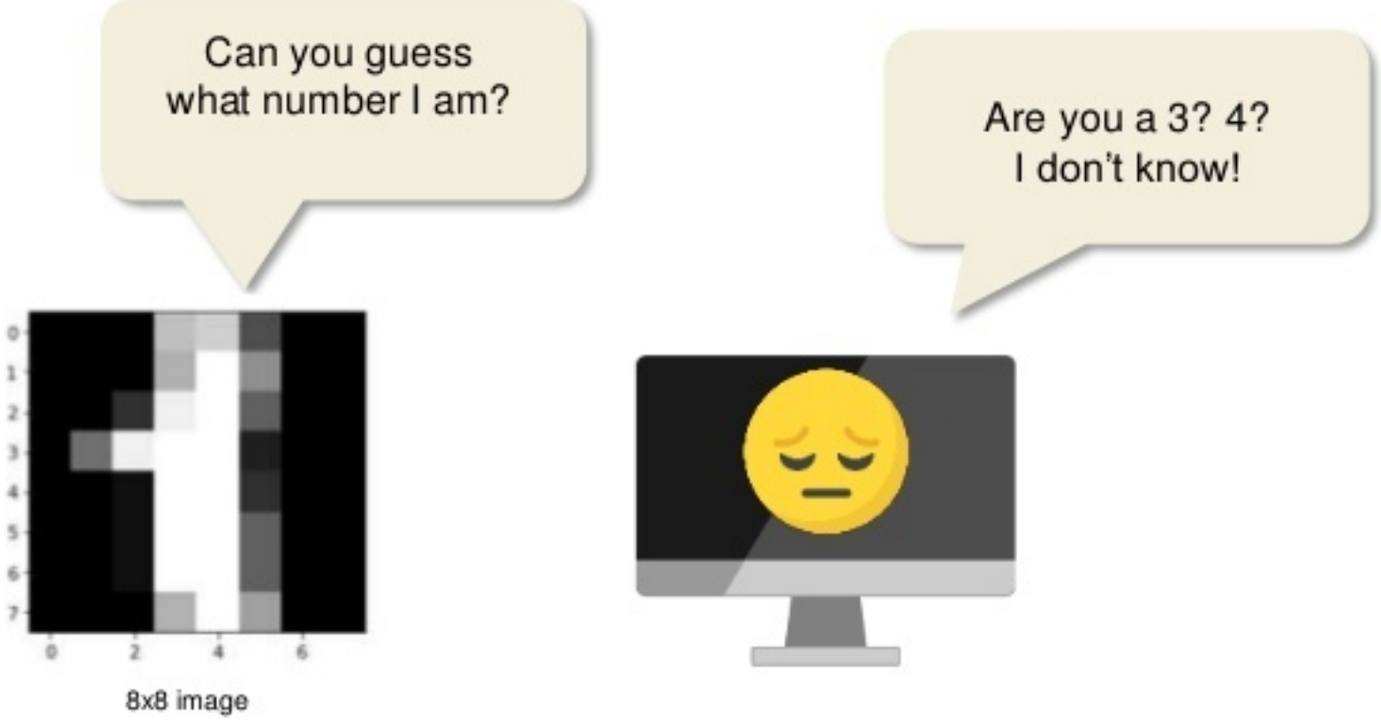
Determines the possibility of patient survival, taking age, ISS and RTS into consideration

©Simplilearn. All rights reserved.

Use Case – Predicting numbers in images

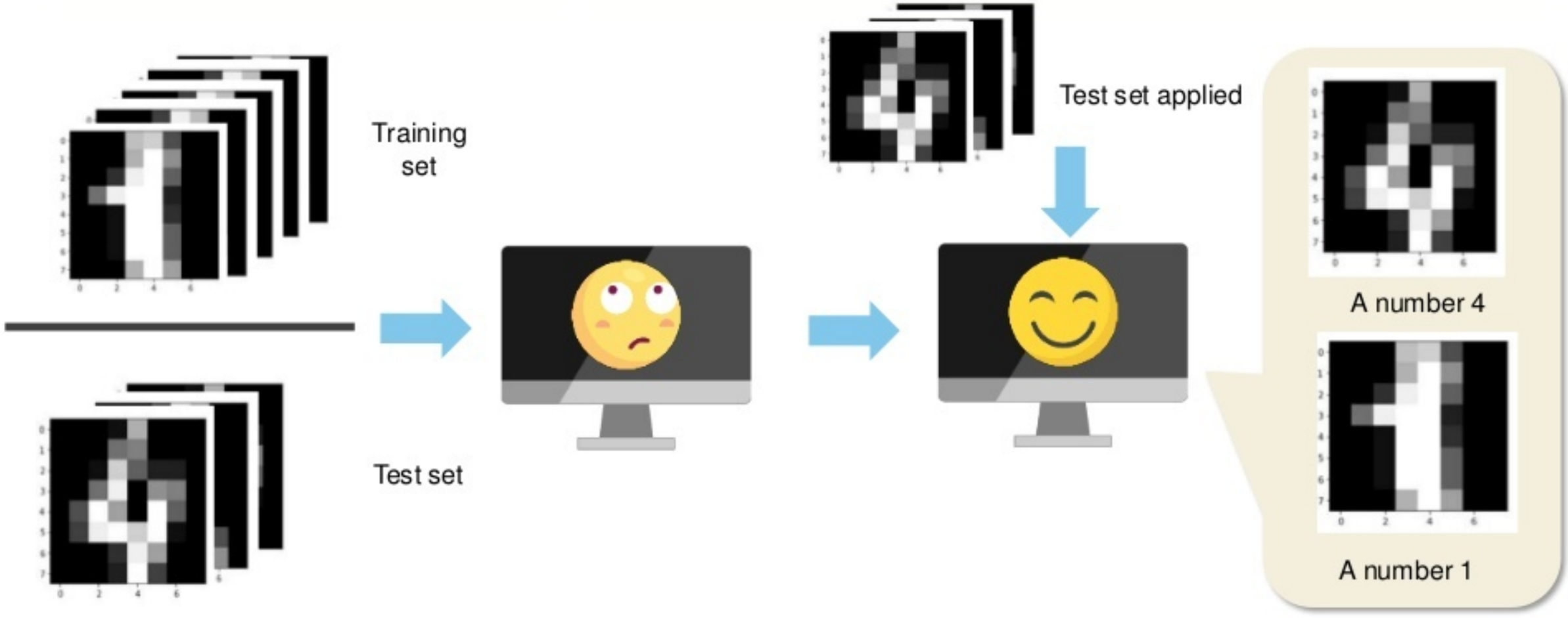
©Simplilearn. All rights reserved.

Use Case – Predicting numbers



©Simplilearn. All rights reserved.

Use Case – Predicting numbers



Dividing the data set

The model being trained

Model identifies number in images

©Simplilearn. All rights reserved.

Use Case – Implementation

Importing libraries and their associated methods

```
from sklearn.datasets import load_digits
from sklearn.model_selection import train_test_split
import numpy as np

import matplotlib.pyplot as plt
import seaborn as sns
from sklearn import metrics
%matplotlib inline
digits = load_digits()
```

Determining the total number of images and labels

```
print("Image Data Shape", digits.data.shape)
print("Label Data Shape", digits.target.shape)
```

```
Image Data Shape (1797, 64)
Label Data Shape (1797,)
```

©Simplilearn. All rights reserved.

Use Case – Implementation

Displaying some of the images and labels

```
import numpy as np
import matplotlib.pyplot as plt

plt.figure(figsize=(20,4))
for index, (image, label) in enumerate(zip(digits.data[0:5], digits.target[0:5])):
 plt.subplot(1, 5, index + 1)
 plt.imshow(np.reshape(image, (8,8)), cmap=plt.cm.gray)
 plt.title('Training: %i\n' % label, fontsize = 20)
```

©Simplilearn. All rights reserved.

Use Case – Implementation

Dividing dataset into Training and Test set

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(digits.data, digits.target, test_size=0.23, random_state=2)
```

```
print(x_train.shape)
```

```
(1383, 64)
```

```
print(y_train.shape)
```

```
(1383,)
```

```
print(x_test.shape)
```

```
(414, 64)
```

```
print(y_test.shape)
```

```
(414,)
```

©Simplilearn. All rights reserved.

Use Case – Implementation

Import the Logistic Regression model

```
from sklearn.linear_model import LogisticRegression
```

Making an instance of the model and training it

```
logisticRegr = LogisticRegression()
logisticRegr.fit(x_train, y_train)
```

```
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
          intercept_scaling=1, max_iter=100, multi_class='ovr', n_jobs=1,
          penalty='l2', random_state=None, solver='liblinear', tol=0.0001,
          verbose=0, warm_start=False)
```

Predicting the output of the first element of the test set

```
print(logisticRegr.predict(x_test[0].reshape(1,-1)))
```

```
[4]
```

Predicting the output of the first 10 elements of the test set

```
logisticRegr.predict(x_test[0:10])
```

```
array([4, 0, 9, 1, 8, 7, 1, 5, 1, 6])
```

©Simplilearn. All rights reserved.

Use Case – Implementation

Predicting for the entire dataset

```
predictions = logisticRegr.predict(x_test)
```

Determining the accuracy of the model

```
score = logisticRegr.score(x_test, y_test)
print(score)
```

```
0.9420289855072463
```

Representing the confusion matrix in a heat map

```
plt.figure(figsize=(9,9))
sns.heatmap(cm, annot=True, fmt=".3f", linewidths=.5, square = True, cmap = 'Blues_r');
plt.ylabel('Actual label');
plt.xlabel('Predicted label');
all_sample_title = 'Accuracy Score: {0}'.format(score)
plt.title(all_sample_title, size = 15);
```

©Simplilearn. All rights reserved.

Use Case – Implementation

Accurately predicting the image to contain a zero

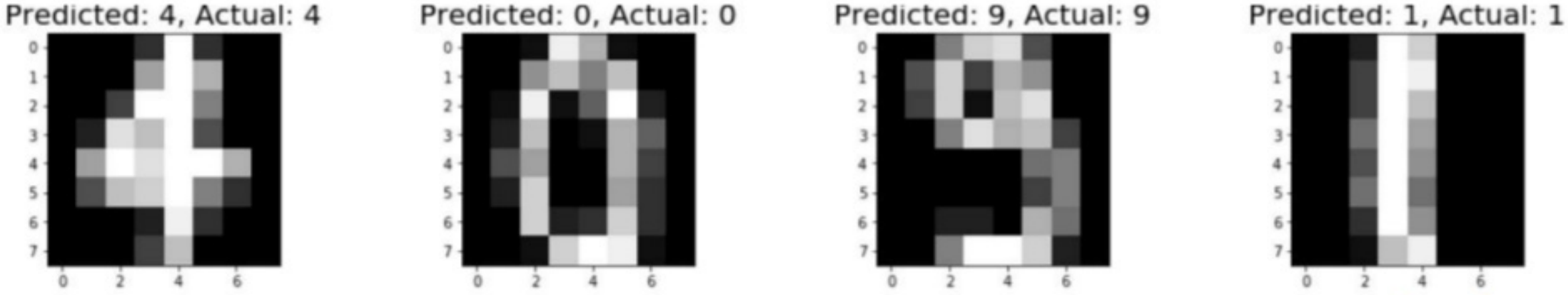
Inaccurately predicting the image to contain a seven

©Simplilearn. All rights reserved.

Use Case – Implementation

Presenting predictions and actual output

```
index = 0
misclassifiedIndex = []
for predict, actual in zip(predictions, y_test):
    if predict==actual:
        misclassifiedIndex.append(index)
    index +=1
plt.figure(figsize=(20,3))
for plotIndex, wrong in enumerate(misclassifiedIndex[0:4]):
    plt.subplot(1,4, plotIndex +1)
    plt.imshow(np.reshape(x_test[wrong], (8,8)),cmap=plt.cm.gray)
    plt.title("Predicted: {}, Actual: {}" .format(predictions[wrong], y_test[wrong]), fontsize=20)
```

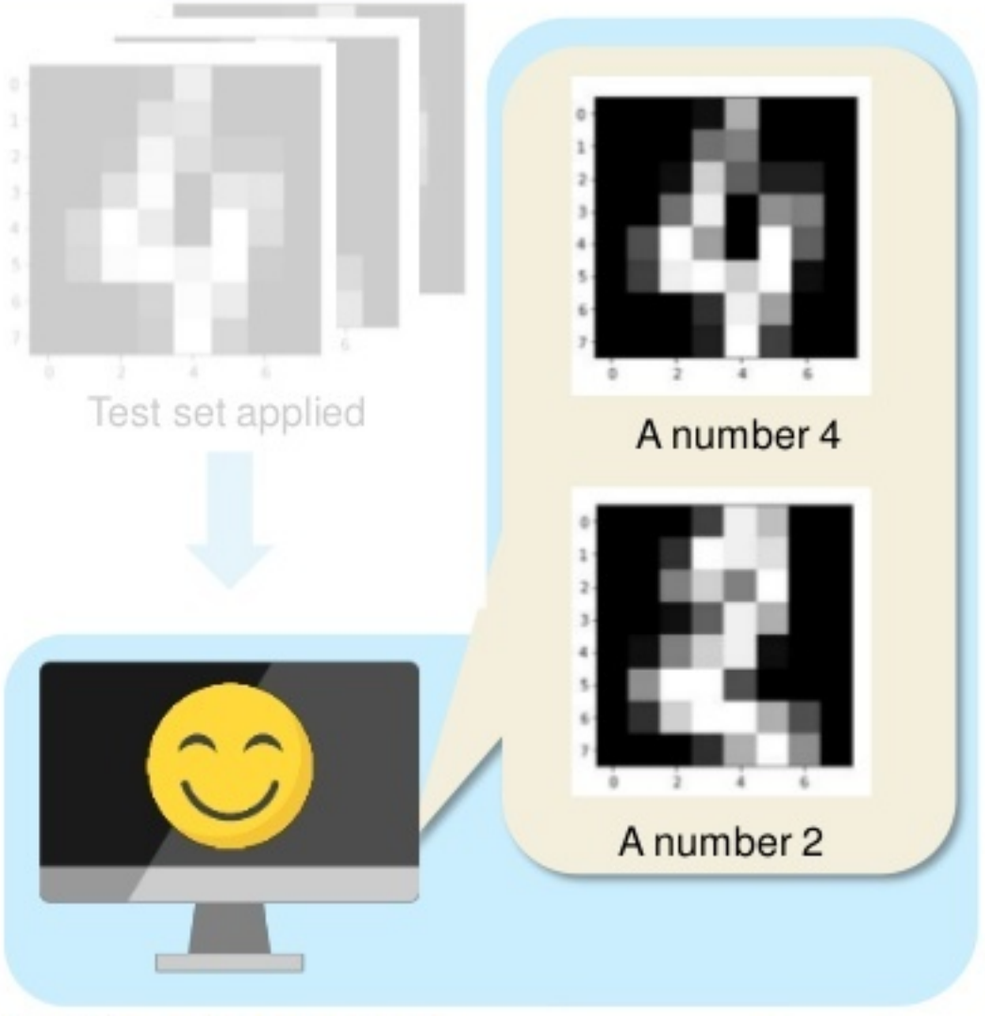


©Simplilearn. All rights reserved.

Use Case – Predicting numbers

Dividing the data set

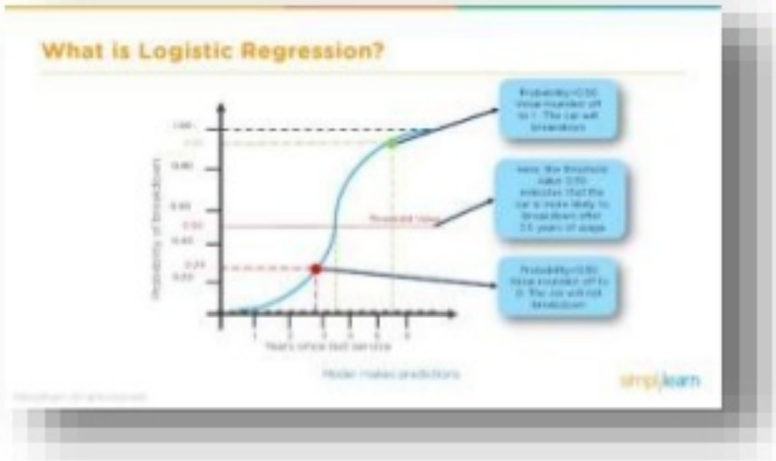
The model being trained

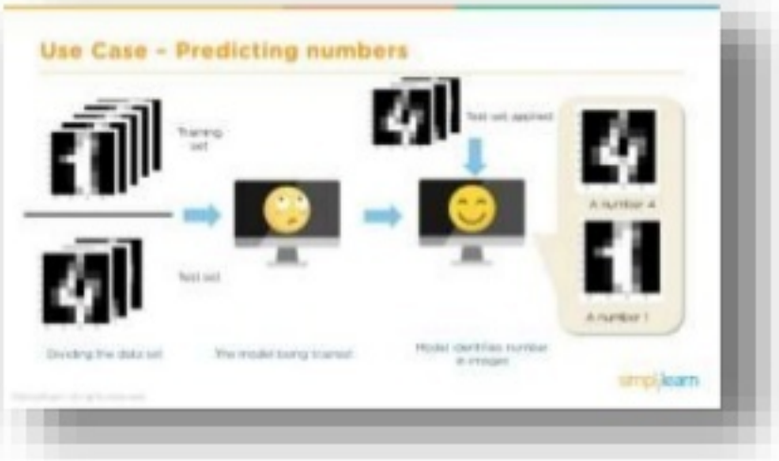


Model identifies number in images

©Simplilearn. All rights reserved.

Here’s What You’ve Learnt so Far





©Simplilearn. All rights reserved.

THANK YOU
For more information, visit
www.simplilearn.com
simplilearn