TechTalent Academy
Safeguarding Policy

*"Protecting an adult's right to live in **safety, free from abuse and neglect**. It is about people and organisations working together to **prevent and stop both the risks and experience of abuse or neglect**, while at the same time making sure that the **adult's wellbeing is promoted** including, where appropriate, having regard to their views, wishes, feelings and beliefs in deciding on any action. This must recognise that adults sometimes have complex interpersonal relationships and may be ambivalent, unclear or unrealistic about their personal circumstances."*

If you have a safeguarding concern, please raise this with your tutor or via the safeguarding link on our website:

https://www.techtalent.co.uk/safeguarding-statement

TechTalent's safeguarding lead is: **Max Ruddock**

techtalent academy

Starter Activity.

Elon Musk and experts say AI development should be paused immediately

The letter, signed by more than 1,000 people, warned of potential risks to society and civilisation by human-competitive AI systems in the form of economic and political disruptions.

https://news.sky.com/story/elon-musk-and-others-sign-open-letter-calling-for-pause-on-ai-development-12845039

techtalent academy

TechTalent Academy

Data Science Course

Deep Learning

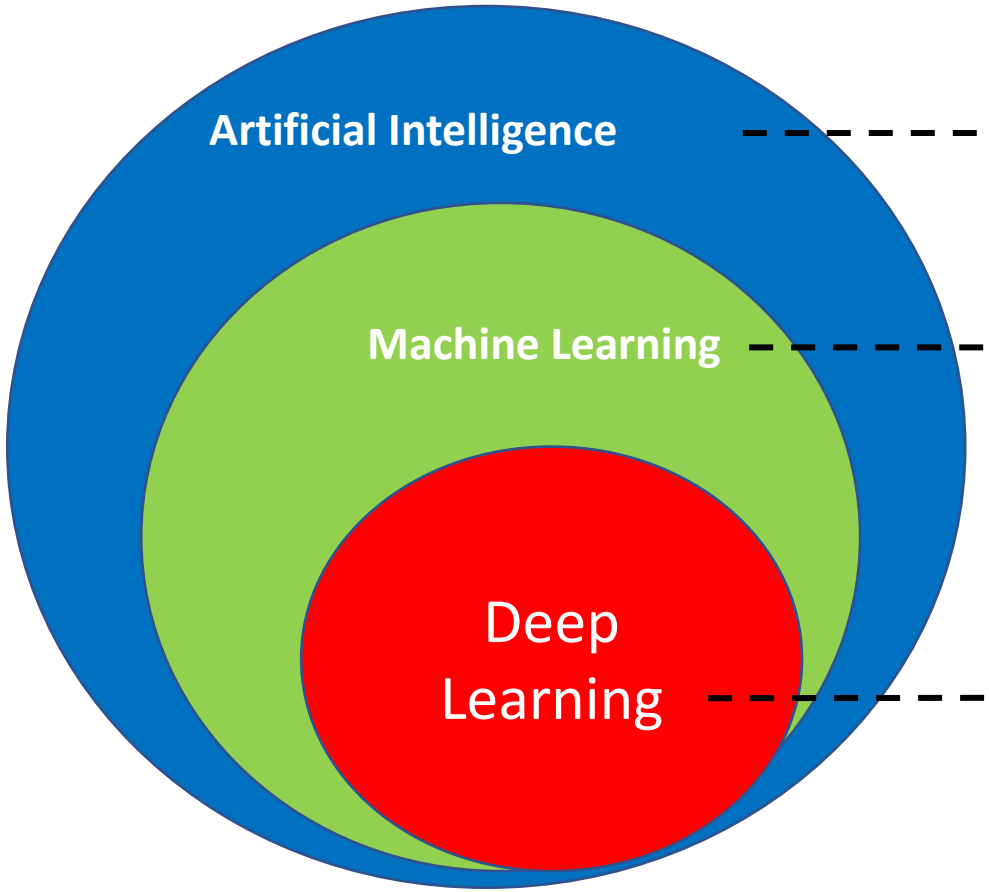
techtalent academy

Lesson Objectives.

- Deep Learning
- Artificial Neural Networks
- What is TensorFlow?

Deep Learning.

Deep learning is a subset of Machine learning



Based on human behaviour/intelligence

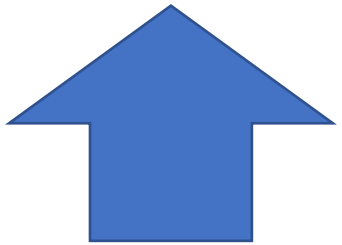
Based on data-autonomous learning

Based on Big data- structure like brain algorithm

techtalent academy

What is Machine Learning?

“Machine learning is a subfield of artificial intelligence, which is broadly defined as the capability of a machine to imitate intelligent human behaviour. Artificial intelligence systems are used to perform complex tasks in a way that is similar to how humans solve problems.”

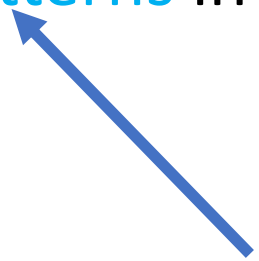


There a lot going on there

techtalent academy

What is Deep Learning?

"Machine learning is turning things (Big data) Into numbers and finding patterns in those numbers"

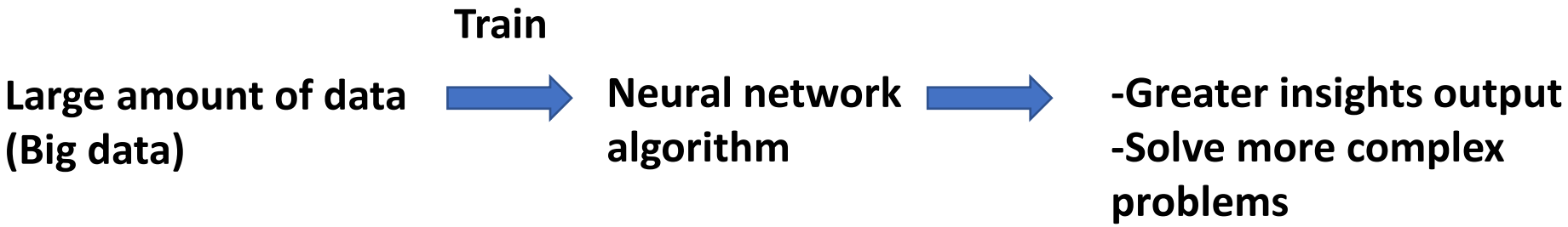


The computer does the start.
How?
Coding and maths
We're going to be writing the code or the Deep learning code with TensorFlow

techtalent academy

Deep Learning Concepts.

- **Definition: Universal technique to solve a larger set of problems**
- **Neural Networks combined with large dataset**
- **Application**: involves the **training** of large Neural Networks to process and analyse vast amounts of data to derive greater insight and to solve more complex problems



techtalent academy

What is Deep Learning?

Traditional programming

Input

Rules

1. Cut vegetables
2. Season chicken
3. Preheat oven
4. Cook chicken for 30-minutes
5. Add vegetables

Output

Machine Learning

Input

Output

Rules

1. Cut vegetables
2. Season chicken
3. Preheat oven
4. Cook chicken for 30-minutes
5. Add vegetables

techtalent academy

Why use Machine or Deep Learning?

- For a complex problem like (Say a self driving car) can you think of all the rules ?

techtalent academy

When Should You Not Use ML or DL?

- “If you can build a system that doesn’t require machine learning do that.”

Best Practices for ML Engineering ---- *Martin Zinkevich* Google

techtalent academy

What Is Deep Learning Good for?

- **Problems with long lists of rules – Machine learning/deep learning may help when the traditional approach fails.**

- **Continually changing environments – deep learning can adapt ('learn') to new scenarios**
- **Discovering insights within large collections of data – can you imagine trying to handcraft rules for what 101 different kinds of food look like?**

techtalent academy

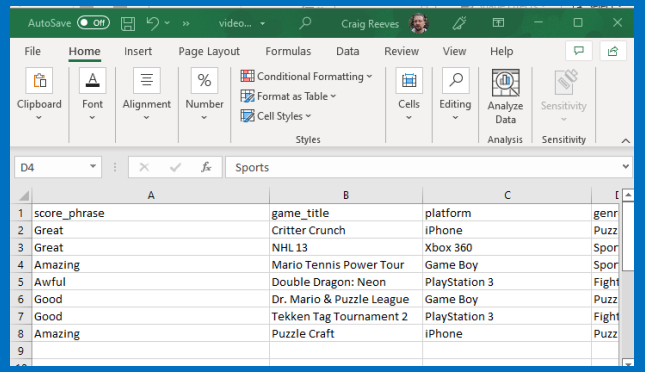
What Is Deep Learning Not Good for?

- **When you need explainability – the patterns learned by a deep learning model are typically uninterpretable by a human**
- **When the traditional approach is a better option– if you can accomplish what you need with a simple rule-based system.**
- **When errors are unacceptable – The outputs of a deep learning model aren't always predictable**
- **When you don't have much data– Deep learning models usually require a fairly large amount of data to produce good results. (We will see how to get good results without large amounts of data)**

techtalent academy

Machine Learning vs. Deep Learning.

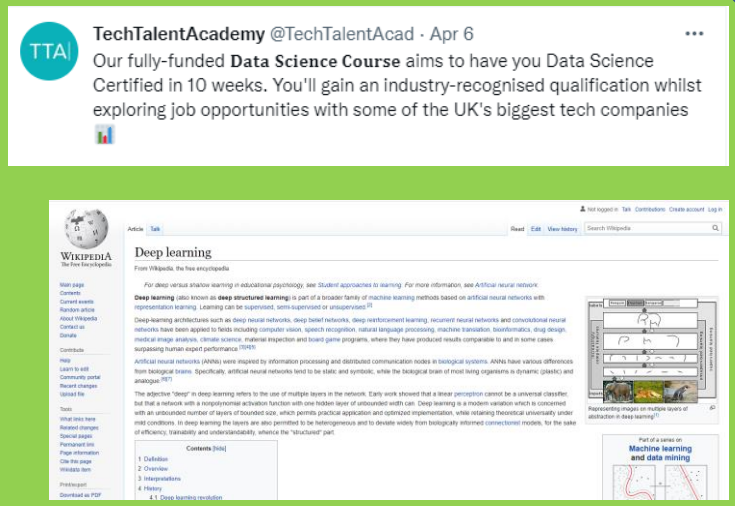
Machine Learning



	A	B	C	D
1	score_phrase	game_title	platform	genr
2	Great	Critter Crunch	iPhone	Puzz
3	Great	NHL 13	Xbox 360	Spor
4	Amazing	Mario Tennis Power Tour	Game Boy	Spor
5	Awful	Double Dragon: Neon	PlayStation 3	Fight
6	Good	Dr. Mario & Puzzle League	Game Boy	Puzz
7	Good	Tekken Tag Tournament 2	PlayStation 3	Fight
8	Amazing	Puzzle Craft	iPhone	Puzz
9				

Structured data

Deep Learning



unstructured data

techtalent academy

Common Algorithms In ML & DP.

- Random forest
- Naïve Bayes
- Nearest neighbour
- Support vector machine
- …… Many more

Structured data

- Neural networks
- Fully connected neural network
- Convolutional neural network
- Recurrent neural network
- Transformer (this is new)
- …… Many more

Unstructured data

techtalent academy

Artificial Neural Networks.

- *Deep learning* is an advanced form of machine learning that tries to emulate the way the human brain learns.

- In your brain, you have nerve cells called neurons, which are connected to one another by nerve extensions that pass electrochemical signals through the network.

- When the first neuron in the network is stimulated, the input signal is processed, and if it exceeds a particular threshold, the neuron is *activated* and passes the signal on to the neurons to which it is connected.

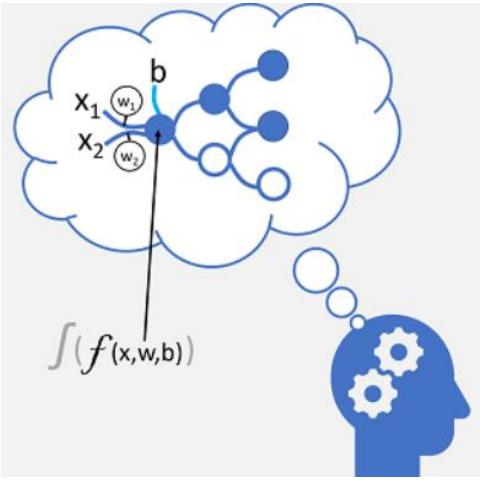
- These neurons in turn may be activated and pass the signal on through the rest of the network. Over time, the connections between the neurons are strengthened by frequent use as you learn how to respond effectively.

- For example, if someone throws a ball towards you, your neuron connections enable you to process the visual information and coordinate your movements to catch the ball. If you perform this action repeatedly, the network of neurons involved in catching a ball will grow stronger as you learn how to be better at catching a ball.

techtalent academy

Artificial Neural Networks.

- Deep learning emulates this biological process using artificial neural networks that process numeric inputs rather than electrochemical stimuli.

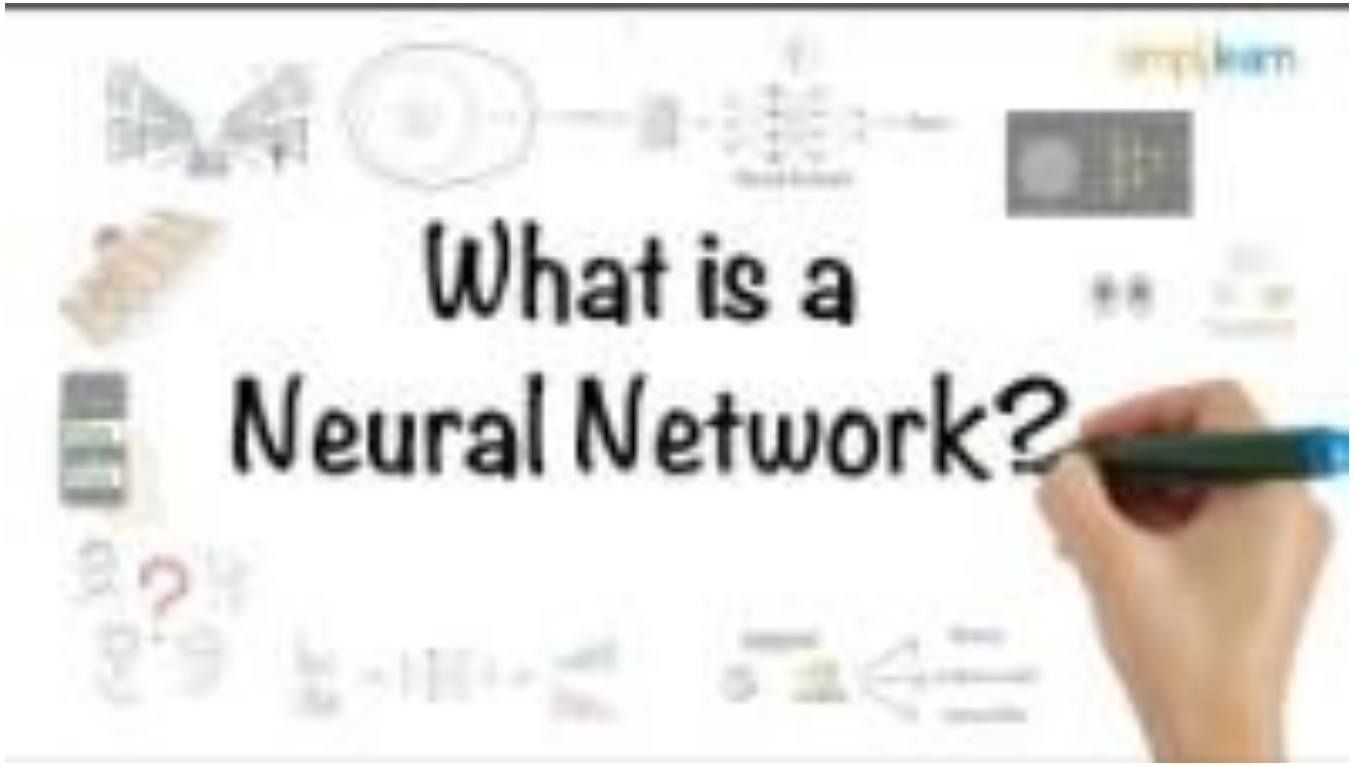


- The incoming nerve connections are replaced by numeric inputs that are typically identified as ***x***. When there's more than one input value, ***x*** is considered a vector with elements named x_1, x_2, and so on.

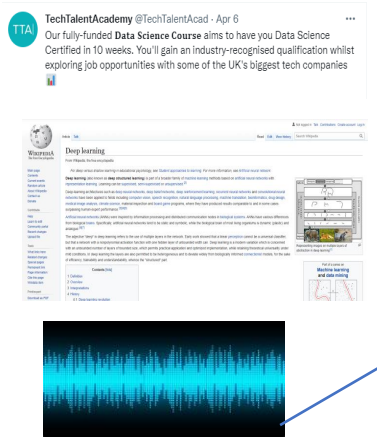
- Associated with each ***x*** value is a *weight* (***w***), which is used to strengthen or weaken the effect of the ***x*** value to simulate learning. Additionally, a *bias* (***b***) input is added to enable fine-grained control over the network. During the training process, the ***w*** and ***b*** values will be adjusted to tune the network so that it "learns" to produce correct outputs.

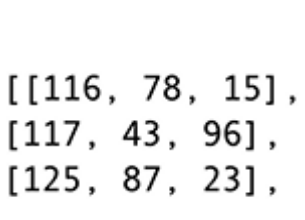
- The neuron itself encapsulates a function that calculates a weighted sum of ***x***, ***w***, and ***b***. This function is in turn enclosed in an *activation function* that constrains the result (often to a value between 0 and 1) to determine whether or not the neuron passes an output onto the next layer of neurons in the network.

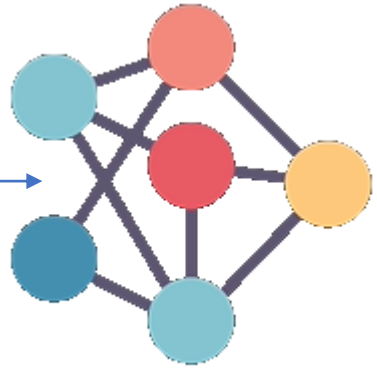
Overview of a Neural Network.

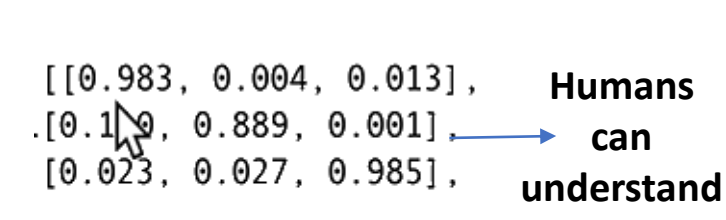


What Are Neural Networks?









Humans can understand

Inputs

Numerical encoding

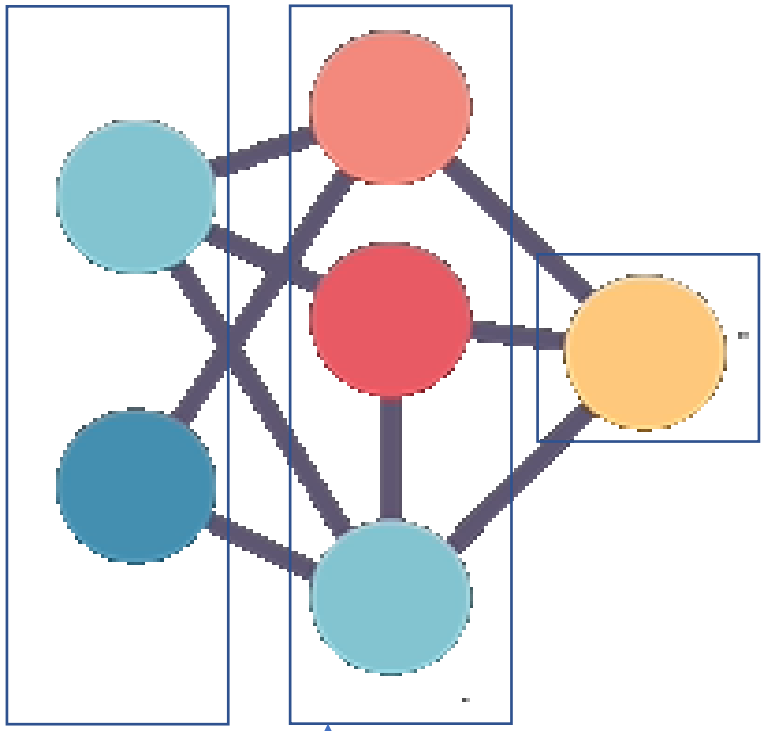
Learns representation (pattens/features/weights)

Representation output

Outputs

Anatomy Of Neural Networks.

Input Layer
(data goes here)
units/neurons =2

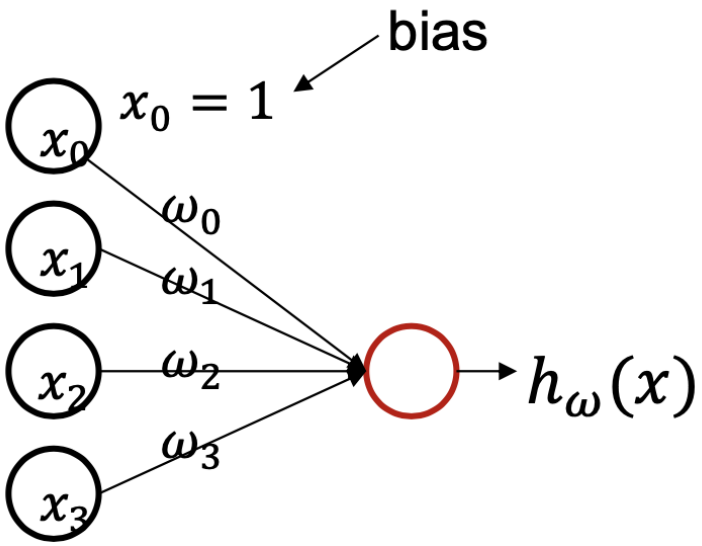


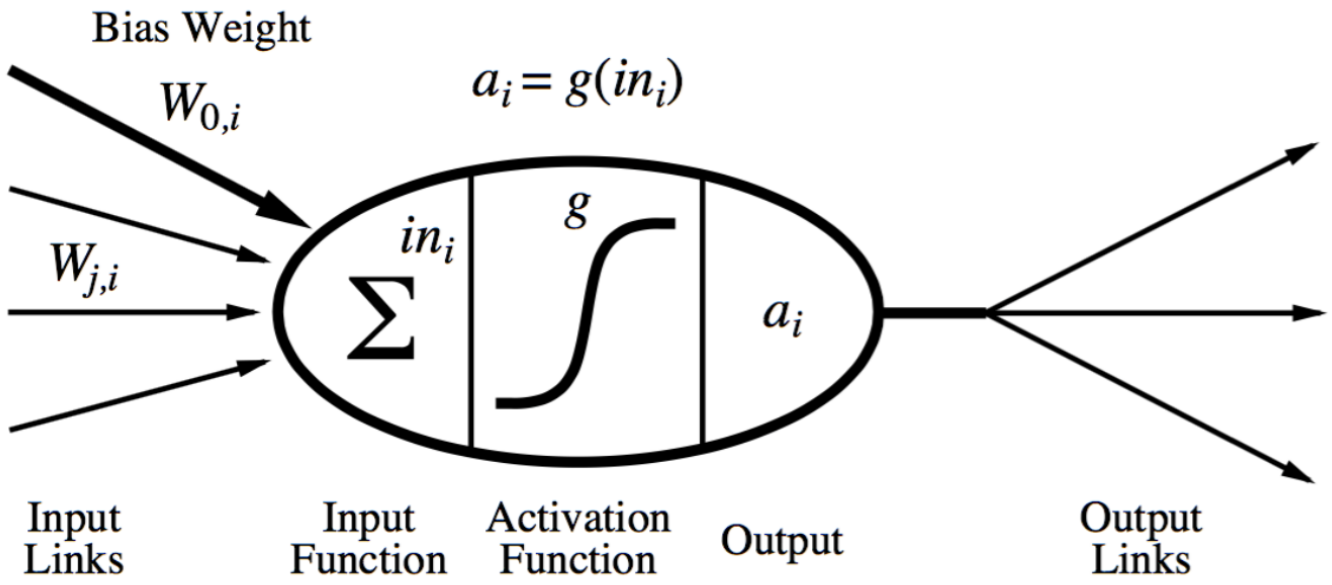
Output Layer
units/neurons =1

Hidden Layer
(Learns patterns in data+)
units/neurons =3

techtalent academy

Inside Each Neuron.





techtalent academy

Forward and Back Propagation.

- Forward:
  - At first Neurons weights are set at random.
  - The training ANN will then make predictions with the training data.
  - Once it gets to the output, it will check how well it has done.
  - With the results it will then start to do back propagation

- Back:
  - The ANN will go back through the ANN and update the weights to a value it thinks might be better to help it predict its label.

- This is repeat multiple times depending on the epochs value that has been set.

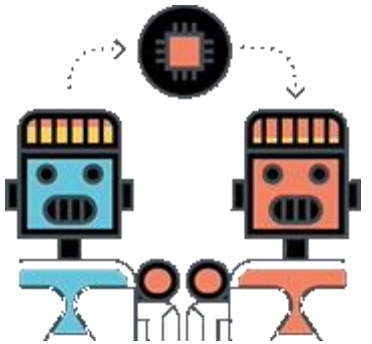
techtalent academy

Types Of Learning.

Supervised Learning

Semi-Supervised Learning

Un-Supervised Learning



Transfer Learning

techtalent academy

What Is Deep Learning Used For?

- ML can be used for literally anything as long as you convert it into numbers and program it to find patterns, Here are some use cases.
- Recommendation – Netflix, Youtube, Amazon
- Translation
- Speech recognition (Google, Siri, Alexa)
- Computer Vision (search for a red jumper in google photos)
- Spam detection

techtalent academy

What Is Deep Learning Used For?

3 main categories for deep learning

- Recommendation---- Netflix, Amazon, YouTube
- Sequence to sequence (seq2seq)
- Classification/regression

techtalent academy

What is TensorFlow?

- Open-source Python library for Deep learning
- You can write fast deep learning code in Python and other languages
- Able to access many pre-build learning models (TensorFlow Hub)
- Whole stack: Pre-process data, Model data, deploy model in your applications
- Original author: Google

TensorFlow

techtalent academy

Why TensorFlow?

Easy model building

Build and train ML models easily using intuitive high-level APIs like Keras with eager execution, which makes for immediate model iteration and easy debugging.

Robust ML production anywhere

Easily train and deploy models in the cloud, on-prem, in the browser, or on-device no matter what language you use.

Powerful experimentation for research

A simple and flexible architecture to take new ideas from concept to code, to state-of-the-art models, and to publication faster.

Source: TensorFlow.org

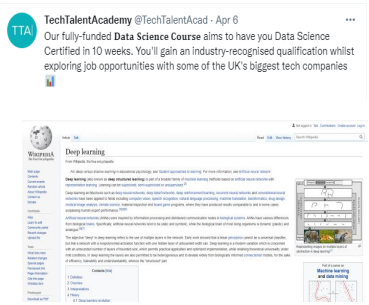
TensorFlow Fundamentals.

techtalent academy

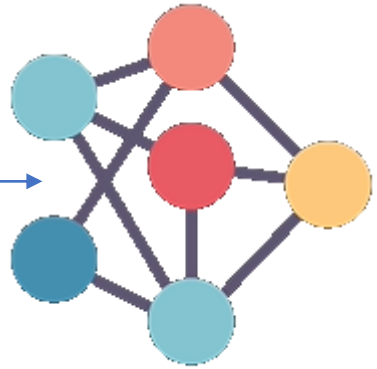
The code is NOT needed for the exam! It is just for demonstration purposes!

What Is A Tensor?

These are tensors !



```
[[116, 78, 15],
[117, 43, 96],
[125, 87, 23],
```



```
[[0.983, 0.004, 0.013],
[0.1  , 0.889, 0.001],
[0.023, 0.027, 0.985],
...,
```

Humans can understand

Inputs	Numerical encoding	Learns representation (pattens/features/weights)	Representation output	Outputs

https://www.youtube.com/watch?v=L35fFDpwIM4

techtalent academy

What Are We Going To Cover.

- TensorFlow Basics & Fundamentals
- Pre-processing data (getting it into tensors)
- Process data
- Using Data Models
- Fitting models to data (learning patterns)
- Making predictions with a model
- Evaluating model predictions
- Saving and loading models
- Using a training model to make predictions on custom data

techtalent academy

TensorFlow Workflow.

- Get the data ready (turn into tensors)
- Build or pick a pre-trained model (to suit the problem)
- Fit the model to the data and make a prediction
- Evaluate the model
- Improve through experimentation
- Save and reload your trained model

techtalent academy

Loading your Dataset.

```
:  import pandas            as pd
   import numpy             as np
   import matplotlib.pyplot as plt
```

```
white = pd.read_csv("http://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-white.csv", sep=";")
```

```
white.head()
```

```
white.tail(25)
```

```
red = pd.read_csv("http://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-red.csv", sep=";")
```

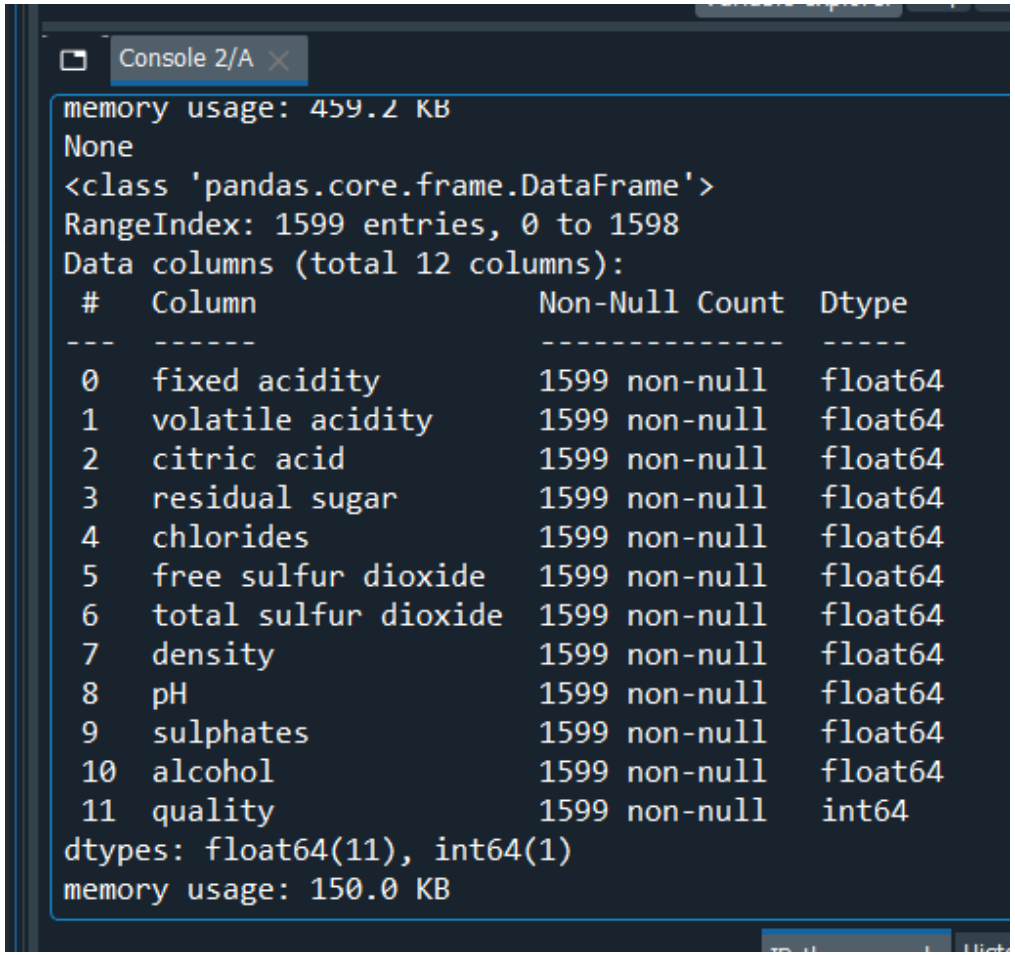
```
red.head()
```

Run a simple .head() .tail() to ensure your import was successful

techtalent academy

Data Exploration.

```
1   # Print info on white wine
2   print(white.info())
3
4   # Print info on red wine
5   print(red.info())
```



techtalent academy

In [4]:
```
# First rows of `red`
red.head()

# Last rows of `white`
white.tail()

# Take a sample of 5 rows of `red`
red.sample(5)

# Describe `white`
white.describe()

# Double check for null values in `red`
pd.isnull(red)
```

Out[4]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	False	False	False	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...
1594	False	False	False	False	False	False	False	False	False	False	False	False
1595	False	False	False	False	False	False	False	False	False	False	False	False
1596	False	False	False	False	False	False	False	False	False	False	False	False
1597	False	False	False	False	False	False	False	False	False	False	False	False
1598	False	False	False	False	False	False	False	False	False	False	False	False

1599 rows × 12 columns

techtalent academy

Visualising The Data –Levels Of Alcohol.

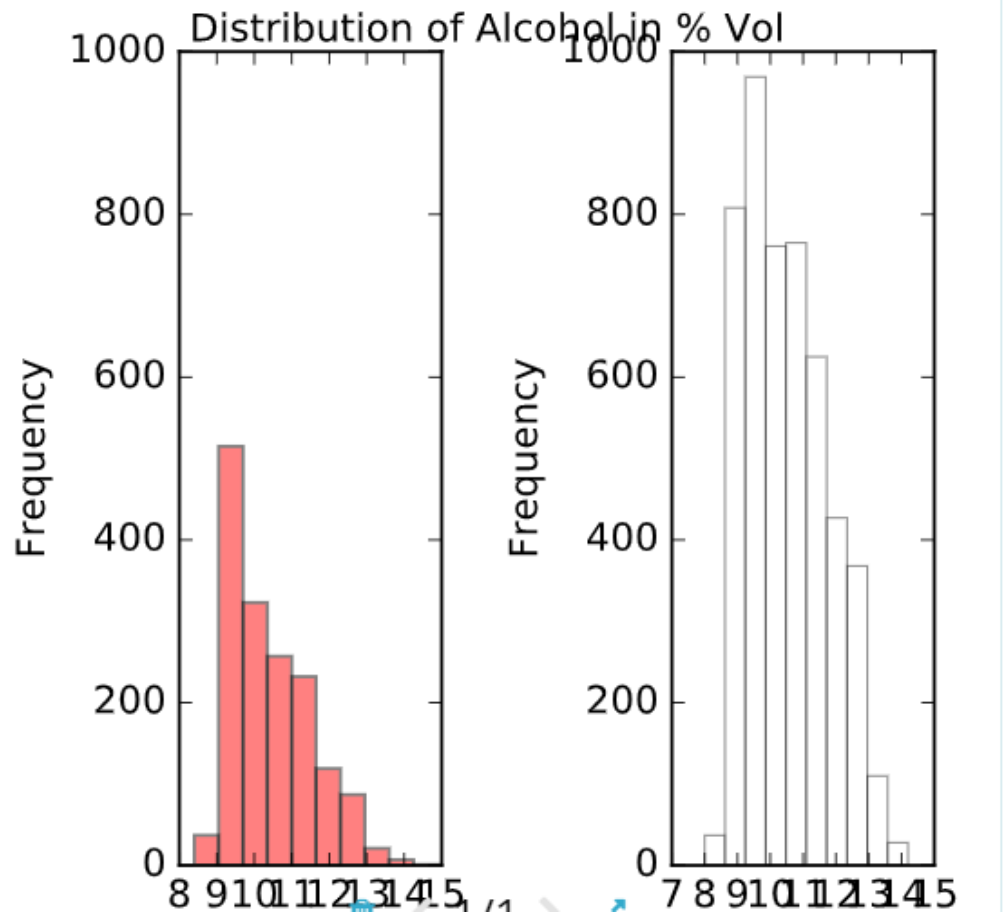
```
import matplotlib.pyplot as plt

fig, ax = plt.subplots(1, 2)

ax[0].hist(red.alcohol, 10, facecolor='red', alpha=0.5,
label="Red wine")
ax[1].hist(white.alcohol, 10, facecolor='white', ec
="black", lw=0.5, alpha=0.5, label="White wine")

fig.subplots_adjust(left=0, right=1, bottom=0, top=0.5,
hspace=0.05, wspace=1)
ax[0].set_ylim([0, 1000])
ax[0].set_xlabel("Alcohol in % Vol")
ax[0].set_ylabel("Frequency")
ax[1].set_xlabel("Alcohol in % Vol")
ax[1].set_ylabel("Frequency")
#ax[0].legend(loc='best')
#ax[1].legend(loc='best')
fig.suptitle("Distribution of Alcohol in % Vol")

plt.show()
```



techtalent academy

Relation Between The Sulphates And The Quality Of The Wine.

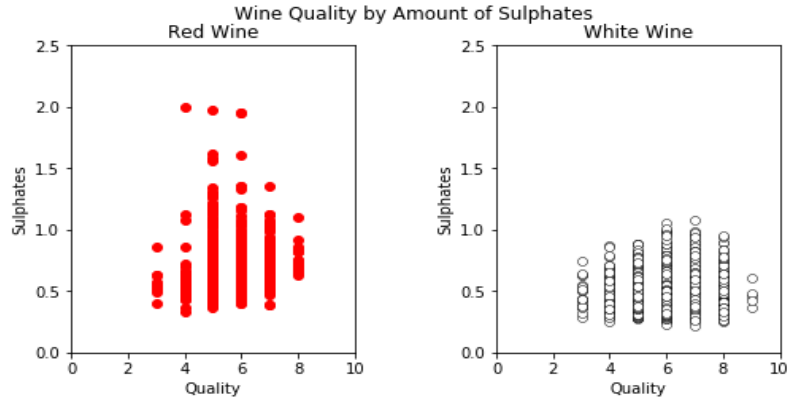
In [7]:
```
import matplotlib.pyplot as plt

fig, ax = plt.subplots(1, 2, figsize=(8, 4))

ax[0].scatter(red['quality'], red["sulphates"], color="red")
ax[1].scatter(white['quality'], white['sulphates'], color="white", edgecolors="black", lw=0.5)

ax[0].set_title("Red Wine")
ax[1].set_title("White Wine")
ax[0].set_xlabel("Quality")
ax[1].set_xlabel("Quality")
ax[0].set_ylabel("Sulphates")
ax[1].set_ylabel("Sulphates")
ax[0].set_xlim([0,10])
ax[1].set_xlim([0,10])
ax[0].set_ylim([0,2.5])
ax[1].set_ylim([0,2.5])
fig.subplots_adjust(wspace=0.5)
fig.suptitle("Wine Quality by Amount of Sulphates")

plt.show()
```



techtalent academy

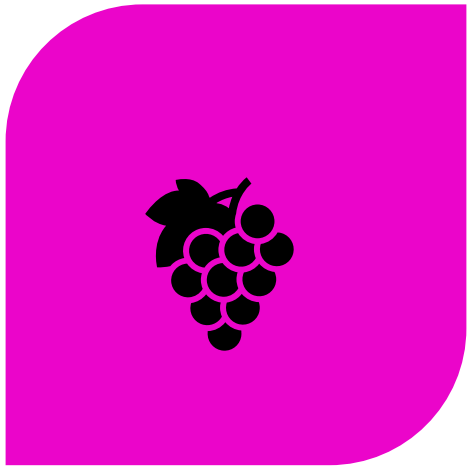
Wrapping Up The Exploratory Data Analysis (EDA).

Recap of what has been seen during your EDA that could be important :

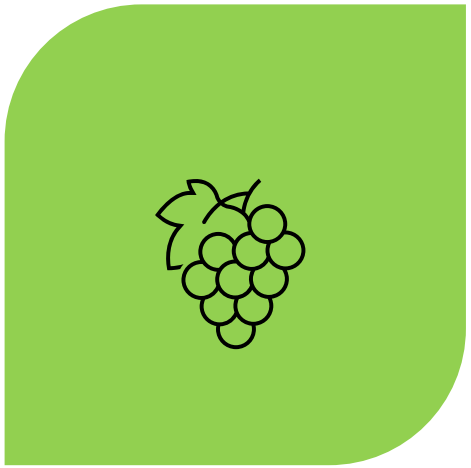
- Some of the variables of your data sets have values that are significantly far apart.
- You have an ideal scenario: there are no null values in the data sets.
- Most wines included in the data set have around 9% of alcohol.
- Red wine seems to contain more sulphates than the white wine, which has less sulphates above 1 g/dm3.

techtalent academy

Predicting Wine Types - Red or White?.



PRE-PROCESS/TEST



INTERMEZZO: CORRELATION MATRIX

techtalent academy

Pre-process Data.

We have explored the data, lets now look at the insights we have gained!

First we need to pre-process the data in order to start building our own neural network!

In [11]:
```
# Add `type` column to `red` with value 1
red['type'] = 1

# Add `type` column to `white` with value 0
white['type'] = 0

# Append `white` to `red`
wines = red.append(white, ignore_index=True)
print (wines)
```

```
      fixed acidity  volatile acidity  citric acid  residual sugar  chlorides  \
0               7.4              0.70         0.00             1.9      0.076
1               7.8              0.88         0.00             2.6      0.098
2               7.8              0.76         0.04             2.3      0.092
3              11.2              0.28         0.56             1.9      0.075
4               7.4              0.70         0.00             1.9      0.076
...             ...               ...          ...             ...        ...
6492            6.2              0.21         0.29             1.6      0.039
6493            6.6              0.32         0.36             8.0      0.047
6494            6.5              0.24         0.19             1.2      0.041
6495            5.5              0.29         0.30             1.1      0.022
6496            6.0              0.21         0.38             0.8      0.020

      free sulfur dioxide  total sulfur dioxide  density    pH  sulphates  \
0                    11.0                  34.0  0.99780  3.51       0.56
1                    25.0                  67.0  0.99680  3.20       0.68
```

techtalent academy

Train and Test Sets.

```
]: from sklearn.model_selection import train_test_split
```

```
]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33, random_state=42)
   print(X_train.shape)
   print(X_test.shape)
```

For now, import the train_test_split from sklearn.model_selection and assign the data and the target labels to the variables X and y.

As we can see that we need to flatten the array of target labels in order to be totally ready to use the X and y variables as input for the train_test_split() function.

techtalent academy

Standardise The Data.

Standardisation is a way to deal with the values that lie so far apart.

The scikit-learn package offers you a great and fast way of getting your data standardised:

Import the Standard Scaler module from *sklearn.preprocessing* and you're ready to scale your train and test data!

```
[ ]: from sklearn.preprocessing import StandardScaler
```

```
[ ]: scalar = StandardScaler().fit(X_train)
     X_train = scalar.transform(X_train)
     X_test = scalar.transform(X_test)
```

```
[ ]: import tensorflow as tf

     from tensorflow               import keras
     from tensorflow.keras.models import Sequential
     from tensorflow.keras.layers import Dense
```

Now that we've pre-processed the data, we can move on to the real work: building our own neural network to classify wines.

techtalent academy

Multi-layer Perceptron.

A quick way to start building your multi-layer perceptron is to use the Keras Sequential model.

This is a linear stack of layers. You can easily create the model by passing a list of layer instances to the constructor, you set this up by running model = Sequential().

```
[ ]: model = Sequential()
```

creating the model's layers:

```
[ ]: # Input layer
model.add(Dense(12, activation='relu', input_shape=(12,)))

#Hidden layer
model.add(Dense(8, activation='relu'))

# Output label
model.add(Dense(1, activation='sigmoid'))
```

techtalent academy

Information From The Model.

```
[29]: print("Output shape:", model.output_shape)

Output shape: (None, 1)
```

```
[30]: model.summary()

Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 12)                156
_________________________________________________________________
dense_1 (Dense)              (None, 8)                 104
_________________________________________________________________
dense_2 (Dense)              (None, 1)                 9
=================================================================
Total params: 269
Trainable params: 269
Non-trainable params: 0
```

```
[31]: print("Model weights:", model.get_weights())

Model weights: [array([[ 0.03435898, -0.24827
         0.20286858, -0.32607782,  0.29336548
        -0.14217973,  0.4732268 ],
       [ 0.32511783,  0.4507426 ,  0.4869498
         0.03820753, -0.11768901, -0.4701327
         0.1668328 ,  0.28152716],
       [ 0.36597955, -0.21855116,  0.41177273
         0.10882175,  0.27422035, -0.47514927
        -0.29566026,  0.39479446],
       [ 0.41026127, -0.34389877, -0.24463332
        -0.39401162, -0.4610442 ,  0.28947282
```

At first these weights are randomly generated. As the model learns, the weights are assigned new values.

techtalent academy

Compile and Fit.

To compile your model and fit the model to the data, use: compile() and fit()

```
model.compile(loss='binary_crossentropy',
              optimizer='adam',
              metrics=['accuracy'])

model.fit(X_train, y_train,epochs=20, batch_size=1, verbose=1)
```

techtalent academy

Test Data Predictions.

```
y_pred = model.predict(X_test)
y_pred[:5]
```

```
array([[1.3831258e-04],
       [9.9752289e-01],
       [8.0875907e-06],
       [2.6916542e-08],
       [3.7994080e-10]], dtype=float32)
```

```
predictions = [1 if p > 0.5 else 0 for p in y_pred]
```

```
predictions[:15]
```

```
[0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 1]
```

```
y_test[:15]
```

```
array([0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 1], dtype=int64)
```

Now we can use this data to predict which is a red/white wine?
1 means it is likely to be a red wine and 0 it is likely to be a white wine!

techtalent academy

Evaluate Model.

```
score = model.evaluate(X_test, y_test, verbose=1)
score
```

```
68/68 [==============================] - 0s 932us/step - loss: 0.0272 - accuracy: 0.9944
[0.027239905670285225, 0.9944055676460266]
```

This is using the model just built using the test data to make a prediction using data the model hasn't used.

This evaluates the performance of the model, eg: how good the model is.

techtalent academy

Evaluation Metrics.

Before you start re-arranging the data and putting it together in a different way, it's always a good idea to try out different evaluation metrics.

For this, you can rely on scikit-learn (>> import as sklearn) for this. You will test out some basic classification evaluation techniques like:

- The confusion matrix – This is a breakdown of predictions into a table showing correct predictions and the types of incorrect predictions that have been made.
- You should only see numbers in the diagonal, which means that all your predictions were correct!
- Precision is a measure of a classifier's exactness. The higher the precision, the more accurate the classifier.

```
from sklearn.metrics import confusion_matrix, precision_score
```

```
confusion_matrix(y_test, predictions)
```

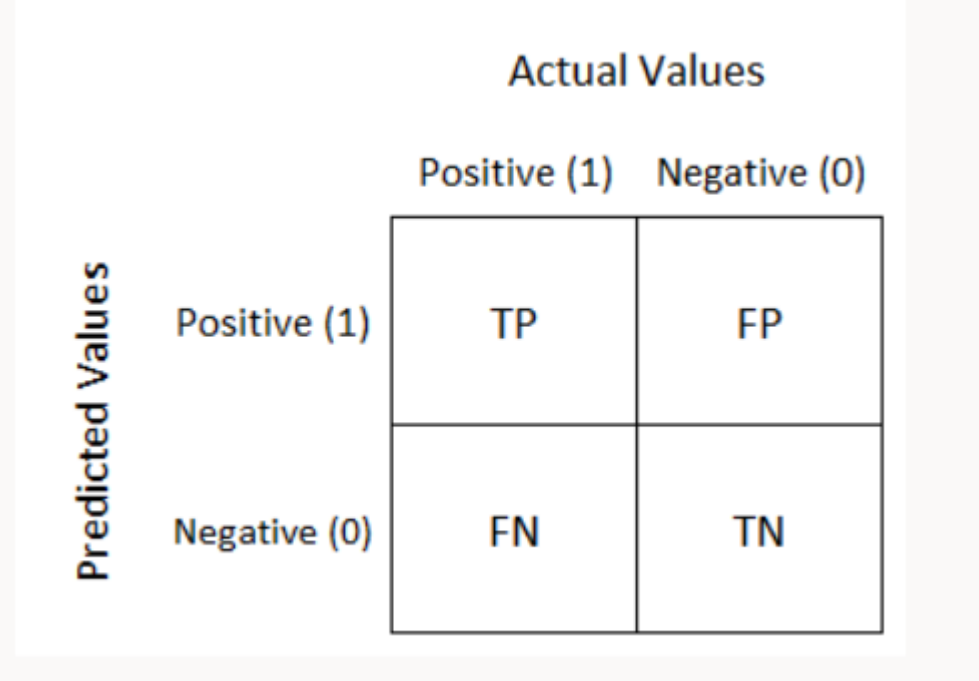
```
array([[1585,    3],
       [   9,  548]], dtype=int64)
```

```
precision_score(y_test, y_pred.round())
```

```
0.9945553539019963
```

techtalent academy

The Confusion Matrix.



Additional AI concepts for the BCS exam.

ML compliments Knowledge-Based Systems (KBS)

- KBS is a **form of AI** designed to capture human expertise/knowledge (within a knowledge base) and apply a set of rules to identify an outcome (through an inference engine)

- **Example:** driverless cars use KBS to make decisions and use Neural Network for image recognition for navigation using the car's camera

techtalent academy

Machine Learning Agent Role.

-Agent that **learn from experience**

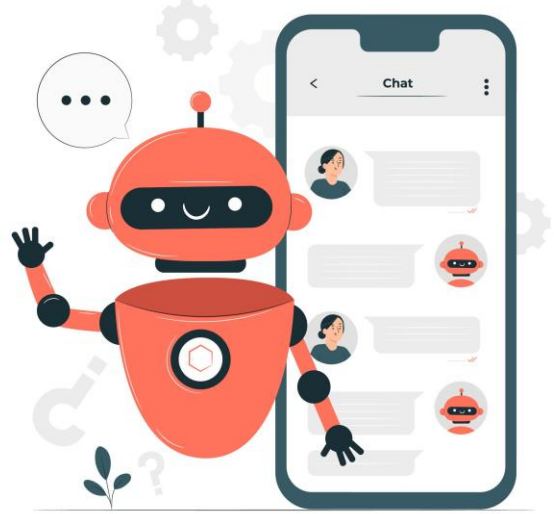
-It starts with some knowledge and adapt **automatically** through learning

-Commonly use in ML

-Designed to undertake specific tasks given specific data

-Though repetition of tasks they learn to improve each time

Examples include chat bots, driverless cars, facial recognition



techtalent academy

Plenary.

1. Deep learning involves the training of what type of algorithm?
2. Artificial networks are made of what type of layers?
3. What is the Python library used for using Deep learning algorithms?