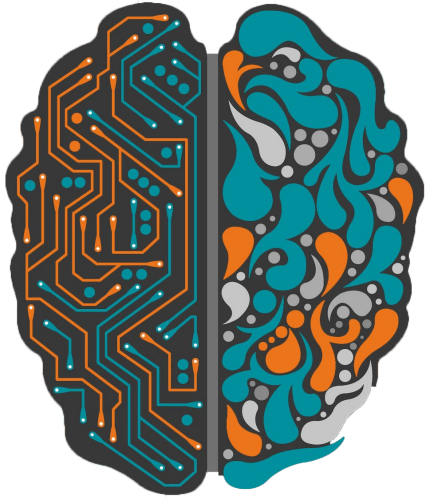
Windesheim Flevoland



Deep learning

Vincent van Dijk

9 januari 2019

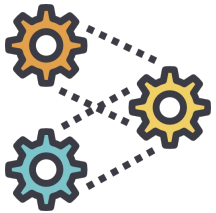
Windesheim Flevoland

Inhoud

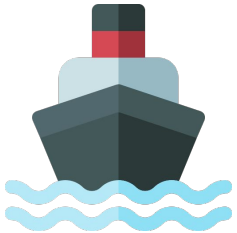
Why

Use

How

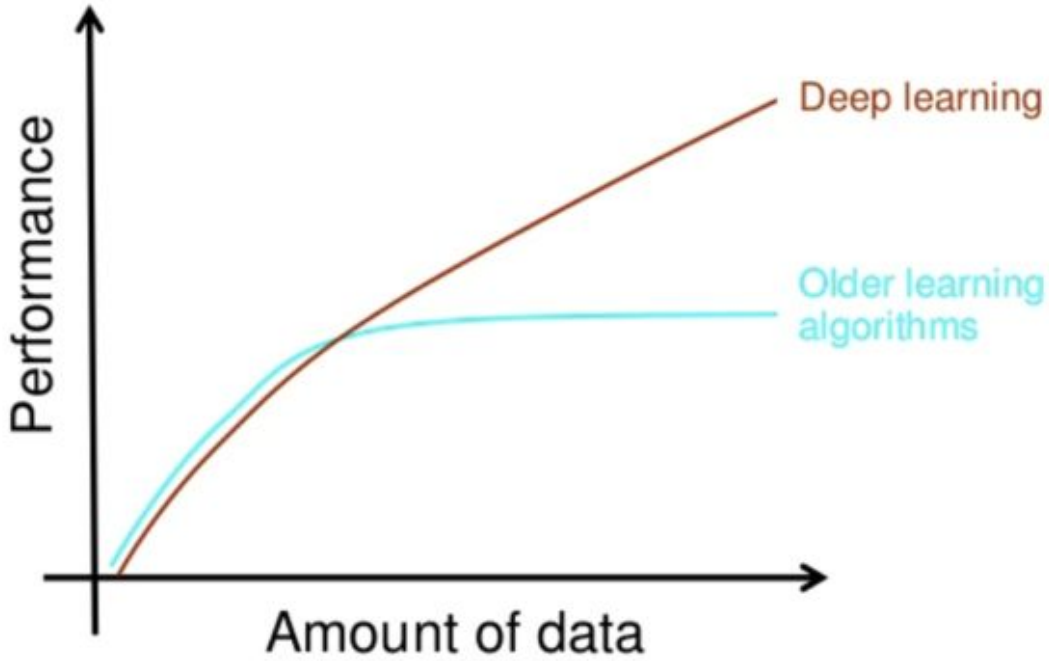


Titanic



Windesheim Flevoland

Why deep learning



Windesheim Flevoland

Use

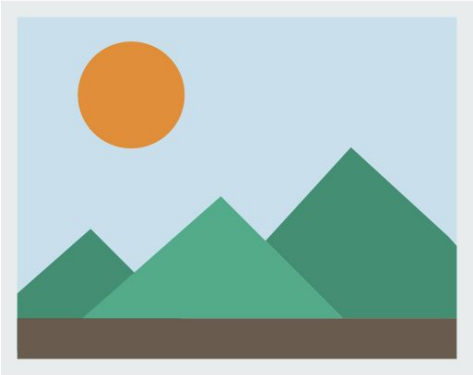


Image recognition

Reinforcement learning

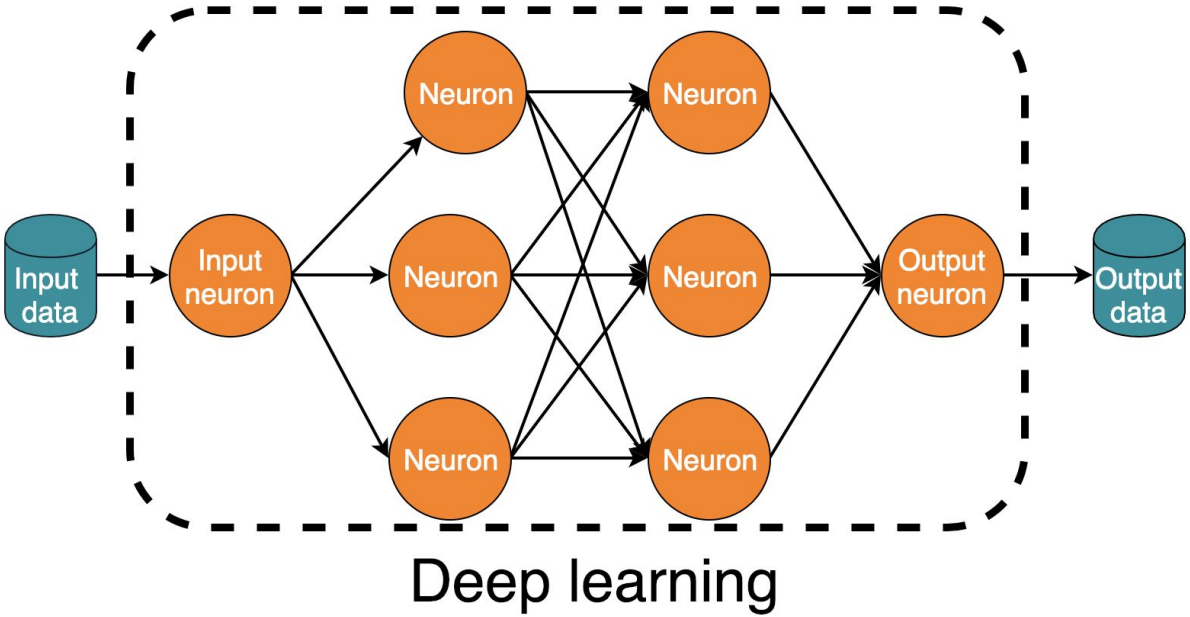
Voice recognition

Windesheim Flevoland

What I will explain

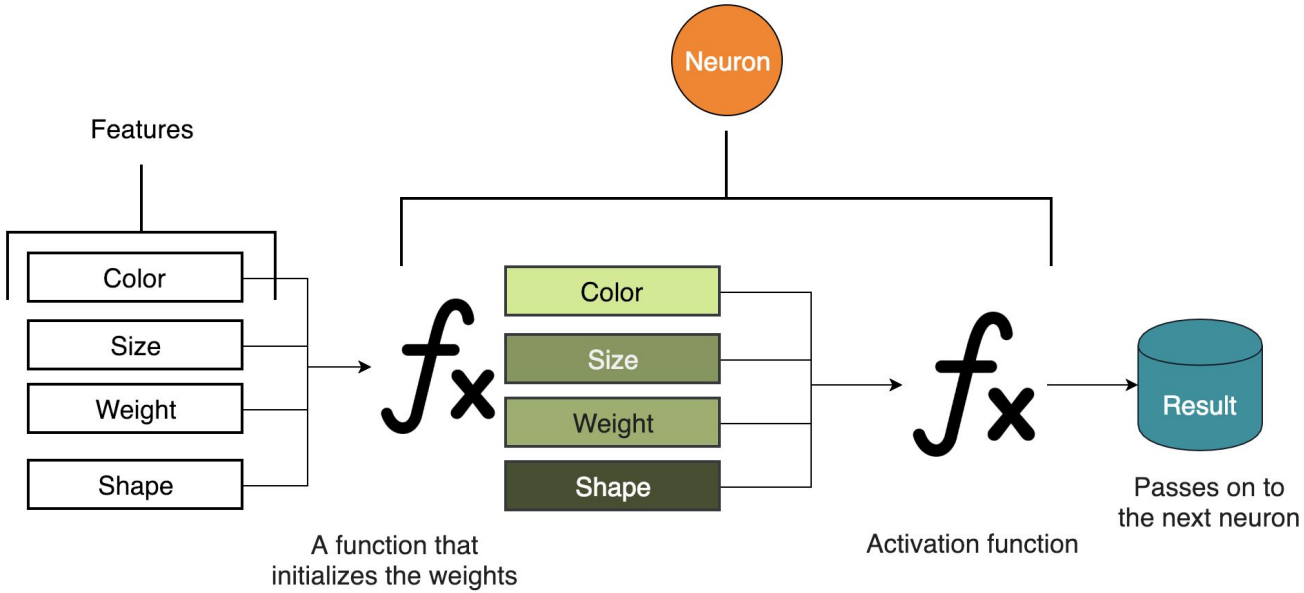
```
# Deeeeeep learning time
# Install: conda install keras
from keras.models import Sequential
from keras.layers import Dense

# Define the layers
model = Sequential()
model.add(Dense(32, activation='relu', input_dim=len(X.columns))) # Input layer

model.add(Dense(30, activation='relu')) # Hidden layer
model.add(Dense(30, activation='relu')) # Hidden layer

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


# Compile the layers
model.compile(
              optimizer='adam', # The optimize algorithm
              loss='binary_crossentropy', # Because of our predict a binary (1=survived, 0=died)
              metrics=['accuracy'] # Aiming for the best accuracy
        )
```

Network

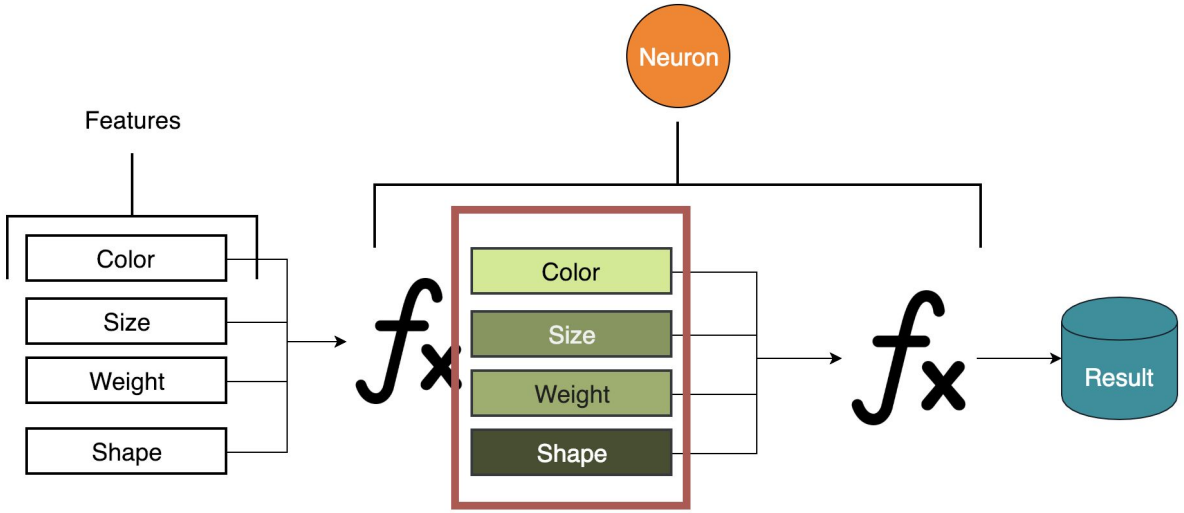

Input data
Input neuron
Neuron
Neuron
Neuron
Neuron
Neuron
Neuron
Output neuron
Output data
Deep learning

Windesheim Flevoland

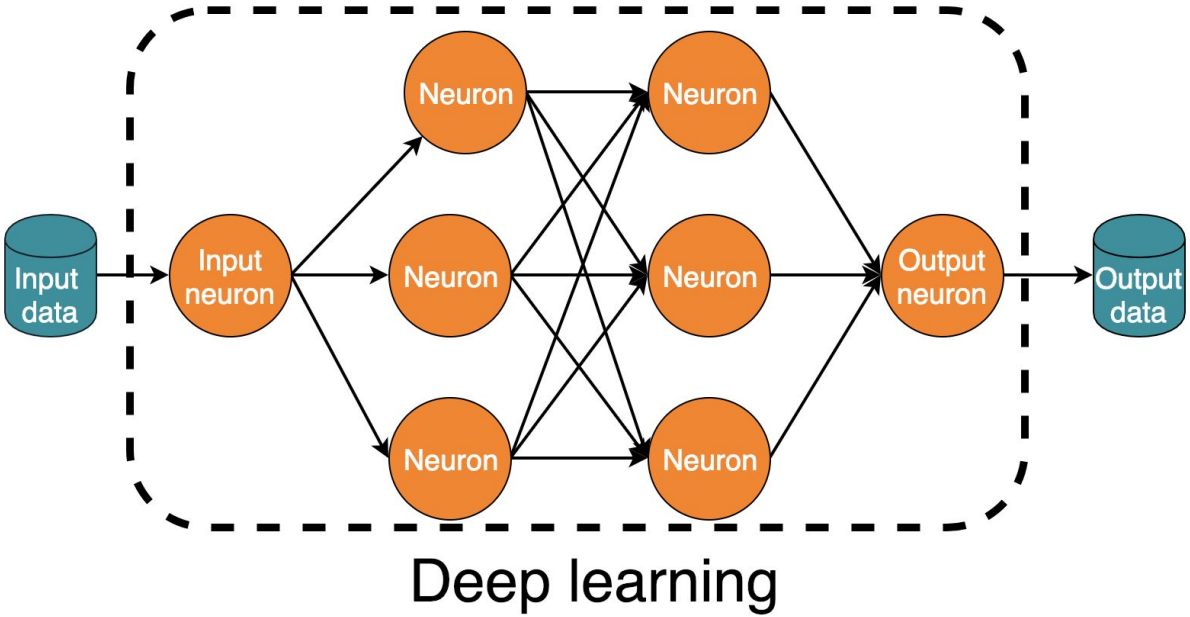
Neuron



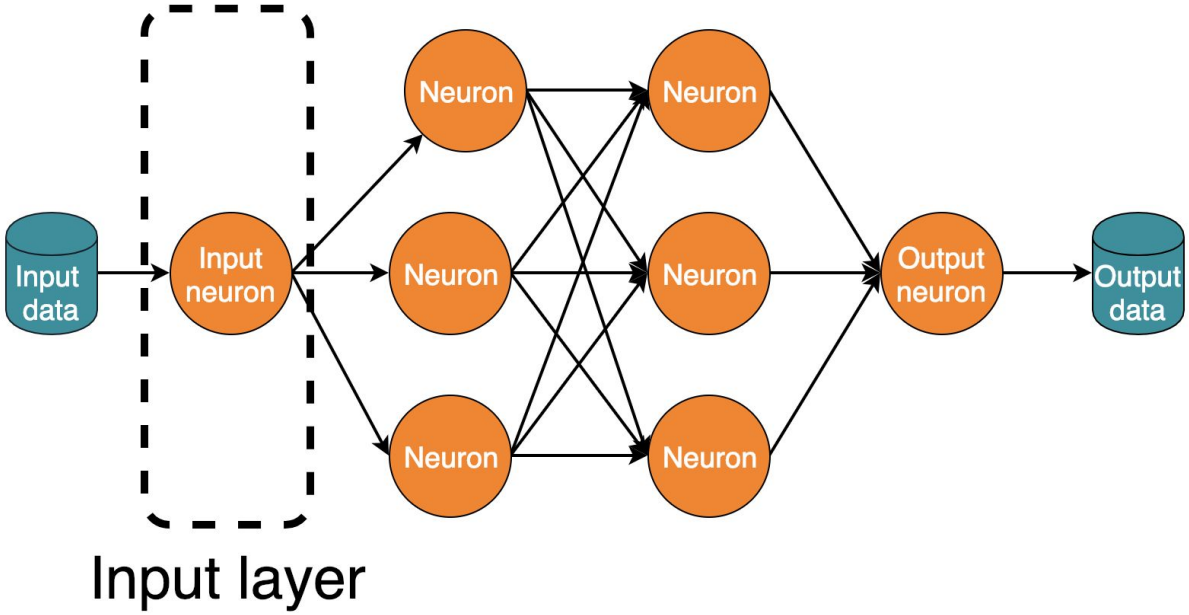
Algorithm


Neuron
Features
Color
Size
Weight
Shape
Color
Size
Weight
Shape
Result
The magic of the deep learning algorithm

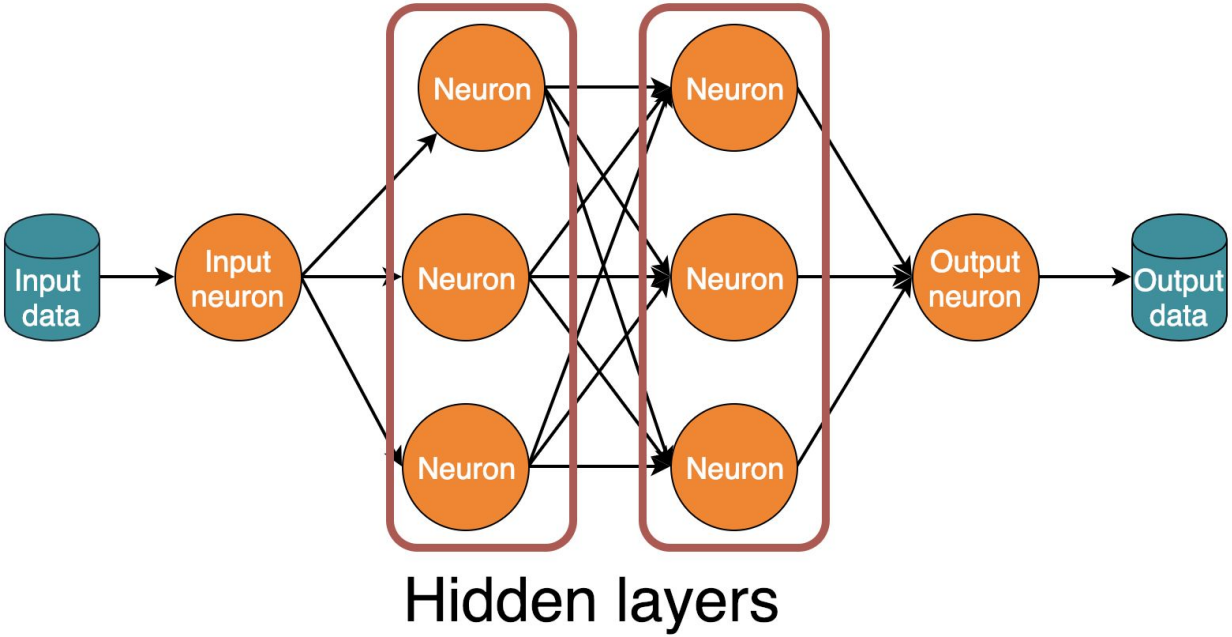
Back to the network



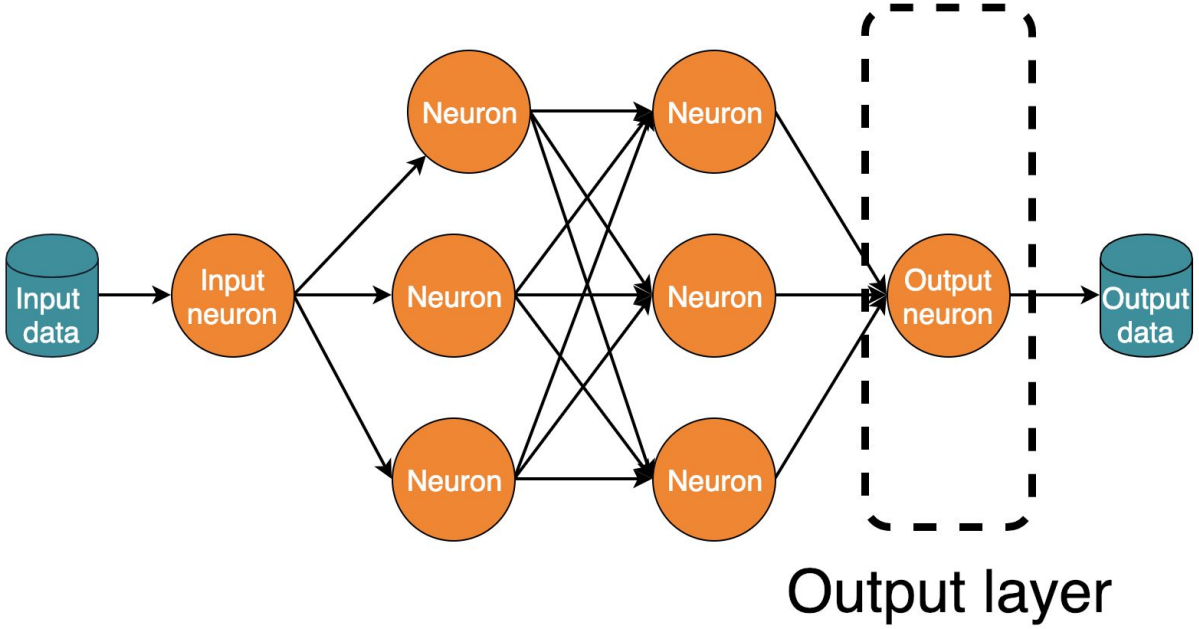
Input layer


Input data
Input neuron
Neuron
Neuron
Neuron
Neuron
Neuron
Neuron
Output neuron
Output data
Input layer

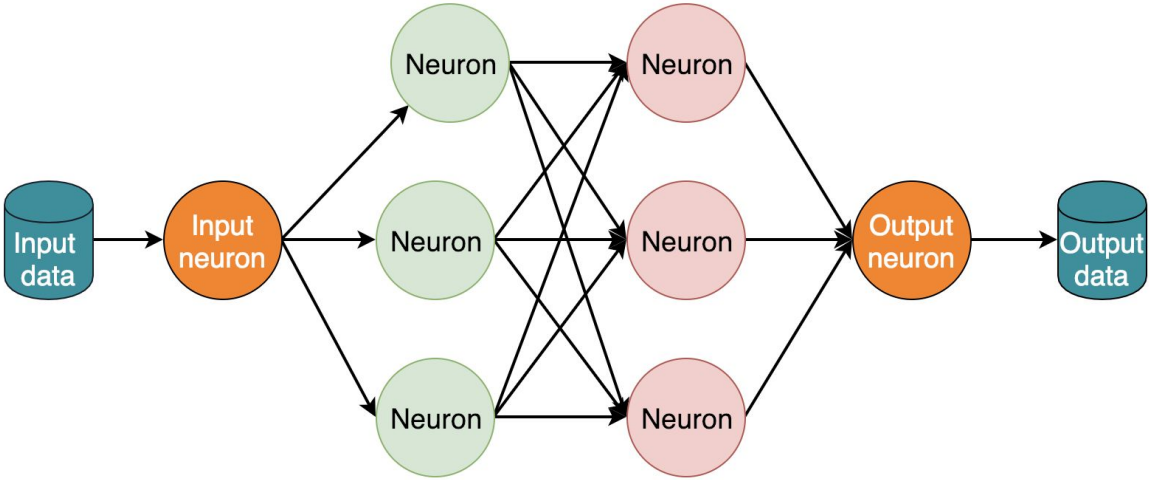
Hidden layer



Output layer



Output layer



Every layer can be of a different type

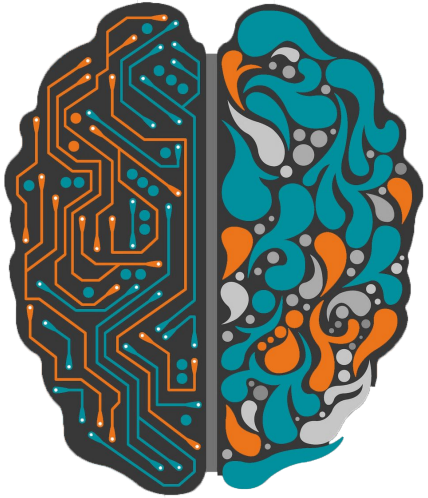
Example layers

Convolution layer for images

Long short-term memory

More layers at
https://keras.io/layers/

Windesheim Flevoland



Deep learning

Vincent van Dijk

9 januari 2018