Deep learning

Bilal Al Homsi

introduction

- Deep learning is a **supervised** learning process. •

Deep learning is able to automatically extract and learn filters from the data.

- With deep learning one can achieve the following: •

Classification

- **Localization**
- **Object**

detection • Object segmentation:

- **Semantic** segmentation •

Instance segmentation

Classification

• An artificial neural network can be trained to classify new data points. • There

are two types of classification:

- **Binary** classification
- **Multiclass** classification •

The training process is as follows:

- Training data is fed into the model. • The model attempts to learn features from the labeled data in order to correctly classify unknown data. • The model is evaluated and tested with new unknown data.

Localization

- An artificial neural network can be trained to
  Locate objects in an image.
- The coordinate points of a bounding box are used
  (Bounding Box) determined. This box frames the object you are looking for.

Object detection

- An artificial neural network can be trained to recognize objects in images.
- Both classification and localization of the Objects identified.
- Both the coordinate points and the class of the object are recognized and determined.

Object segmentation

- An artificial neural network can be trained to segment objects in images.
- A distinction is made between **semantic** and instance segmentation.
- Semantic segmentation means that all objects of a class are interpreted as a coherent segment, while instance segmentation means that the individual objects of a class are treated as instances of this class and are segmented separately accordingly.

Low level features, mid level features, High level features

- Low-level features:
  - Pixel values
  - Color information
  - Simple edges
- Mid-level features:
  - Textural features
  - Pattern
  - Simple shapes
- High level features:
  - Complex objects
  - Connections

- In image processing, **features** are *characteristics* that describe objects and their relationships in the image.
- These features will through *convolutional layers* extracted and learned.

Example

https://velog.io/@ktm1237/MIT-Introduction-to-Deep-Learning

Deep learning

- To train an **artificial neural network ,** do the following required:
  - A qualitative **data set**
    - It must be a large, diverse data set and with sufficient variance.
  - A suitable **network architecture**
    - It depends on the task, but the more complex the architecture, the better the training can be.
  - A **powerful computer:** • Training on the **CPU** slowly • Training on the **GPU** fast

record

- In order to achieve good **accuracy** with the neural networks, the data set must cover as many cases of the given problem as possible .
  - **Extensive:**
    - The data set must be relatively large, i.e. the more data it contains, the better the model can learn.
  - **Diverse**
    - The data set must cover as many cases as possible about the problem.
  - **With sufficient variance**
    - Variance means that the same data contains different types.
  - **Balanced distribution of classes in the data set:**
    - The classes have the same distribution, i.e. each class has the same number data points.

record

- Data is one of the most important resources and is needed to to train neural networks for specific tasks.
- It is said, *'As your data is, so is your model'* • And

regarding the question, how can you get data? • Data depends heavily on the task, for example:

  - Self-collected and pre-processed. • Bought.
  - Downloaded from free internet sources. • Etc.

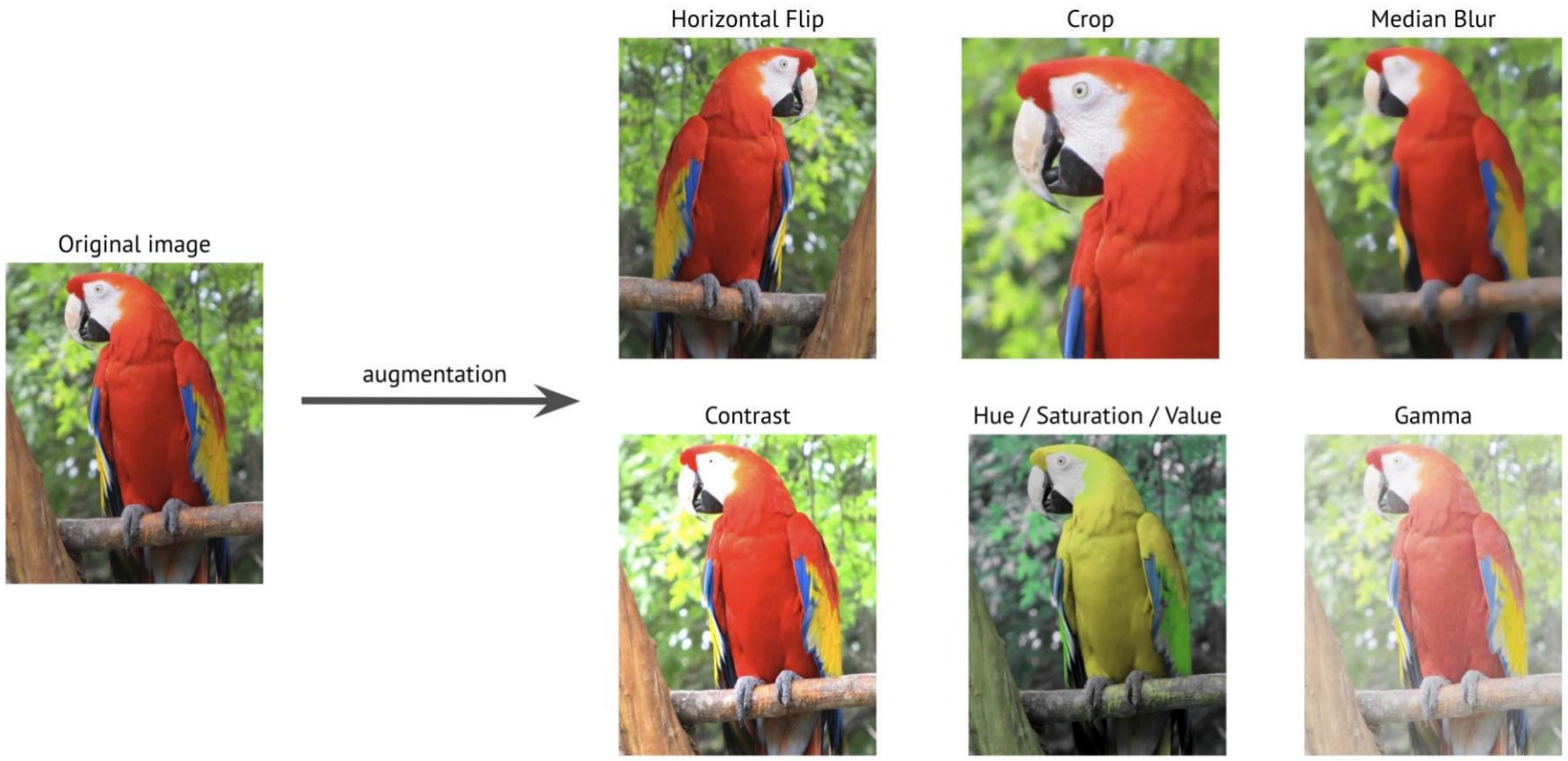
augmentation

- Is a technique used in neural networks to create artificial variations of training data. • So examples are random: • Rotate
  - Cropping
  - Mirroring •

  Zooming (in and out) • Changing

  brightness and contrast
- This technique is used to make the data set contain more images and Variance can be obtained.

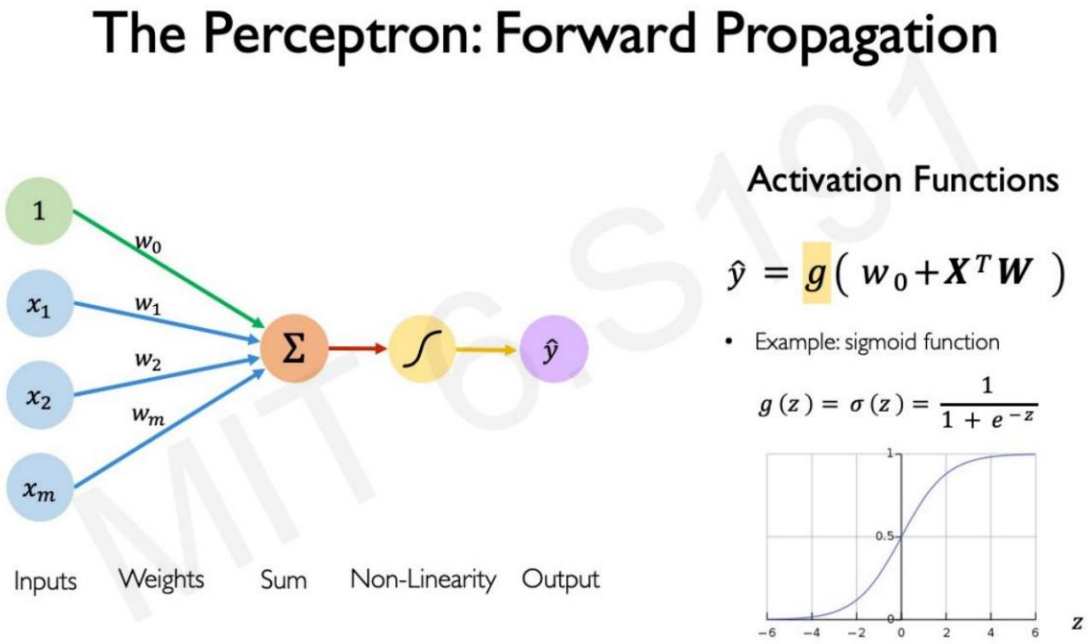
Example of augmented image data



https://albumentations.ai/docs/introduction/image_augmentation/

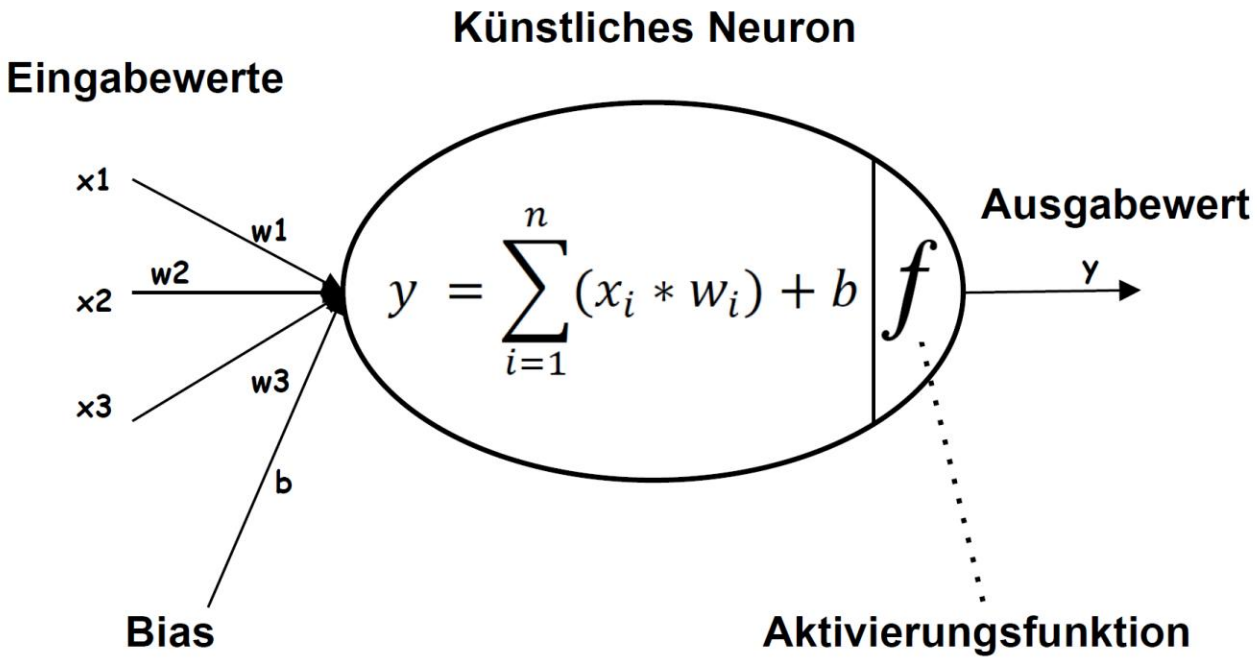
The perceptron

- The input values are initialized with the randomly weights. Then the sum is formed.
- 1*w0 +x1*w1 + x2*w2 + x3*w3
- A so-called activation function is then switched on. This allows some neurons to be activated or not.



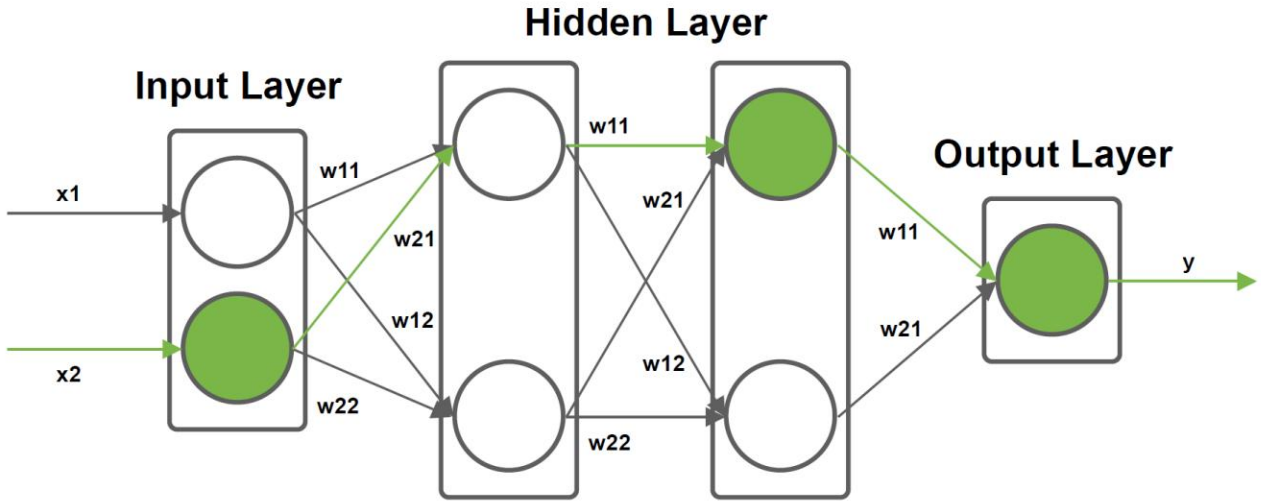
http://introtodeeplearning.com/slides/6S191_MIT_DeepLearning_L1.pdf

The perceptron



Simple artificial neural network

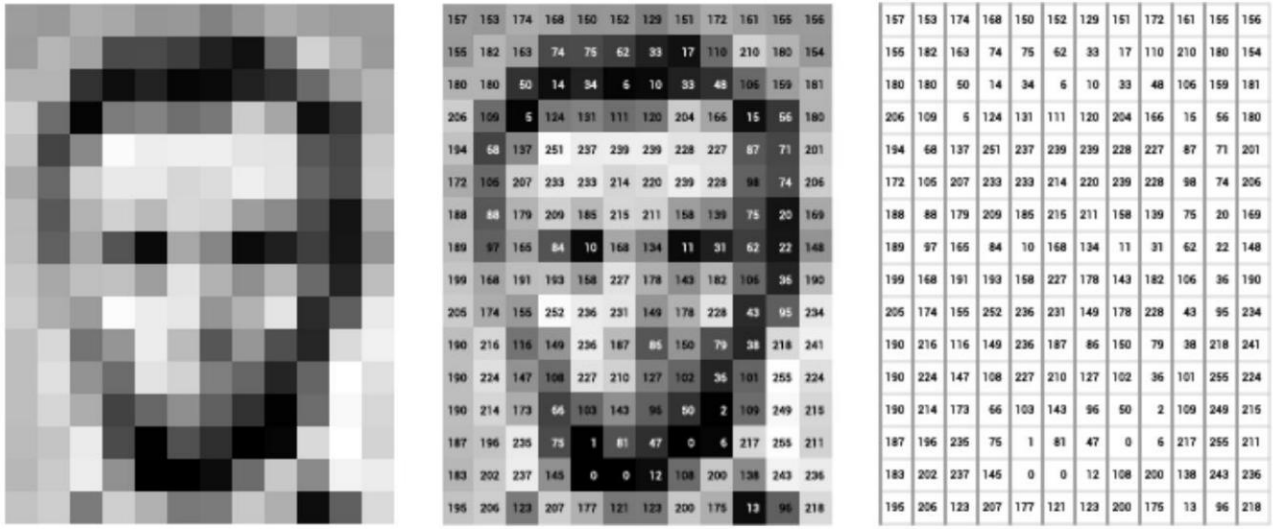
- Following neural network consists of an *input layer* **(Input Layer),** a *hidden layer* **(Hidden Layer)** and an output *layer* **(Output Layer).**
- The hidden layers can be any number. This defines the **depth** of the neural network.
- Each of these layers can contain any number of neurons. This results in a certain **complexity** of the neural network.



Input layer

- The input layer contains as many neurons as the input values. This means, for example,

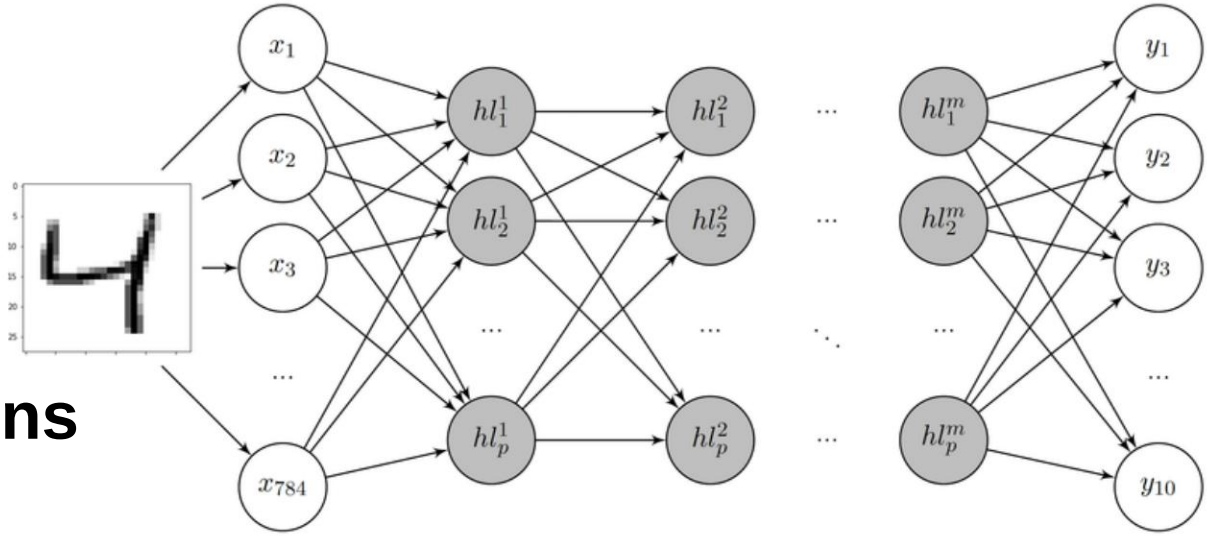
  if the image size is **28*28** pixels, then the input layer must have **784 neurons ,** with each neuron taking a pixel value.



https://ai.stanford.edu/~syyeung/cvweb/tutorial1.html

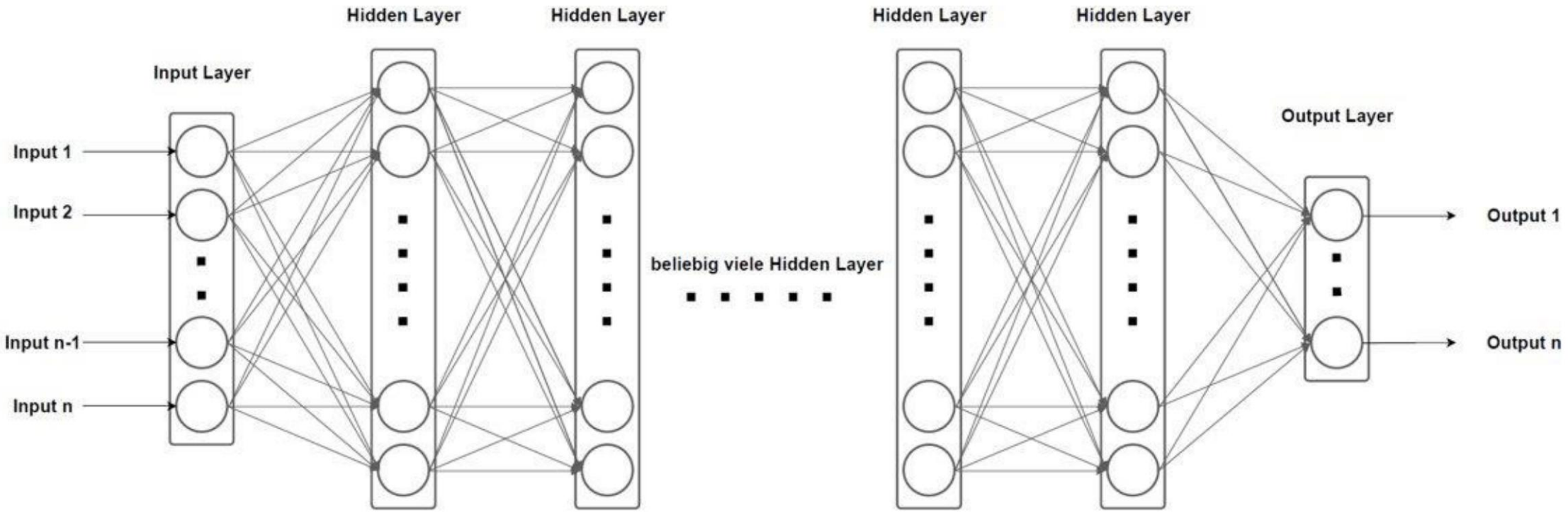
Output layer

- The output layer contains as many neurons as the number of target classes. That means if there are **10 classes** with the aim of classifying the new image into one of these classes , then the output layer must have **10 neurons** so that these 10 classes can be represented.

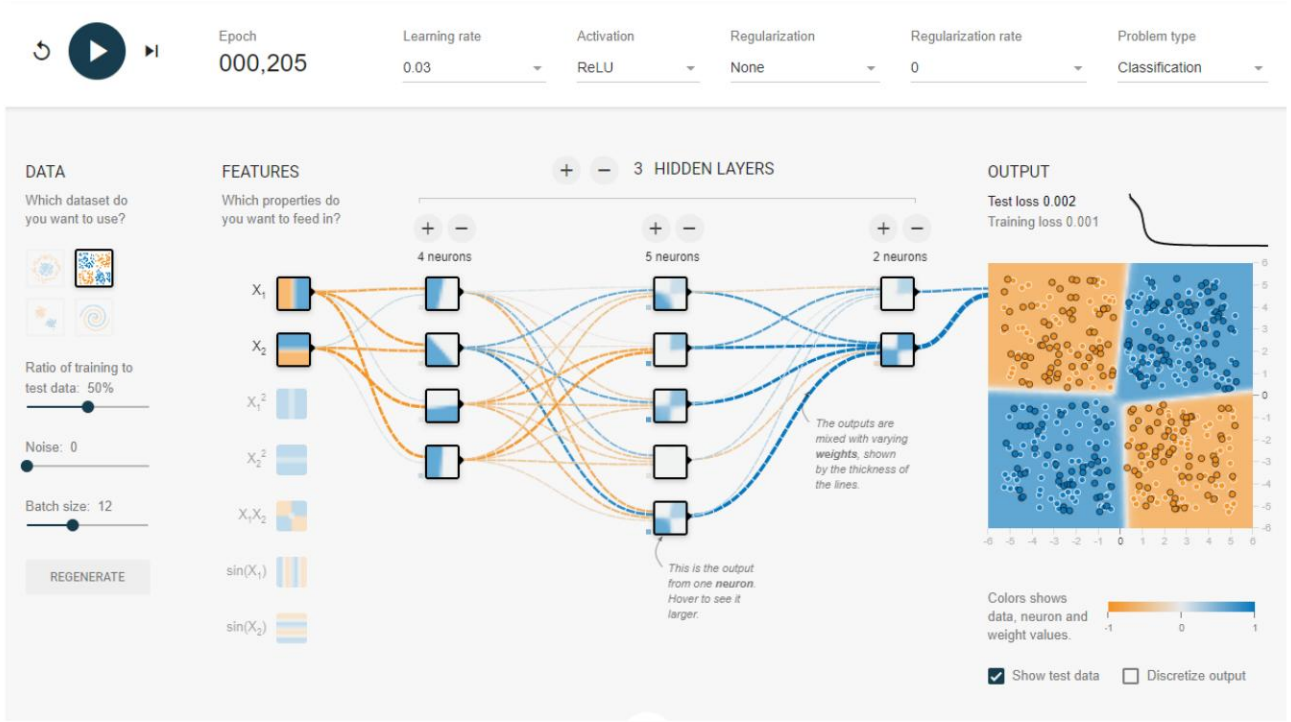


https://www.researchgate.net/figure/Artificial-neural-network-architecture-for-the-MNIST-dataset-classification-The-input_fig4_349991068

A deep neural network



Playground Tensorflow



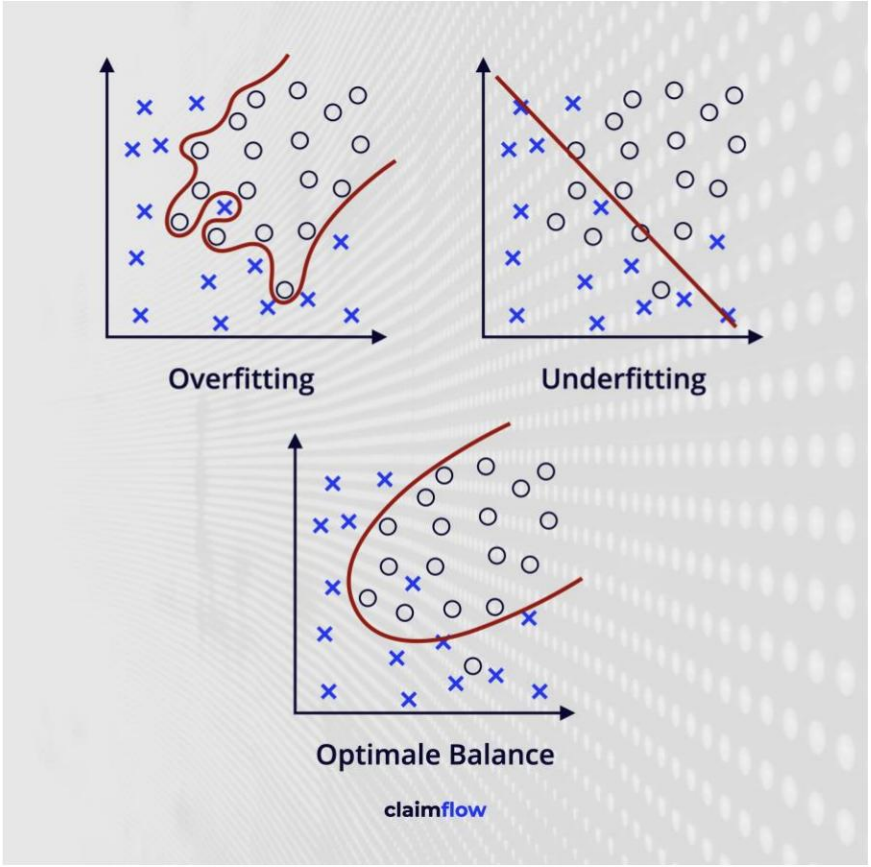
https://playground.tensorflow.org/

Overfitting, underfitting and optimal balance

- **Overfitting:**

It's about the neural network adapting too much to the data.

The model achieves very good results on training data, but not on test data. The reason is that the model has memorized the features of the data. So it cannot generalize.

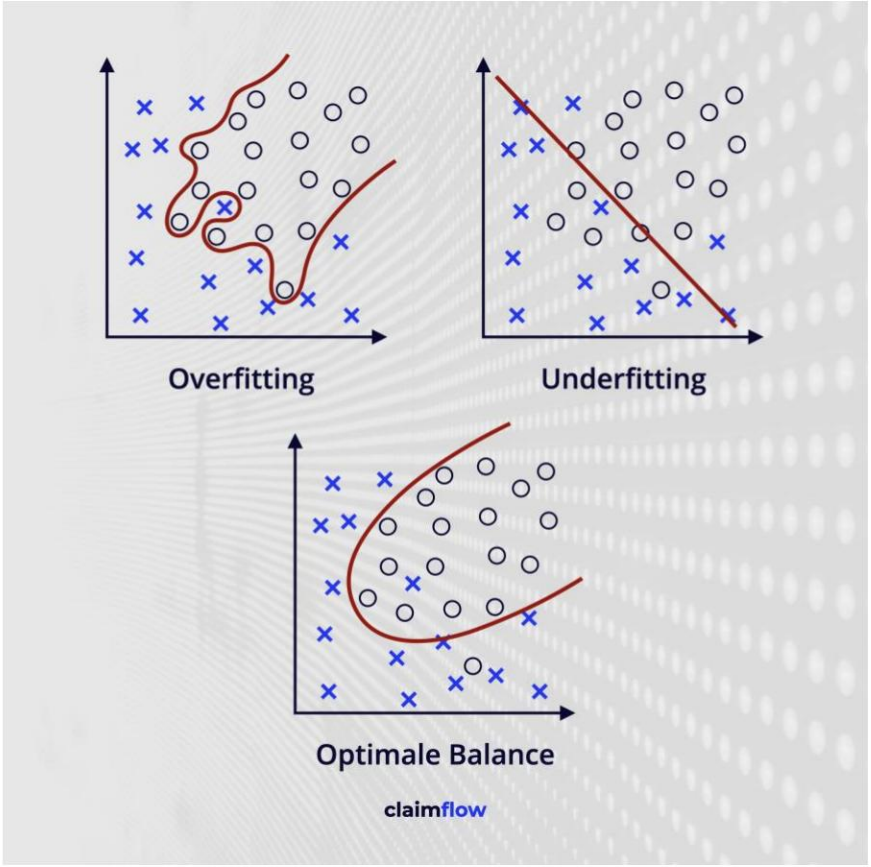


https://www.claimflow.de/post/overfitting-und-underfitting

Overfitting, underfitting and optimal balance

- **Underfitting:**

The point is that the neural network has not learned any features from the data. The model achieves very poor results on training data, but also on test data. The reason is that the model has not understood and learned the features of the data. So it cannot generalize.

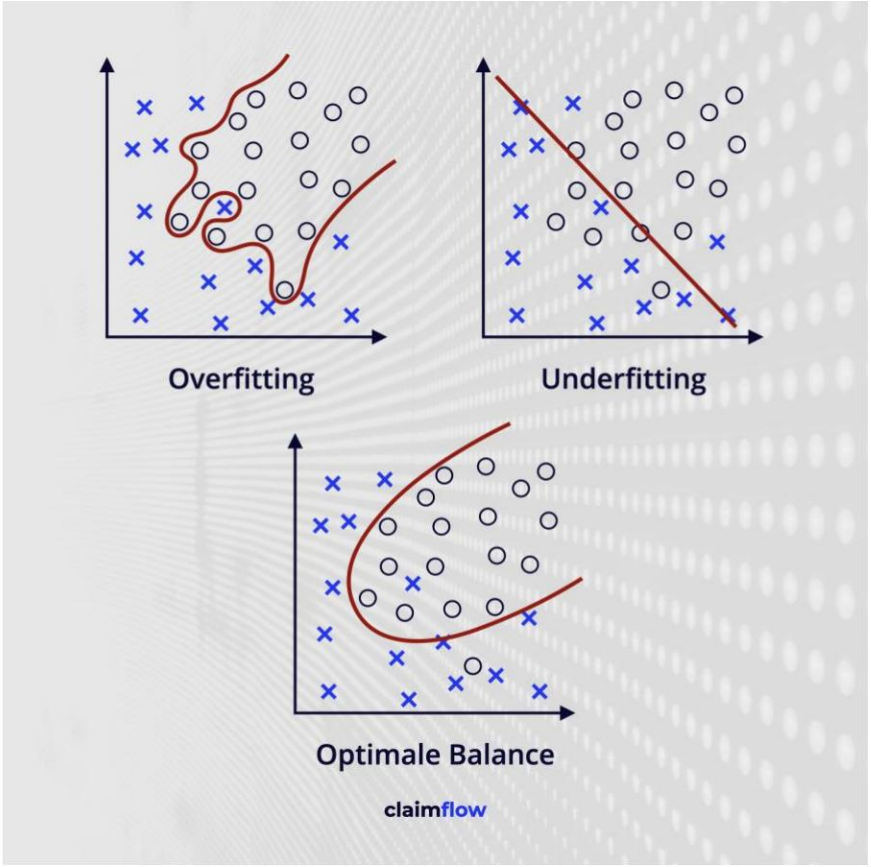


https://www.claimflow.de/post/overfitting-und-underfitting

Overfitting, underfitting and optimal balance

- **Optimal balance:**

The point is that the neural network has learned enough features from the data. The model achieves very good results on training data, but also on test data. The reason is that the model has understood and learned the features of the data . So it can generalize well.

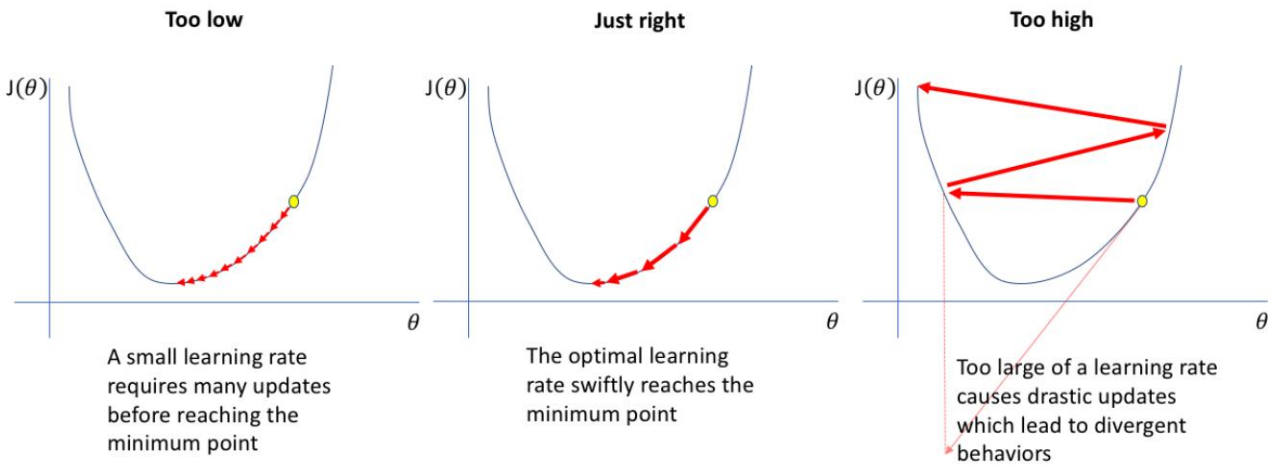


https://www.claimflow.de/post/overfitting-und-underfitting

Learning rate

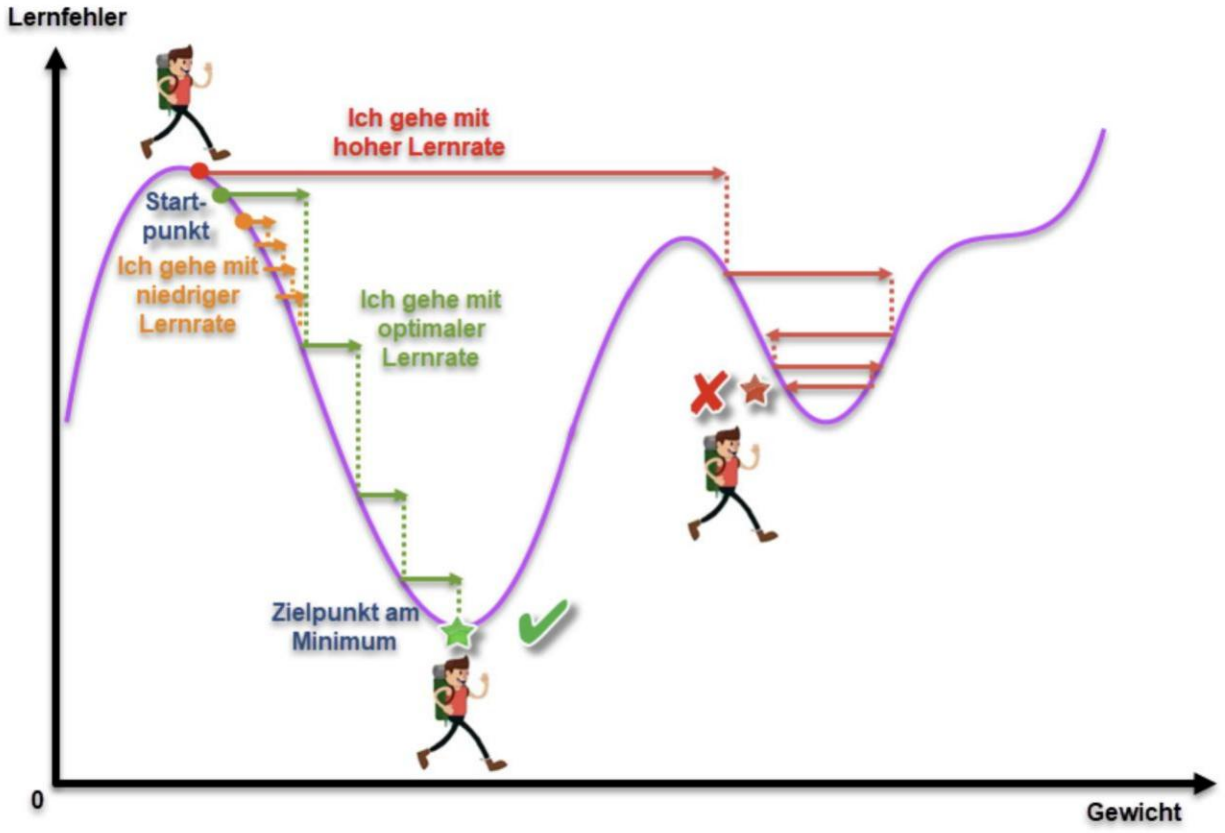
- Learning rate is one of the most important hyperparameters used in optimization.

- The learning rate indicates how large the steps are as the model updates the weights.

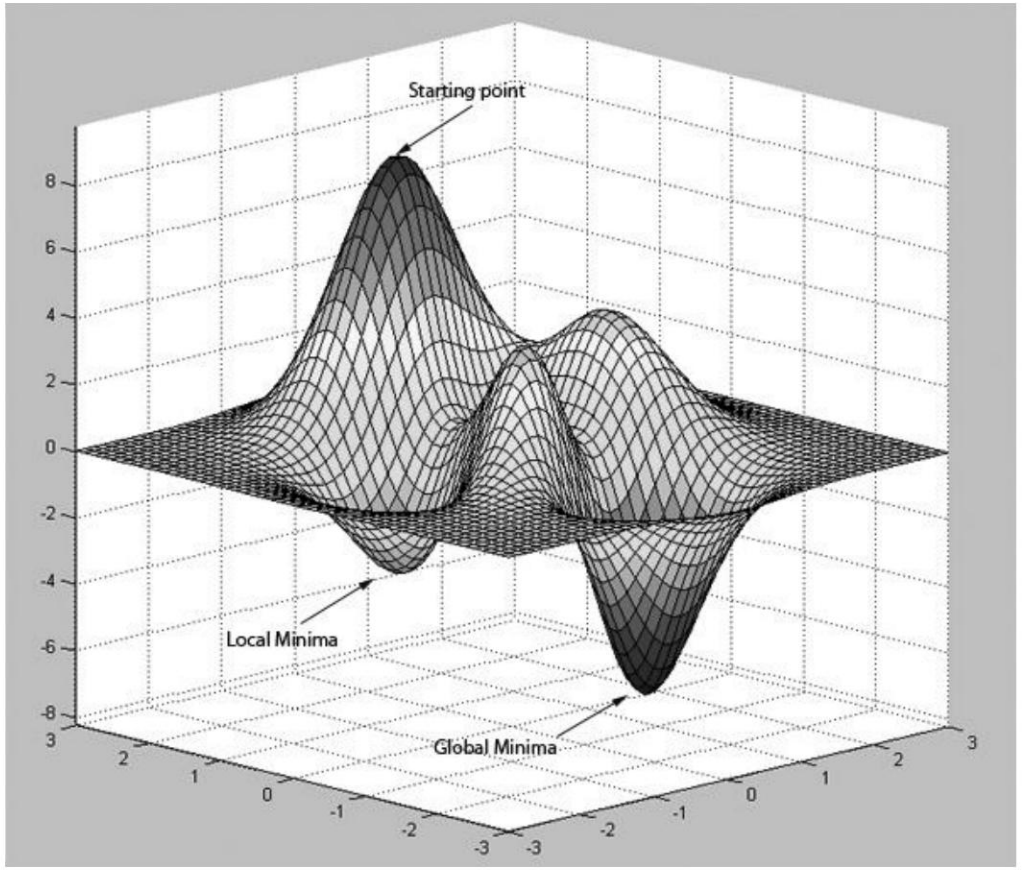


https://www.jeremyjordan.me/nn-learning-rate/

Example of local and global maxima and Minima



https://www.ngw.ch/wp-content/uploads/2019/01/Kinderuniversit%C3%A4t_Reichenbacher_09012019.pdf



https://www.codeplanet.eu/tutorials/csharp/70-kuenstliche-neuronale-netze-in-csharp.html _

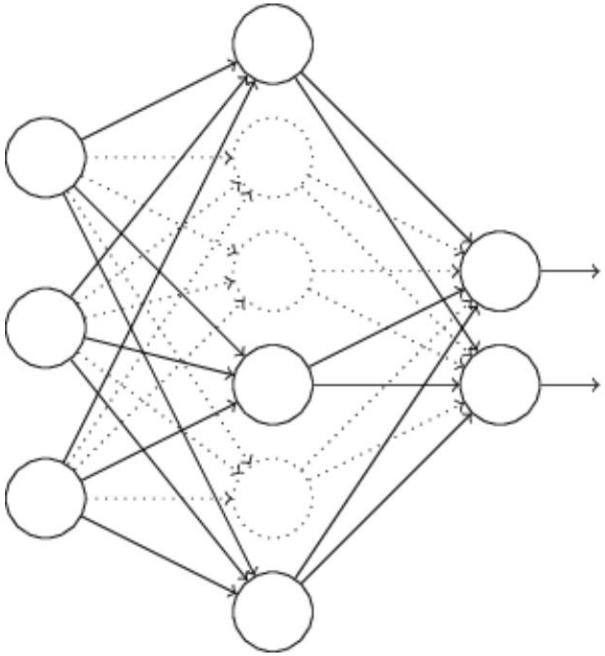
Regularization

- It is about improving the generalization of the model to unseen data.

- There are two options, among others:
  - **Dropout:** Some connections to neurons are intentionally eliminated.
  - **Early Stopping:** An end is set at a specific point so that **overfitting** and **underfitting** can be avoided.

Regularization

Dropout



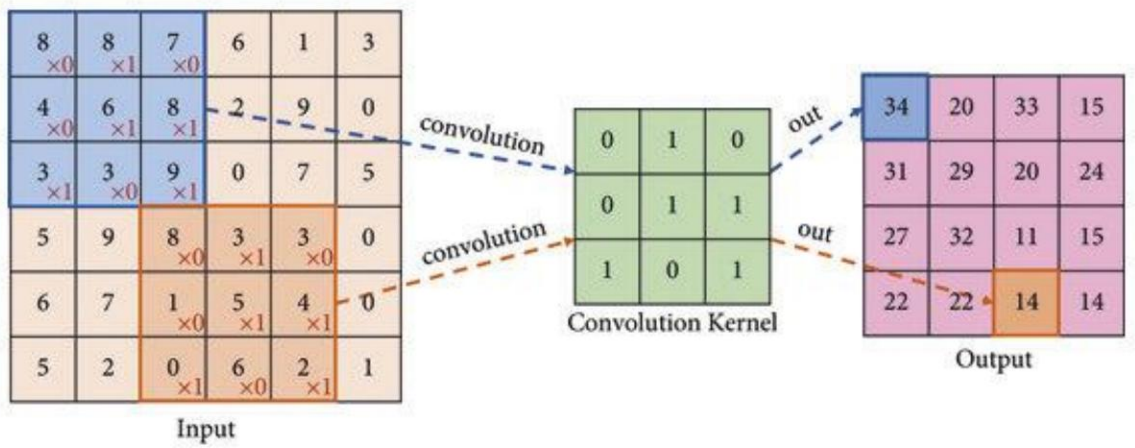
https://kharshit.github.io/blog/2018/05/04/dropout-prevent-overfitting

Early stopping

http://introtodeeplearning.com/slides/6S191_MIT_DeepLearning_L1.pdf

Convolutional operators

- This is a special type of neural network used for *image data .*
  A **filter (kernel)** is used on the image.
- This is moved step by step from left to right, the input values are multiplied by the kernel values and summed up. The result of this operation is gradually entered into a **feature map .**



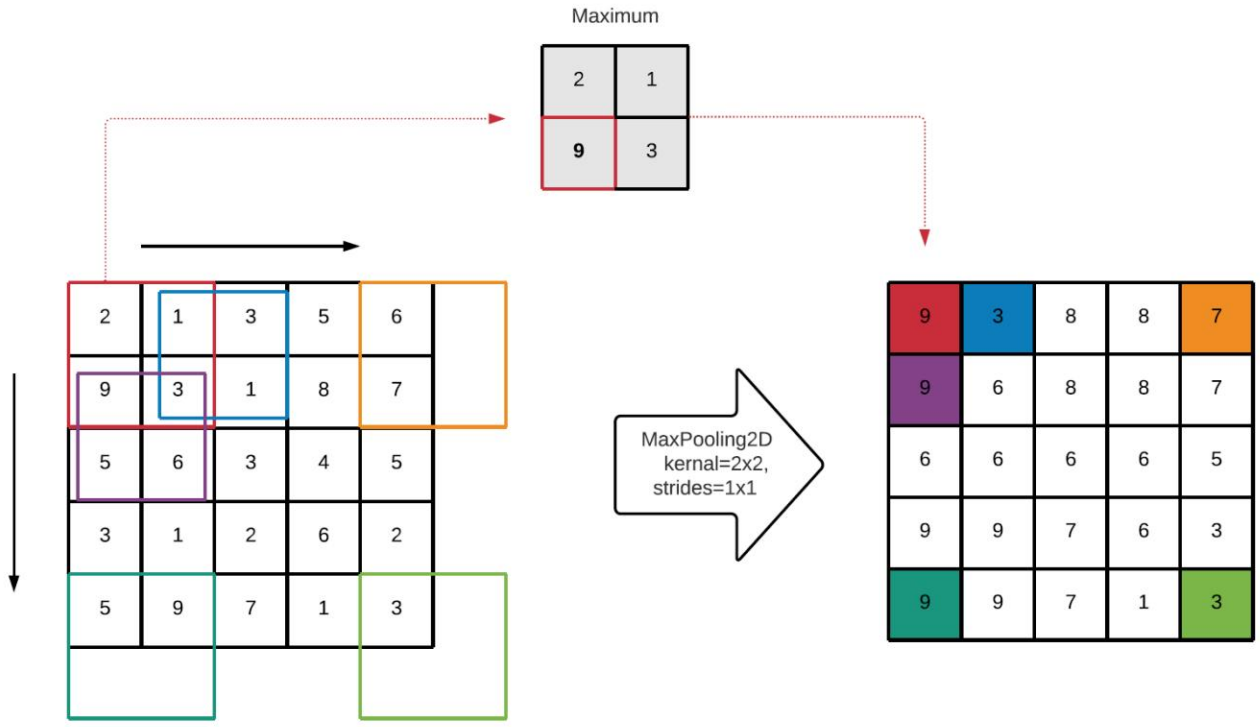
https://www.researchgate.net/figure/Convolution-with-convolution-kernel-of-size-33-stride-of-1-and-no-zero-padding_fig2_356590429 _

Convolutional layers

- In the convolutional layer the following can be determined:
  - The number of filters
  - The size of the kernel
  - The step size *(stride)*
- The following can be achieved with the convolutional layer: • Extraction of features and useful features
  - Reducing the dimension of the image data (more memory and less computing power)

Pooling operator

- The pooling operation is directly after the convolutional operation used.
- The main idea is that reduce the dimension of the *feature map* while retaining the relevant information .

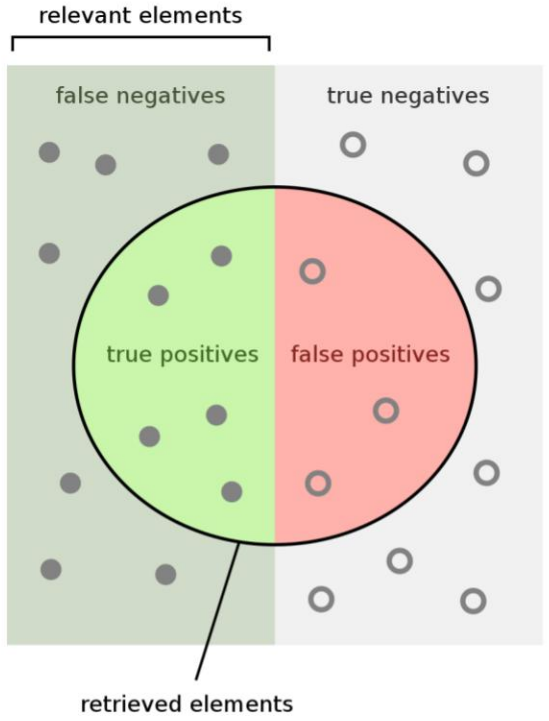


Pooling layers

• The following can be determined in the pooling layer:
• The size of the
kernel • The step size
(stride) • Bounding layer
(padding) • The padding layer helps maintain the dimension of the feature map. This means that the size of the input and output is the same. • The
following can be achieved with the pooling layer: •
MaxPooling2D
• The maximum number in the pooling
kernel is selected.
• **AveragePooling2D** • The average number in the pooling kernel is chosen.

Recall and precision

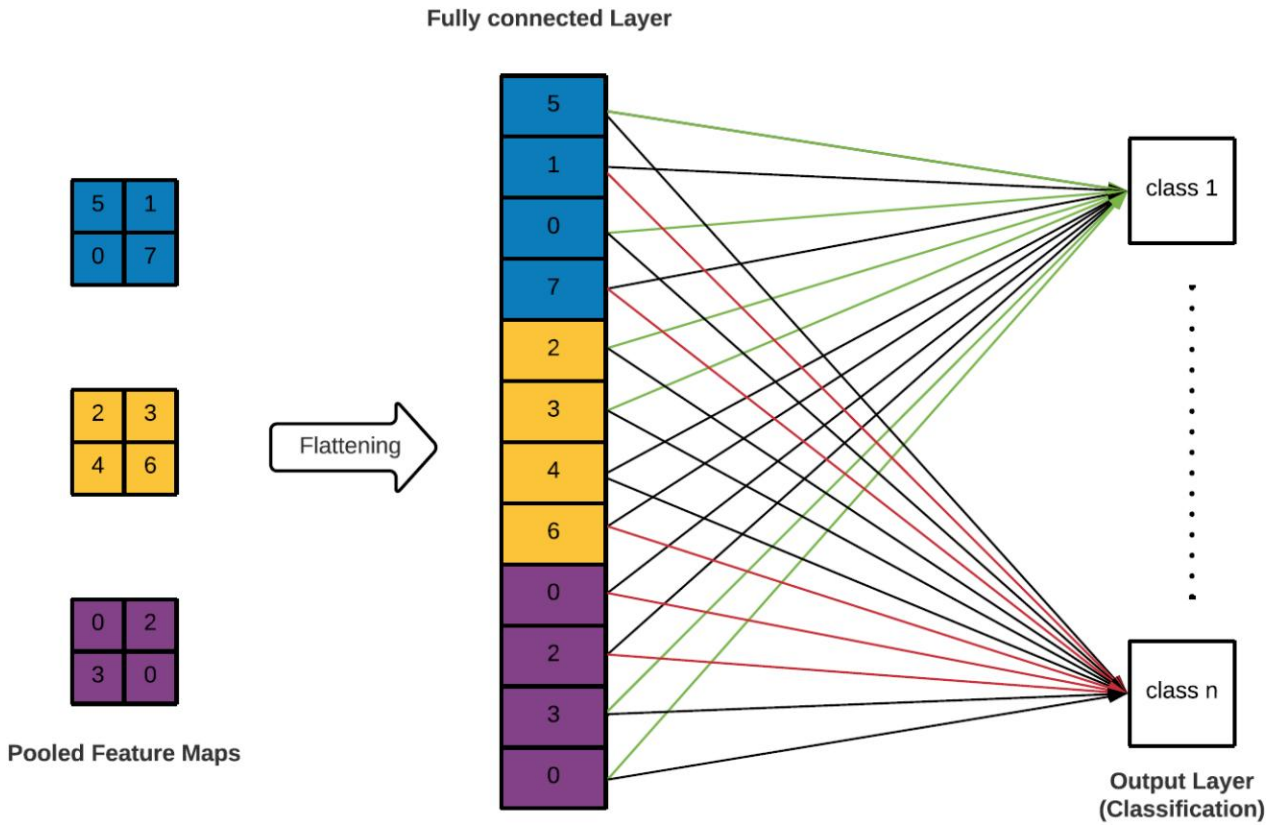
- These are metrics for evaluating the neural networks.
- **Recall:**
  How many relevant data points were found.
- **Precision:**
  How many data points found are relevant or correct.



https://en.wikipedia.org/wiki/Precision_and_recall

Flattening

- *Feature maps* from the final *convolutional layer* are smoothed so that they are fed into the fully connected neural network .
- Each of these values is the input to the input neurons.



Project

- A project is provided in this module. This will be counted as part of the examination performance.
- The project requires:
  - A data set for at least two classes.
    - If necessary, the image data should be preprocessed.
  - A suitable architecture for a convolutional neural network.
    - It is necessary to have different constellations and combinations of hyperparameters to try out.
  - Any result of the accuracy of the model should be included the respective combination of parameters can be saved.
  - A good result is expected for the given classification problem.

Google Colab

- This is like a **Jupyter Notebook** running cloud resources can be carried out.
- The code can run on **CPU, GPU or TPU .**
- This can speed up computing performance.
- The code can be connected to **Google Drive** to the *RAM* and to conserve *storage capacity .* The data can therefore be fetched directly via the cloud.
- For more information see: https://colab.google/