Cagri Ozcinar, PhD

LTTI.00.032 - Machine Learning in Synthetic Biology

Content

- **DL**
- **Multilayer perception**
- **Convolutional neural networks (CNNs)**
- **Recurrent neural networks (RNNs)**
- **Generative adversarial networks (GANs)**

What should you expect?

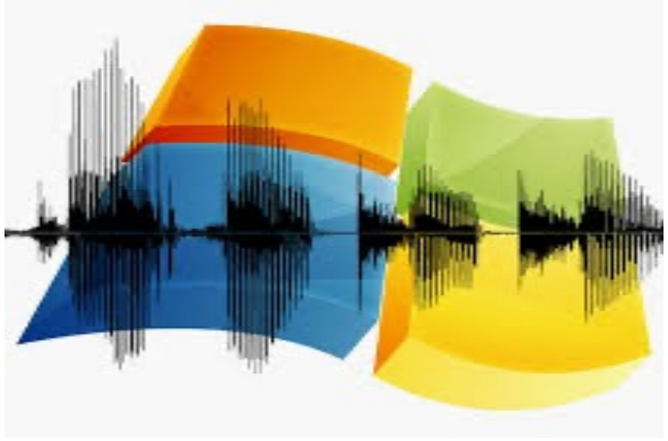
- **Theory**
- **Tutorials**
- **Applications**

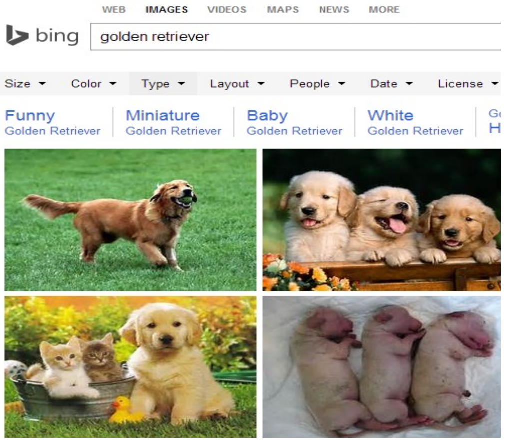
Cagri Ozcinar, PhD

Deep Learning

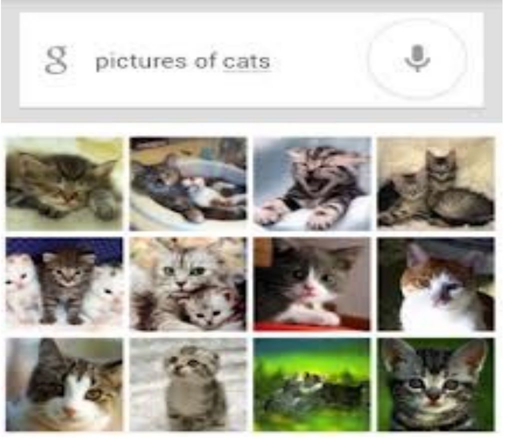
Cagri Ozcinar, PhD

Deep Learning – from Research to Technology





Translate Google



Deep Learning - breakthrough in visual and speech recognition

Cagri Ozcinar, PhD

Classical Computer Vision Pipeline

Cagri Ozcinar, PhD

Classical Computer Vision Pipeline

CV experts

1. Select / develop features: SURF, HoG, SIFT, RIFT, …
2. Add on top of this Machine Learning for multi-class recognition (SVM, DT, etc.) and train classifier

Feature Extraction: SIFT, HoG...

Detection, Classification Recognition

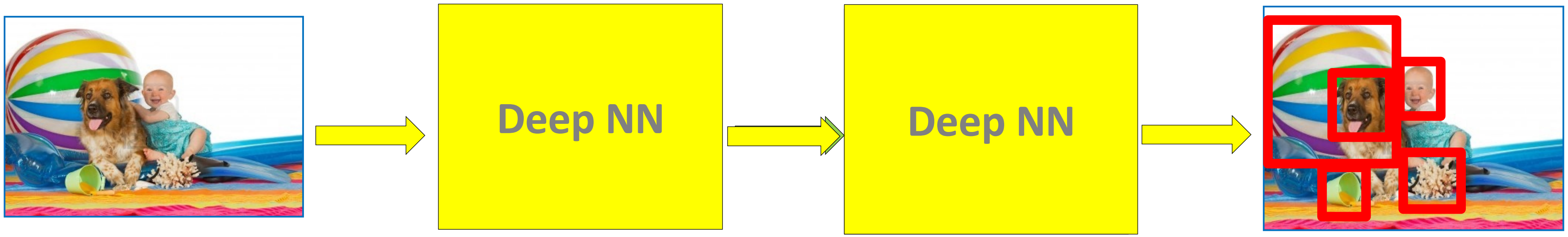
Classical CV feature definition is domain-specific and time-consuming

Deep Learning –based Vision Pipeline

Deep Learning:

- Build features automatically based on training data
- Combine feature extraction and classification

DL experts: define NN topology and train NN



Deep Learning promise:
train good feature automatically,
same method for different domain

Cagri Ozcinar, PhD

Computer Vision +Deep Learning + Machine Learning

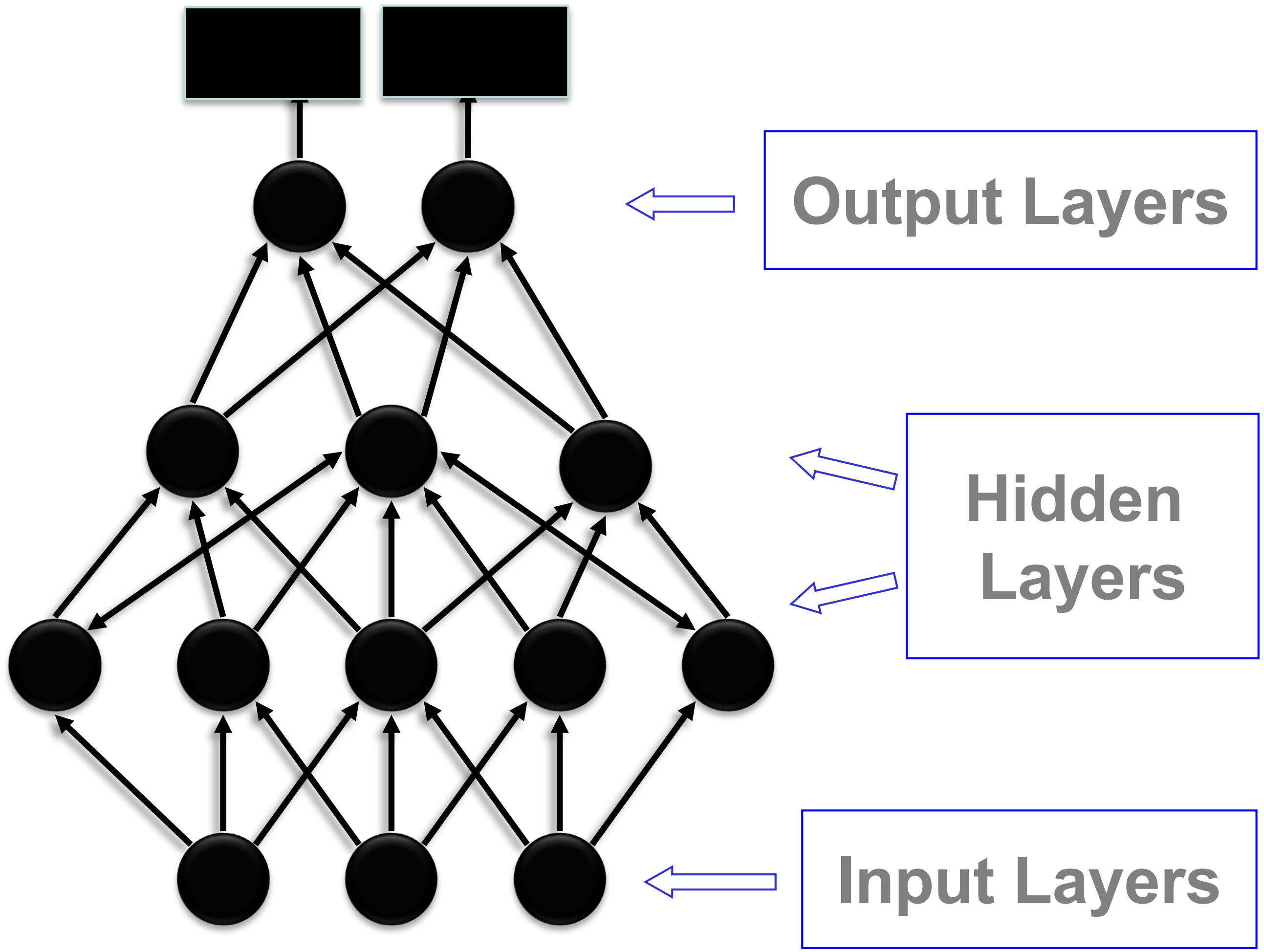
- Combine pre-defined features with learned features;
- Use best ML methods for multi-class recognition

CV+DL+ML experts needed to build the best-in-class

Combine best of Computer Vision
Deep Learning and Machine Learning

Deep Learning Basics

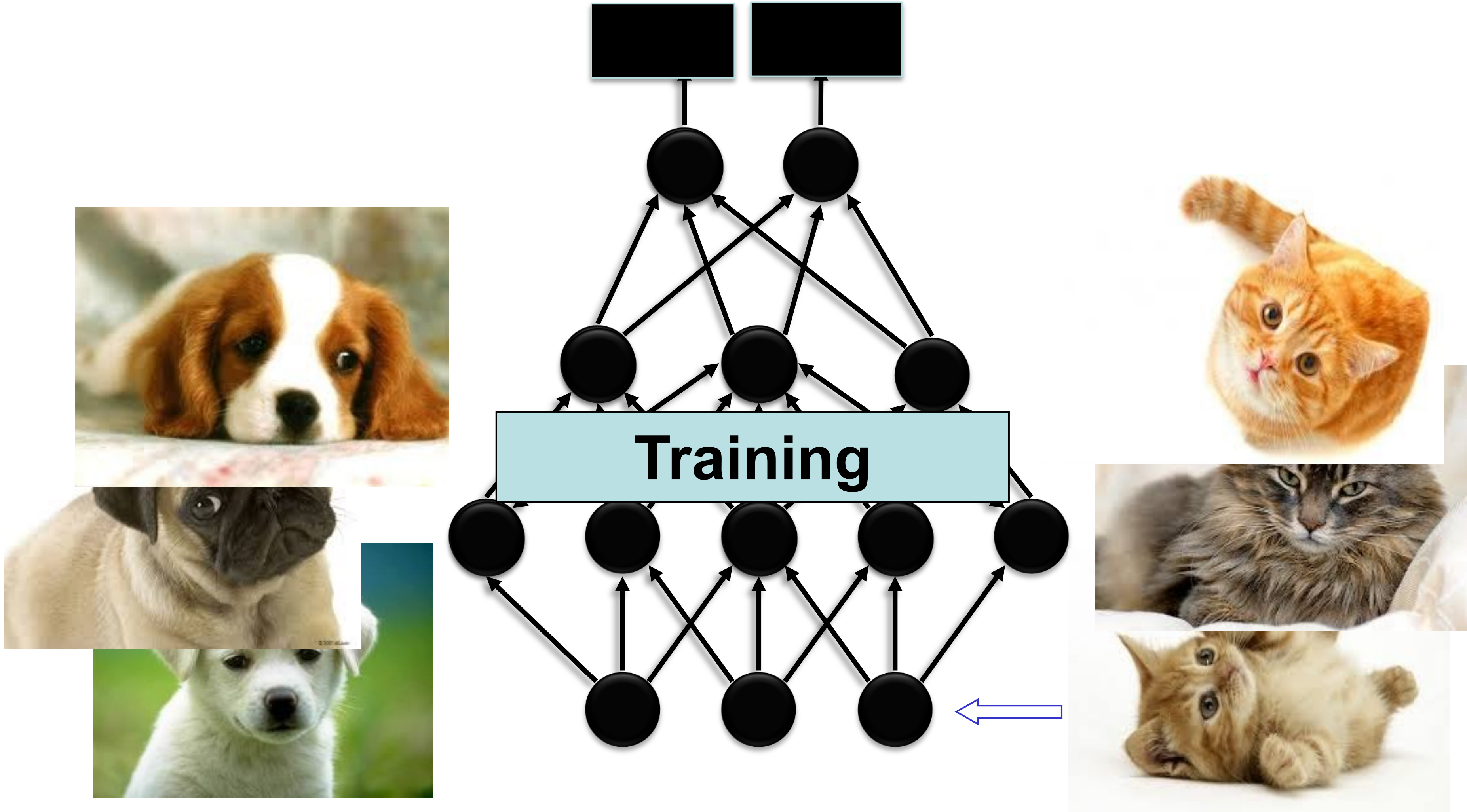
Deep Learning – is a set of machine learning algorithms based on multi-layer networks



Cagri Ozcinar, PhD

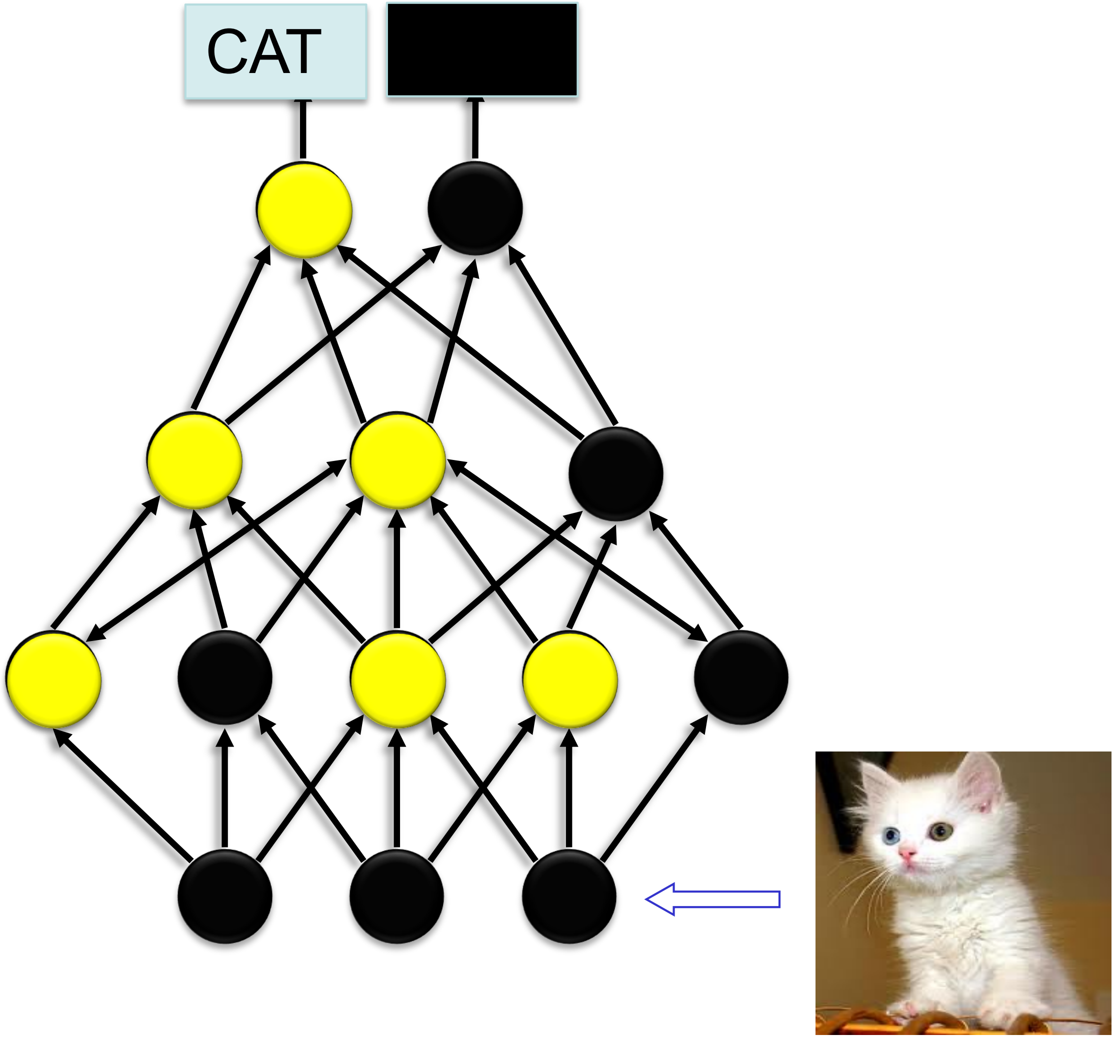
Deep Learning Basics

Deep Learning – is a set of machine learning algorithms based on multi-layer networks



Deep Learning Basics

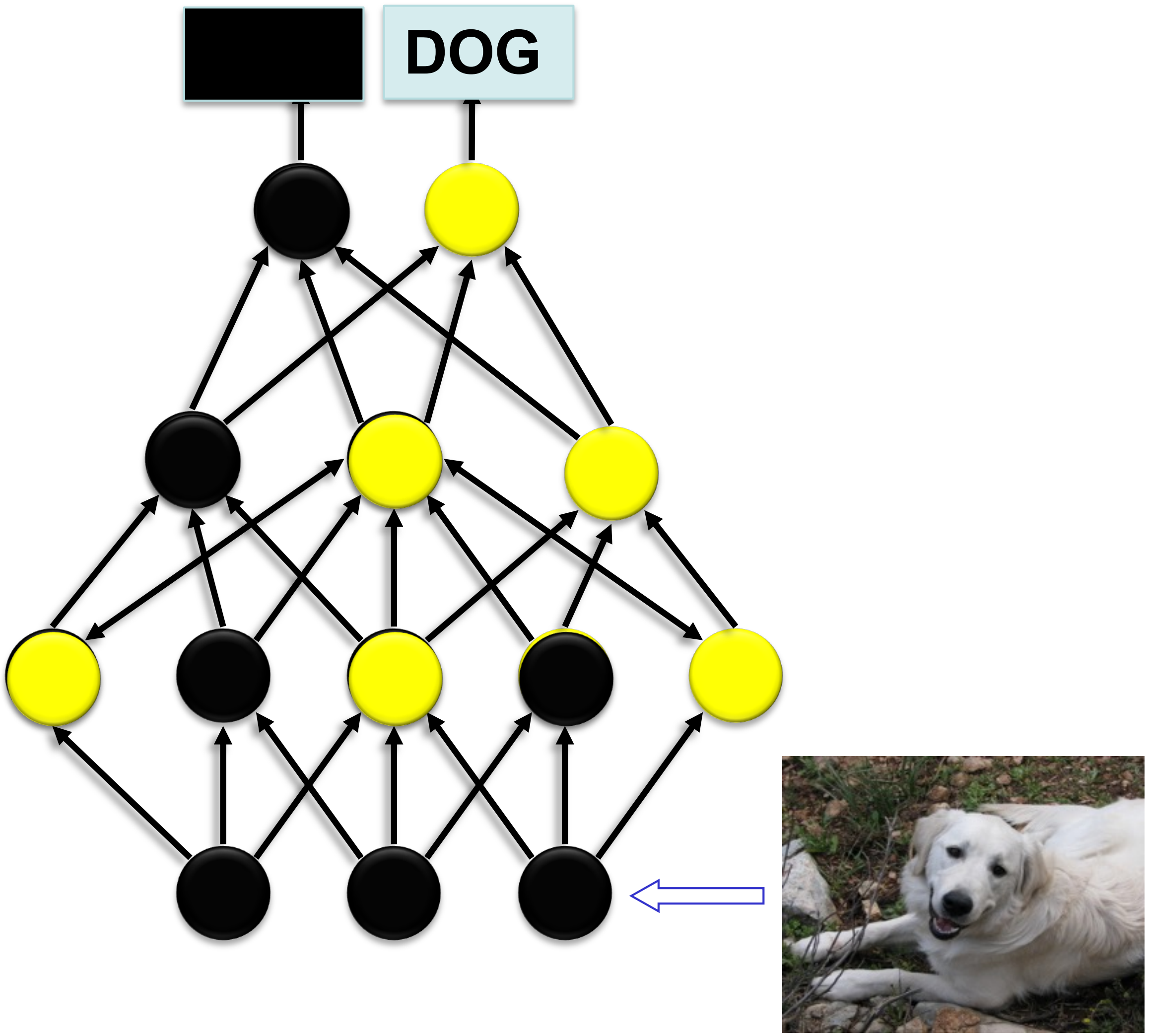
Deep Learning – is a set of machine learning algorithms based on multi-layer networks



Cagri Ozcinar, PhD

Deep Learning Basics

Deep Learning – is a set of machine learning algorithms based on multi-layer networks



Deep Learning Basics

Steps are as follows:

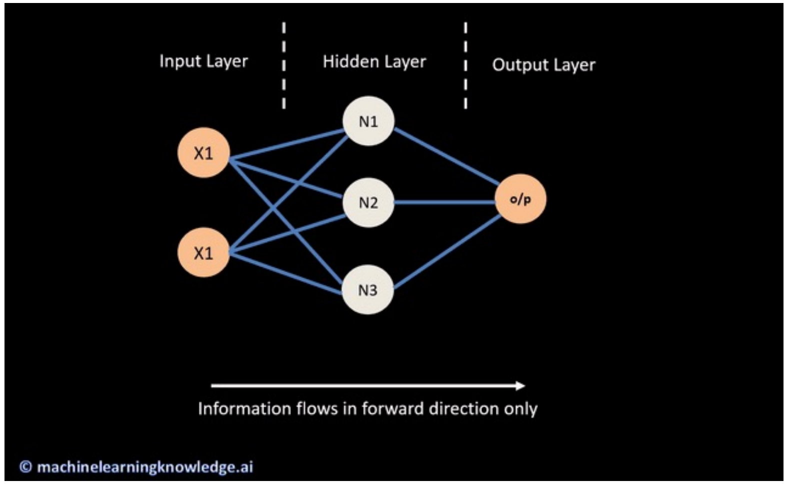
1. Sample a batch of data
2. Forward propagate it through the graph
3. Backpropagate to calculate the gradients
4. Update the parameters using gradients

Cagri Ozcinar, PhD

Deep Learning Basics

Steps are as follows:

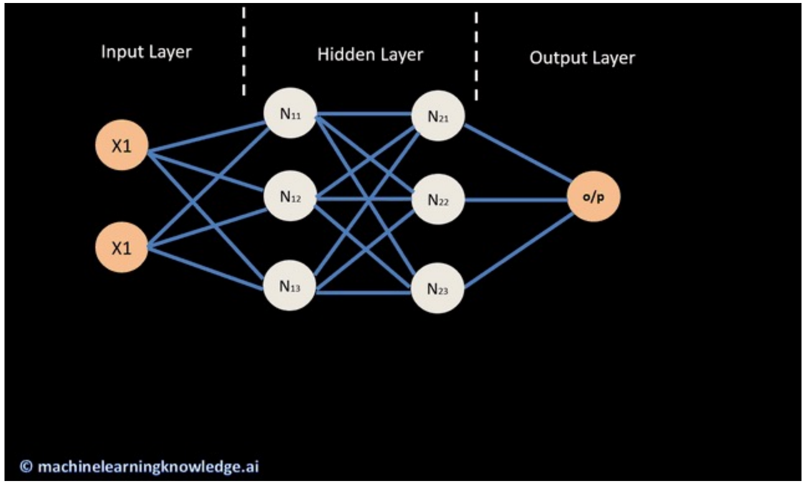
1. Sample a batch of data
2. Forward propagate it through the graph
3. Backpropagate to calculate the gradients
4. Update the parameters using gradients



Deep Learning Basics

Steps are as follows:

1. Sample a batch of data
2. Forward propagate it through the graph
3. Backpropagate to calculate the gradients
4. Update the parameters using gradients



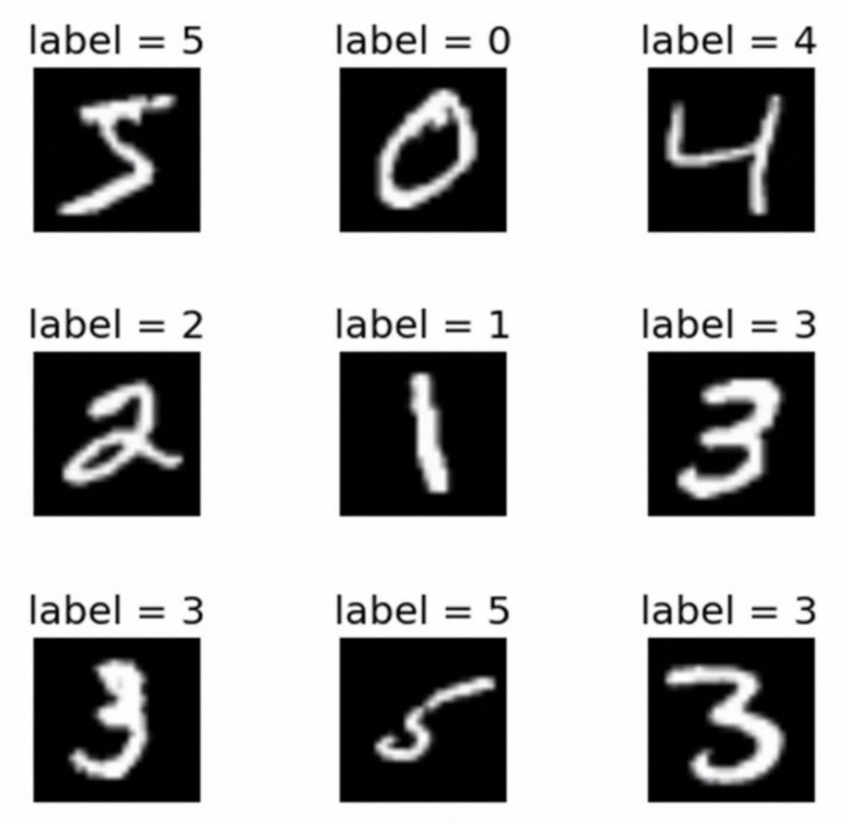
Cagri Ozcinar, PhD

Practical example: MNIST

- MNIST dataset
  - Handwritten digits
  - 70,000 28x28 b/w images
  - Represent the digits from zero through nine

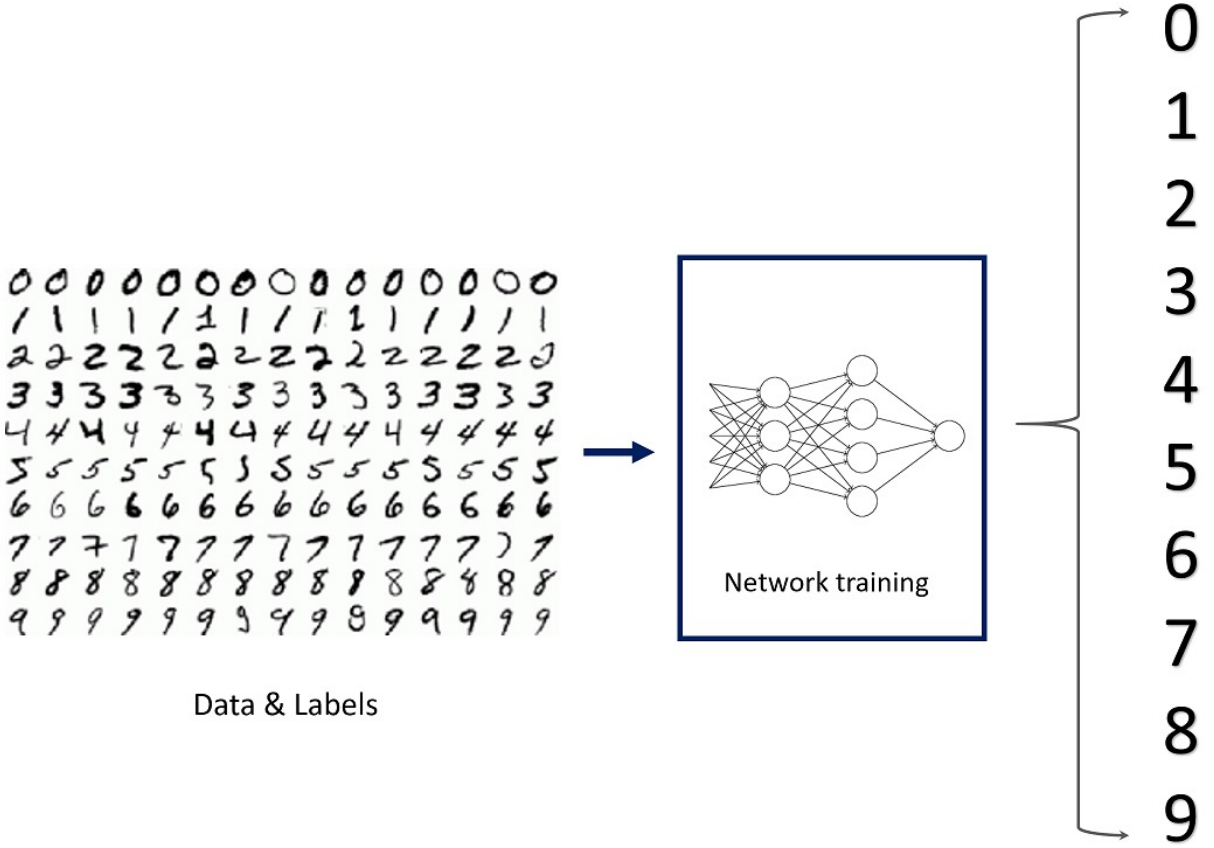
Practical example: MNIST

- Each letter can be written with different style



Practical example: MNIST

- Deep neural networks allow us to train a computer effectively
- Classification to handwritten digits



Cagri Ozcinar, PhD

Practical example: MNIST

- **SoftMax activation** function can be used for multi-class data classification
- Sigmoid activation function can be used for binary classification

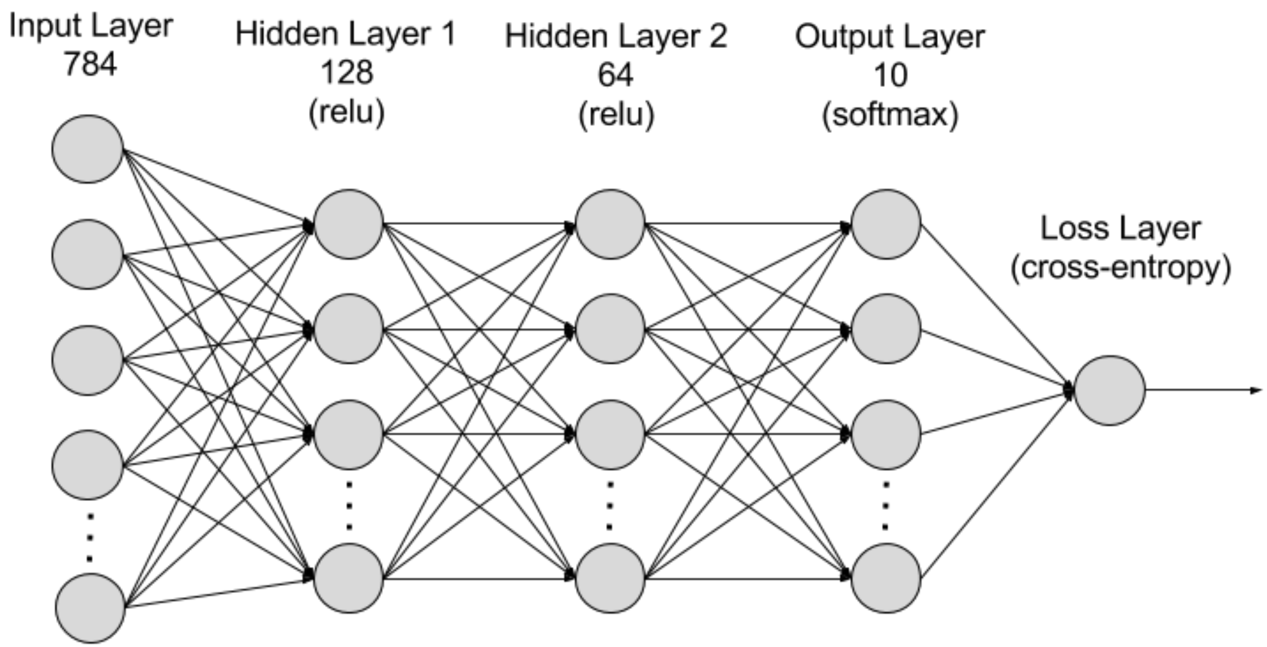
Multi-class

Practical example: MNIST

- Multi-layer perception: we need to flatten a given 28x29 images into

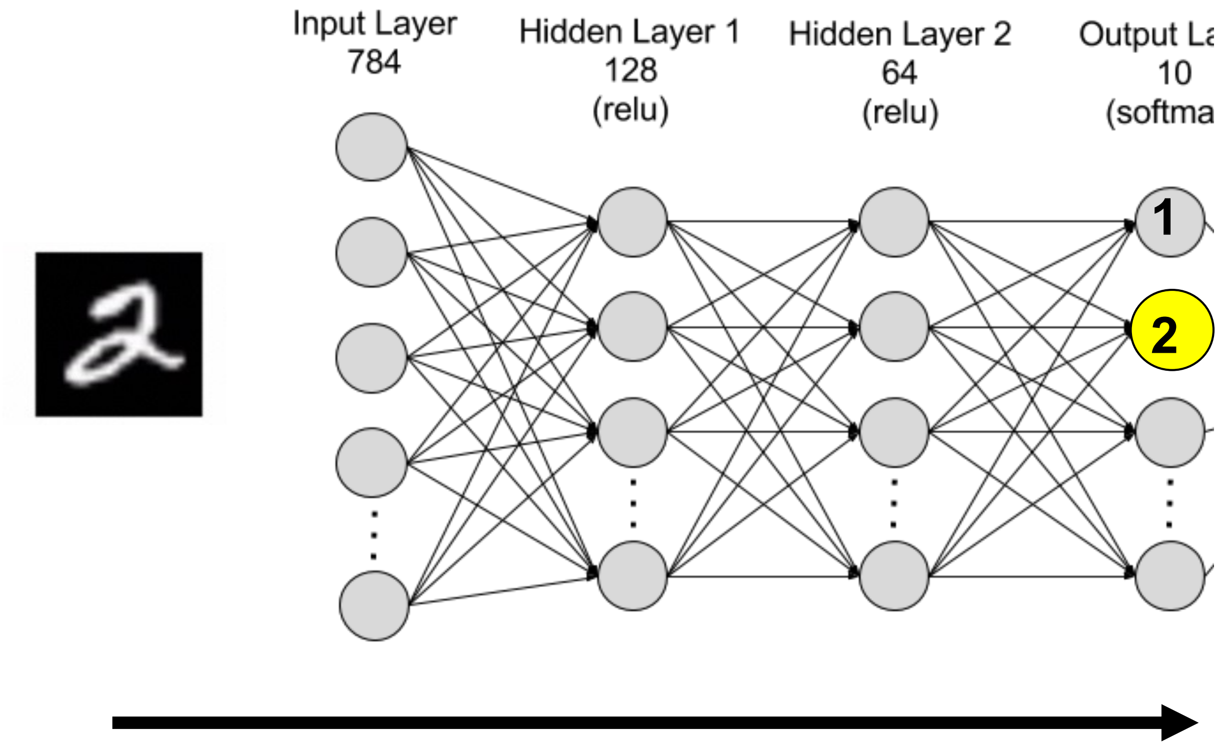
a flat 1-D structure of 784 (28*28) pixel value

- SoftMax (output layer) maps its input to a probability score for each class of output



Cagri Ozcinar, PhD

Practical example: MNIST

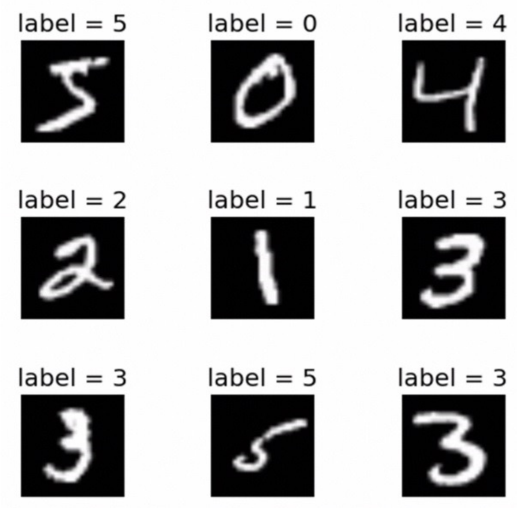


Cagri Ozcinar, PhD

Practical example: MNIST

- Datasets in deep learning can be slip into **training set** and **test set**.

- Training set consists of inputs each corresponding to some answer/label.

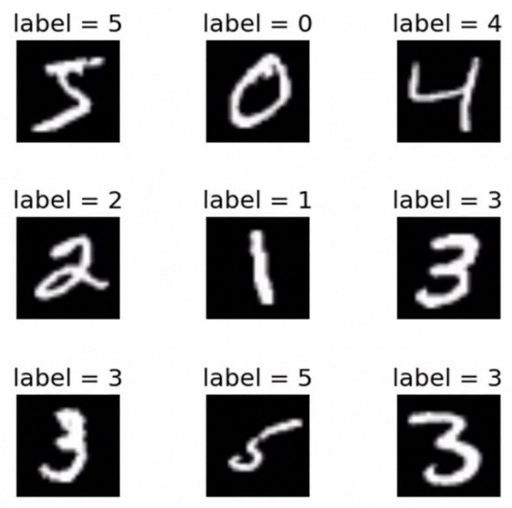


Cagri Ozcinar, PhD

Practical example: MNIST

- Datasets in deep learning can be slip into **training set** and **test set**.

- Training set consists of inputs each corresponding to some answer/label.



Practical example: MNIST

- Test set consists of inputs each corresponding to some answer/label. It consists of new images that never seen before

- In ML, we are interested in how well the ML algorithm performs on data that its never seen before

- It determines the performance in the real world.

Practical example: MNIST

- During the training we also require the test error
- When we train our deep neural network, the model tends to memorize the training data.
- Generalization

Generalization

- Small training error
- The gap between the training error and the test error should be small

Cagri Ozcinar, PhD

Generalization

- Large training error

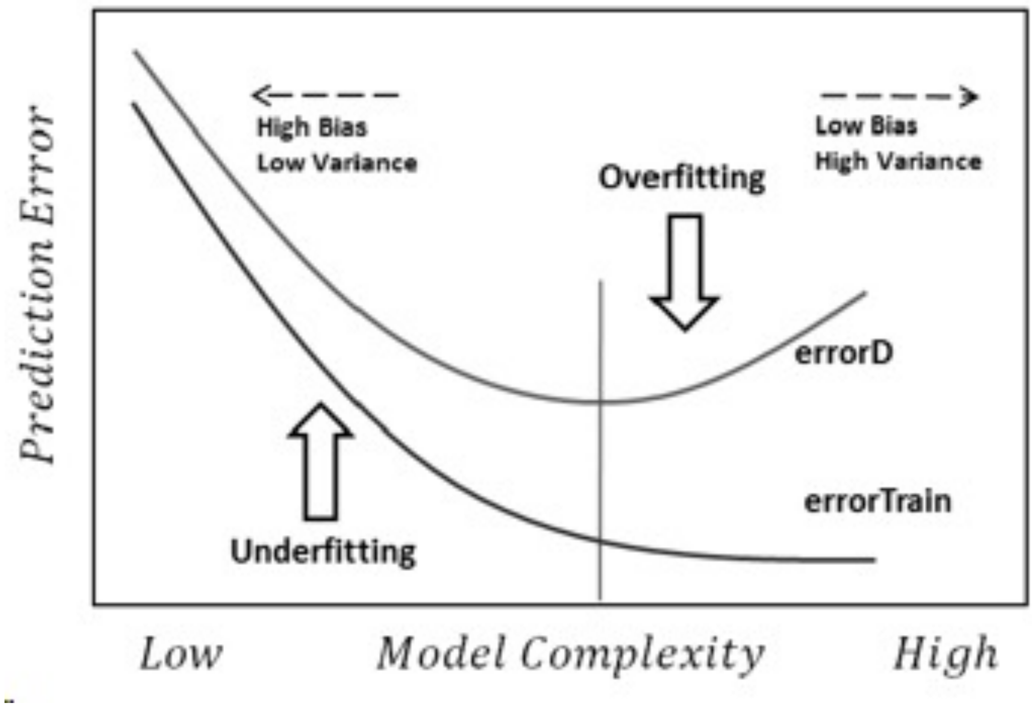
Challenge: under fitting

- Gap between test and training error grows larger

Challenge: over fitting

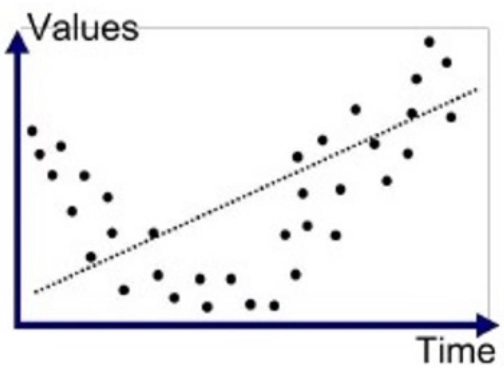
Generalization

- At the beginning, the function is too simple to capture the underlying trend of the data

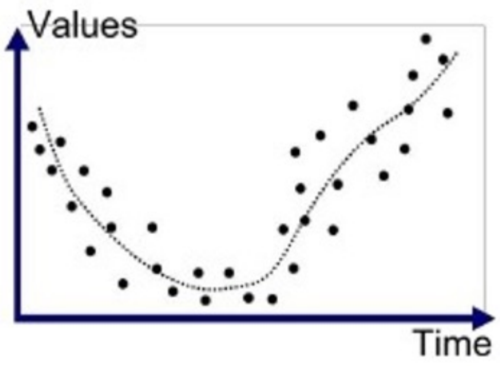


Under fitting

- It is not able to fit the training set
- The network does not have enough capacity to fit the training data



Underfitting

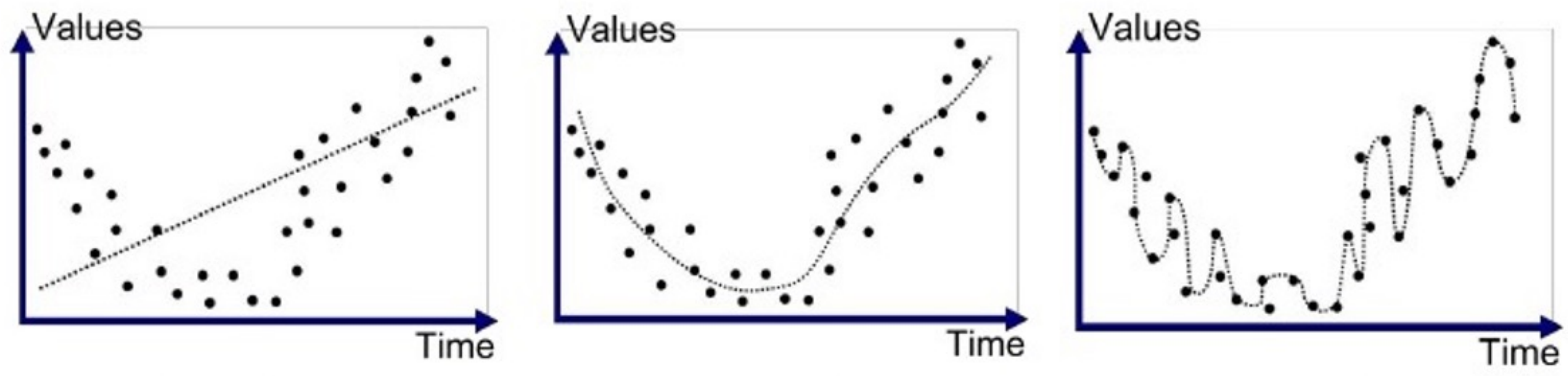


Good fit

Cagri Ozcinar, PhD

Over fitting

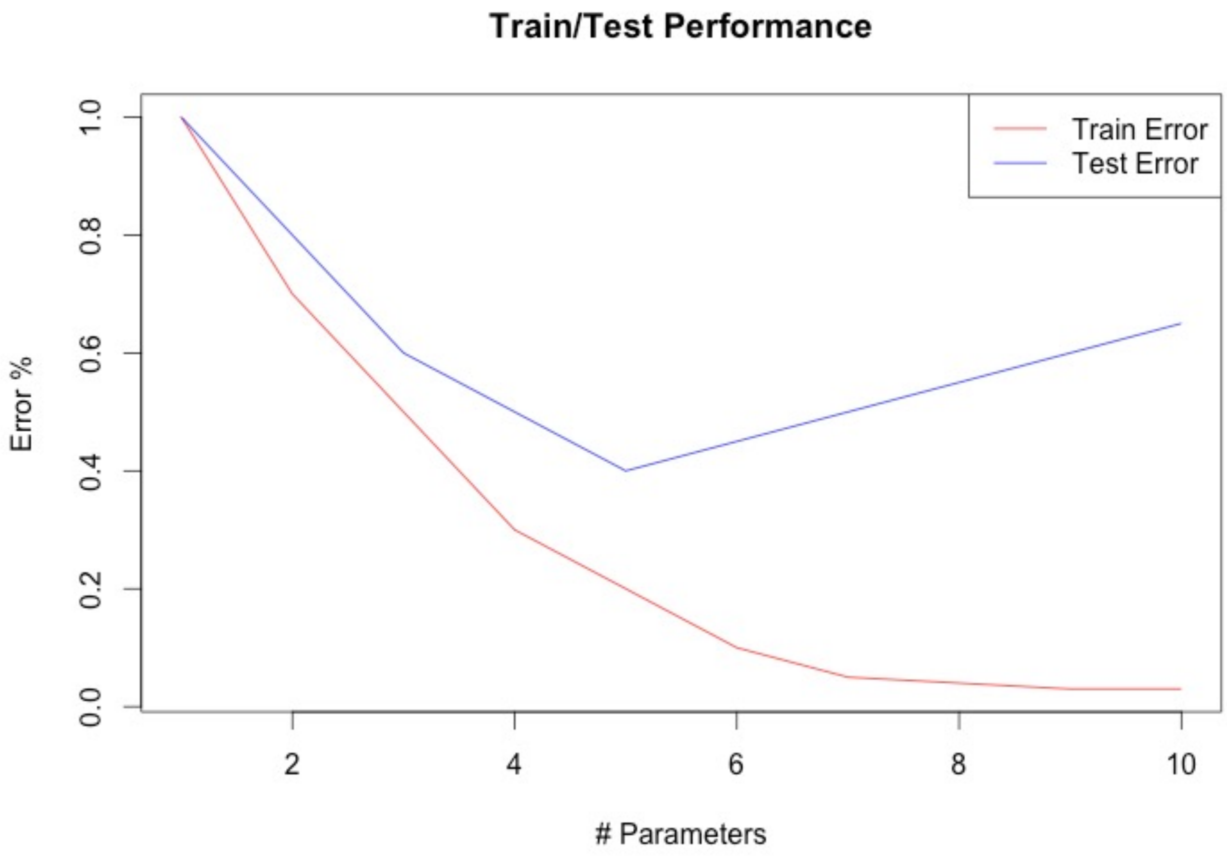
- If the capacity is too high, the network might over fit which introduce the concept of <u>**over fitting**</u>
- It memorize the training data, so when there is a new data, it is not able to effectively generalize to new data



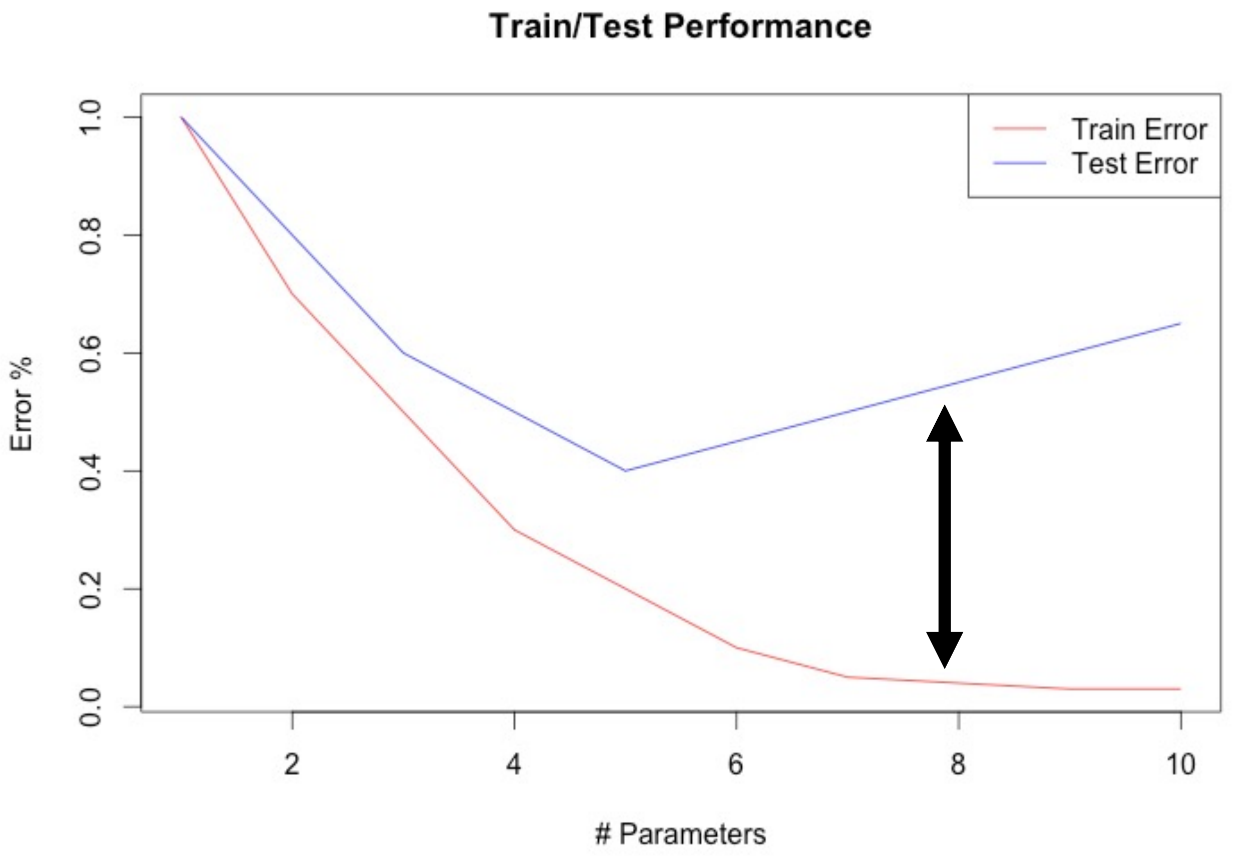
Underfitting **Good fit** **Overfitting**

Cagri Ozcinar, PhD

Generalization



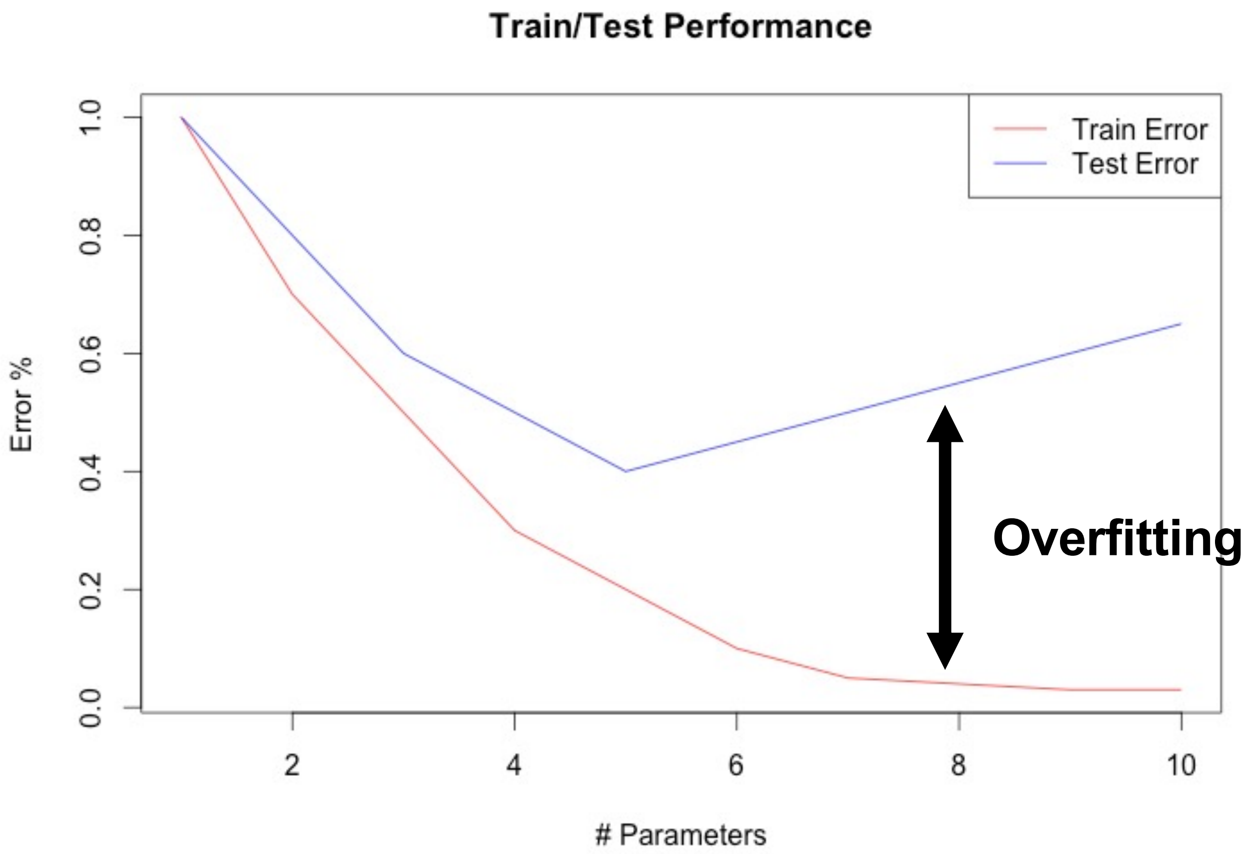
Generalization



1. **Small training error**
2. **Gap between test and training error grows larger**

Cagri Ozcinar, PhD

Generalization



1. **Small training error**
2. **Gap between test and training error grows larger**

Cagri Ozcinar, PhD

How can we fix over fitting?

- Reduce model complexity
- Reduce number of features
- Less epochs
- Mora data (clean and relevant)
- ….

Cagri Ozcinar, PhD

How can we fix under fitting?

- Increase model complexity
- Increase number of features
- More epochs (increase number of epochs)
- Remove noise from the data
- ….