
#DLUPC

Day 6 Lecture 2

# Methodology

[course site]

Javier Ruiz Hidalgo
javier.ruiz@upc.edu

Associate Professor
Universitat Politecnica de Catalunya
Technical University of Catalonia

# Outline

- **Data**
  - training, validation, test partitions
  - Augmentation
- **Capacity of the network**
  - Underfitting
  - Overfitting
- **Prevent overfitting**
  - Dropout, regularization
- **Strategy**

# Outline


validation
machine
Augmentation
deep
test
Data
network
Dropout
Underfitting
Capacity
training
Overfitting
Regularization
Partition

# It’s all about the data...

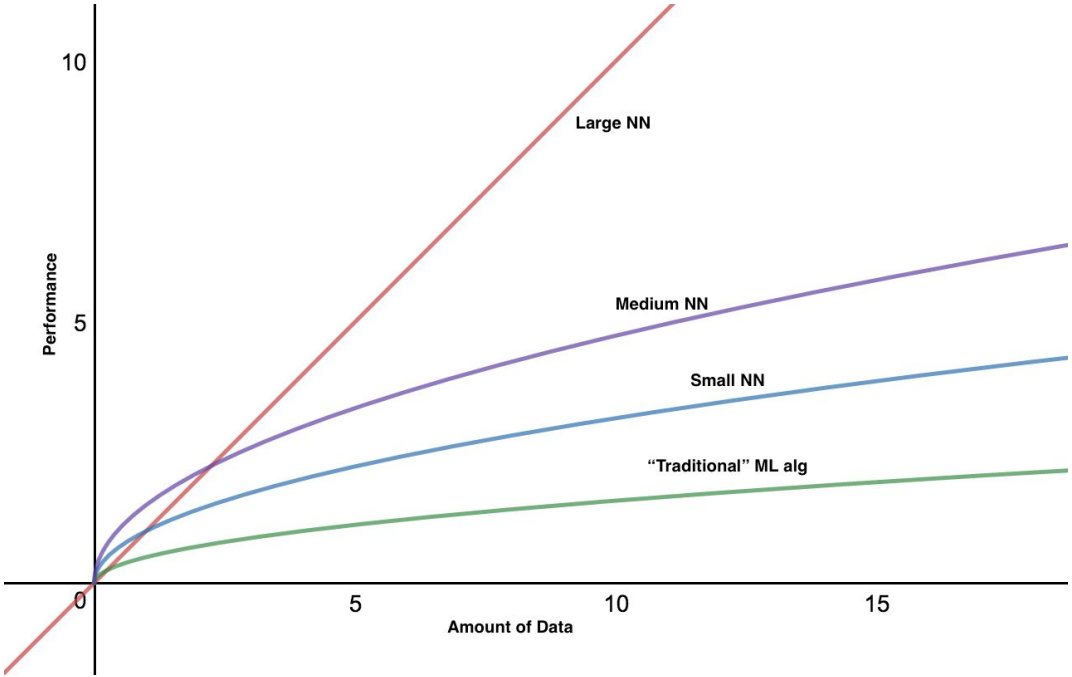
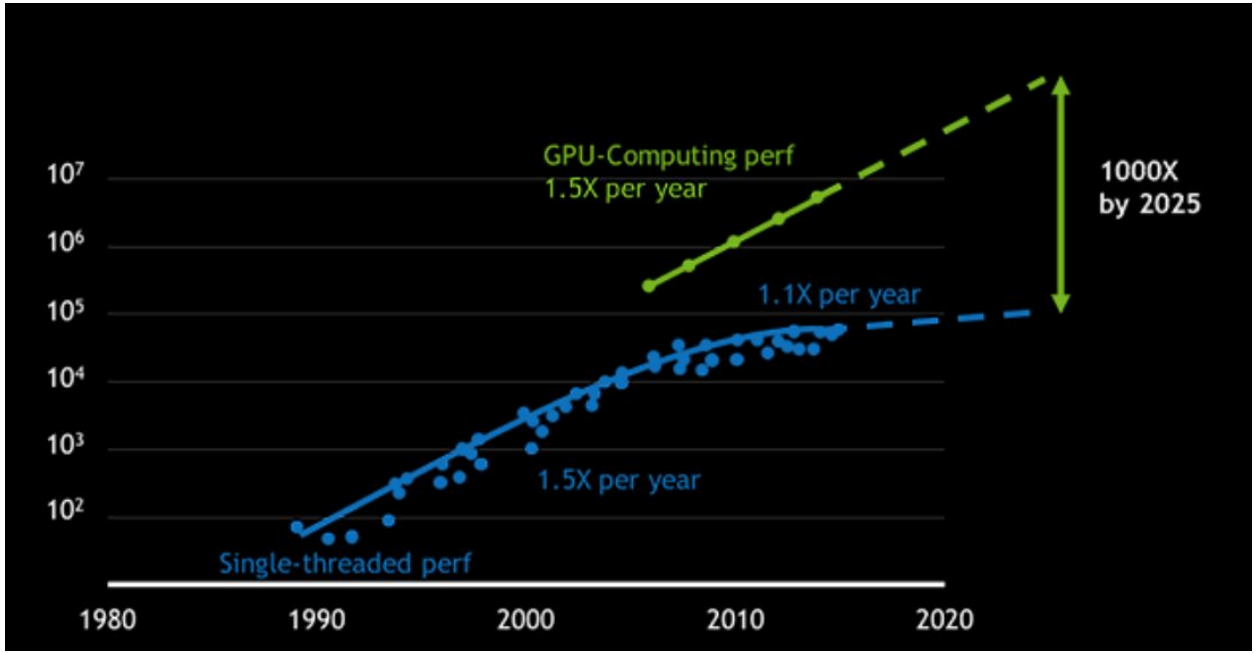




Figure extracted from Kevin Zakka's Blog, Nuts and Bolts of Applying Deep Learning, 2016.

# well, not only data...

- Computing power: GPUs




Source: NVIDIA 2017

# well, not only data...

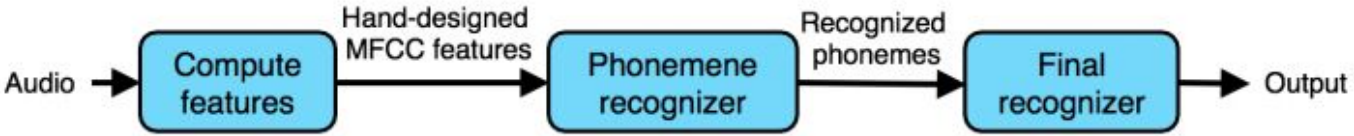
- Computing power: GPUs

- New learning architectures
  - CNN, RNN, LSTM, DBN, GNN, GAN, etc.

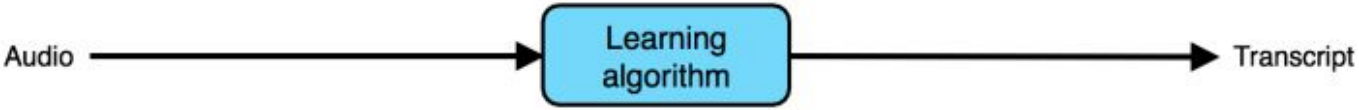
http://www.asimovinstitute.org/neural-network-zoo/

# End-to-end learning: speech recognition

Traditional model




End-to-end learning

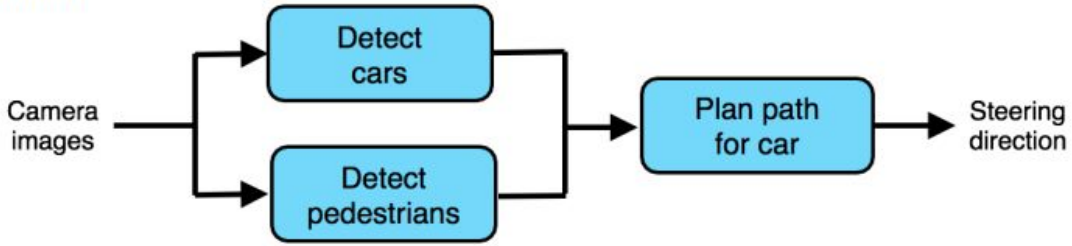



This works well given enough labeled (audio, transcript) data.

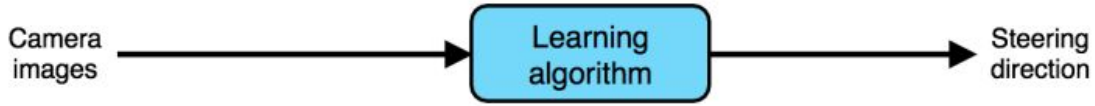
Slide extracted from Andrew Ng NIPS 2016 Tutorial

# End-to-end learning: autonomous driving

Traditional model




End-to-end learning




Given the safety-critical requirement of autonomous driving and thus the need for extremely high levels of accuracy, a pure end-to-end approach is still challenging to get to work. End-to-end works only when you have enough (x,y) data to learn function of needed level of complexity.

Slide extracted from Andrew Ng NIPS 2016 Tutorial

# Network capacity

- Space of representable functions that a network can potencially learn:
  - Number of layers / parameters

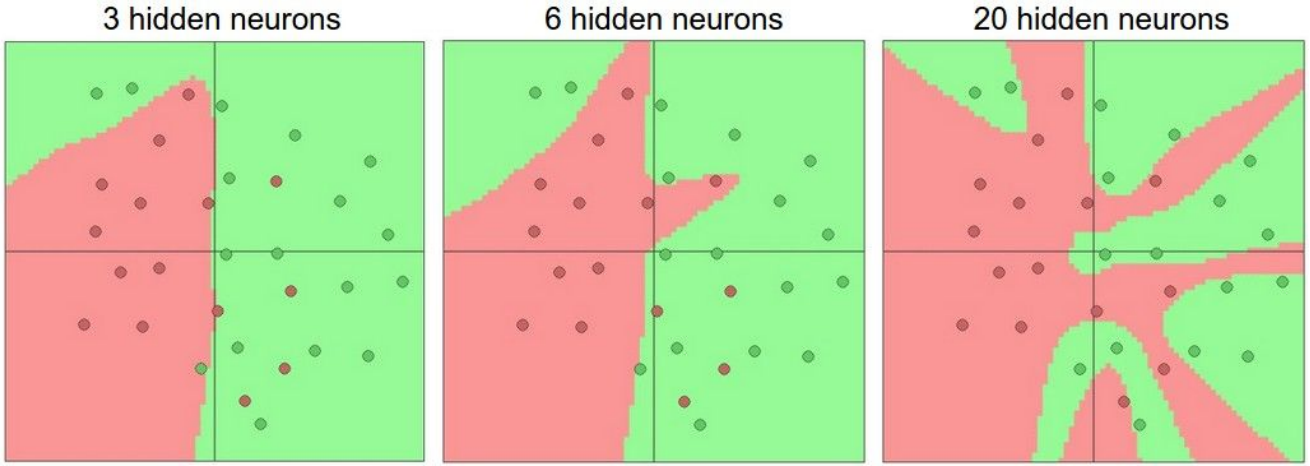
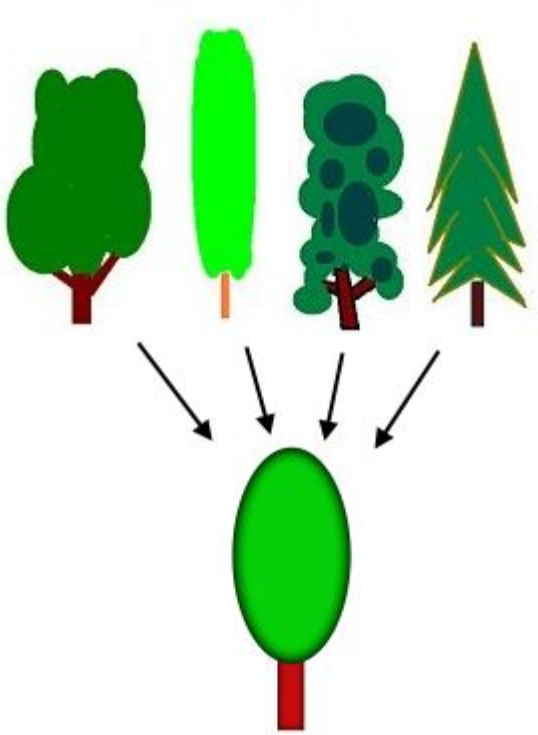



Figure extracted from Stanford CS class CS231n: Convolutional Neural Networks for Visual Recognition.

# Generalization

The network needs to **generalize** beyond the training data to work on new data that it has not seen yet



# Underfitting vs Overfitting

- **Overfitting**: network fits training data too well
  - Excessively complicated model
- **Underfitting**: network does not fit training data well enough
  - Excessively simple model
- Both underfitting and overfitting lead to poor predictions on new data and they do not generalize well

# Underfitting vs Overfitting

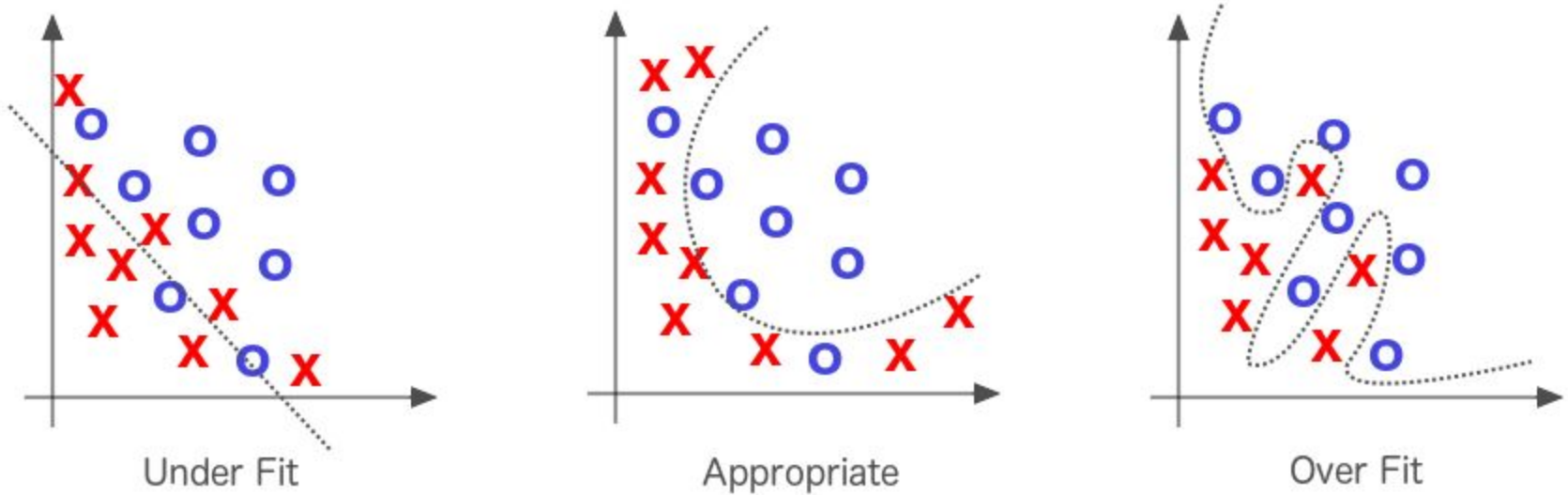



Figure extracted from Deep Learning by Adam Gibson, Josh Patterson, O'Reilly Media, Inc., 2017

# Data partition

How do we measure the generalization instead of how well the network does with the memorized data?

Split your data into two sets: training and test

# Underfitting vs Overfitting

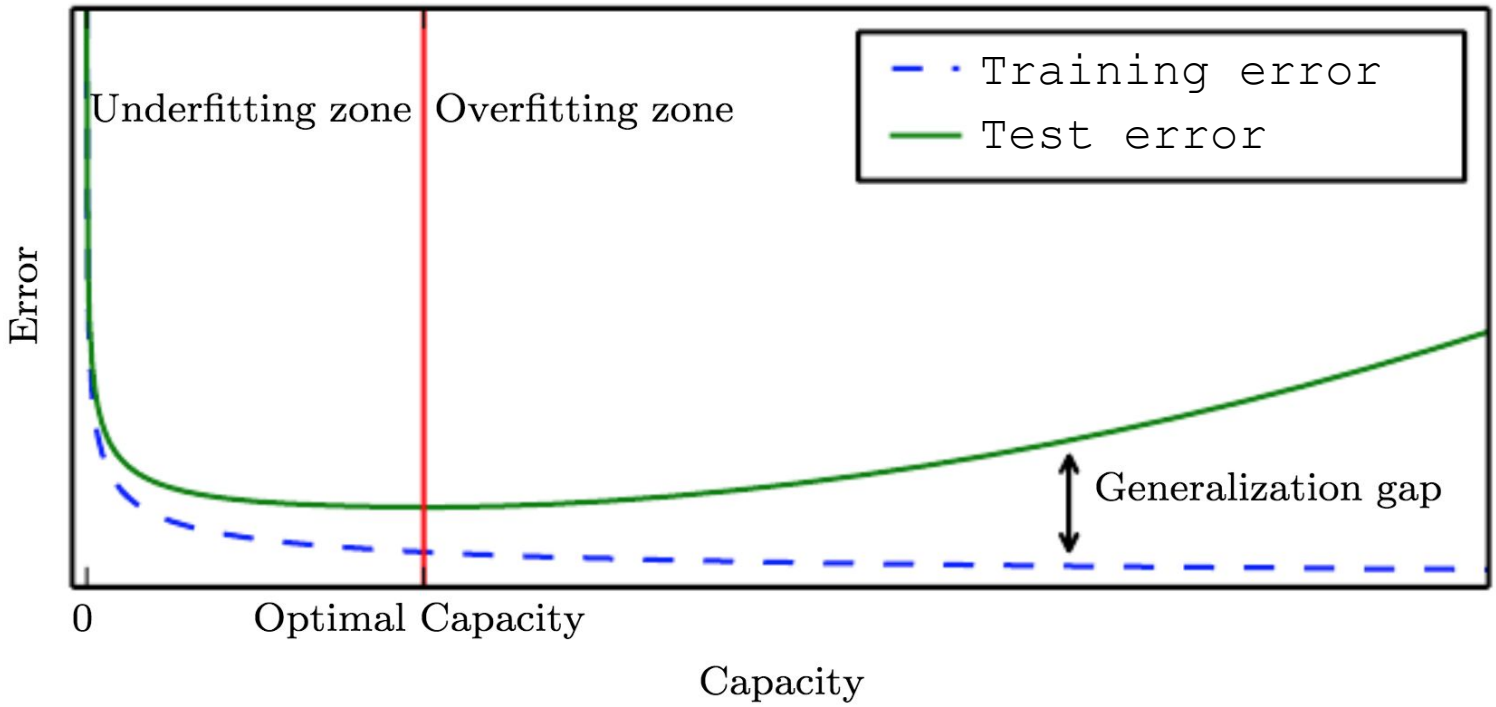



Figure extracted from Deep Learning by Ian Goodfellow and Yoshua Bengio and Aaron Courville, MIT Press, 2016

# Data partition revisited

- Test set **should not** be used to tune your network
  - Network architecture
  - Number of layers
  - Hyper-parameters

- Failing to do so will overfit the network to your test set!
  - https://www.kaggle.com/c/higgs-boson/leaderboard

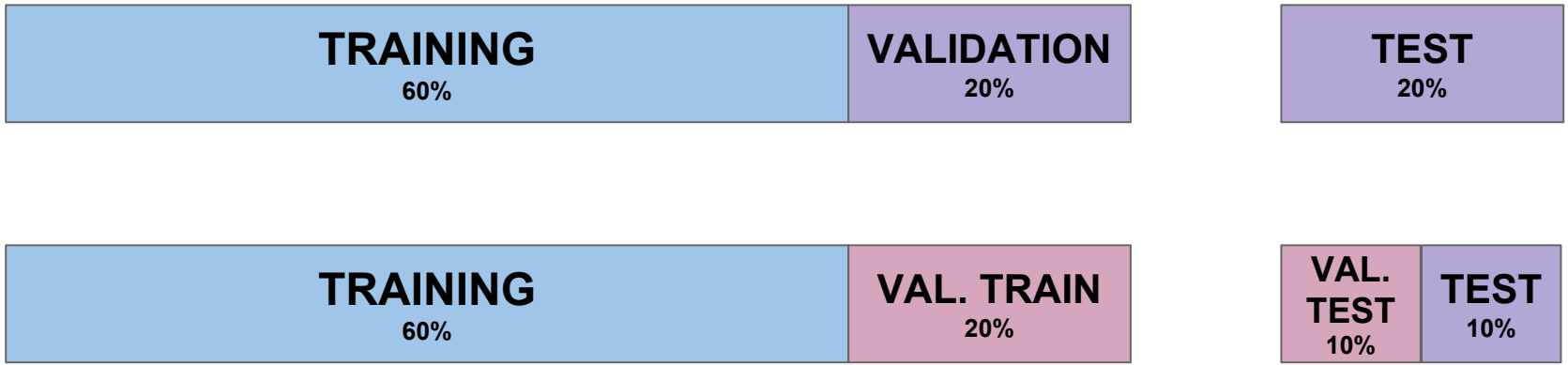
# Data partition revisited (2)

- Add a **validation** set!


- Lock away your test set and use it only as a last validation step

# Data sets distribution

- Take into account the distribution of training and test sets



# The bigger the better?

- Large networks
  - More capacity / More data
  - Prone to overfit
- Smaller networks
  - Lower capacity / Less data
  - Prone to underfit

# The bigger the better?

- In large networks, most local minima are equivalent and yield similar performance.
- The probability of finding a “bad” (high value) local minimum is non-zero for small networks and decreases quickly with network size.
- Struggling to find the global minimum on the training set (as opposed to one of the many good local ones) is not useful in practice and may lead to overfitting.

**Better large capacity networks and prevent overfitting**

Anna Choromanska et.al. “The Loss Surfaces of Multilayer Networks”, https://arxiv.org/pdf/1412.0233.pdf

# Prevent overfitting

- Early stopping
- Loss regularization
- Data augmentation
- Dropout
- Parameter sharing
- Adversarial training

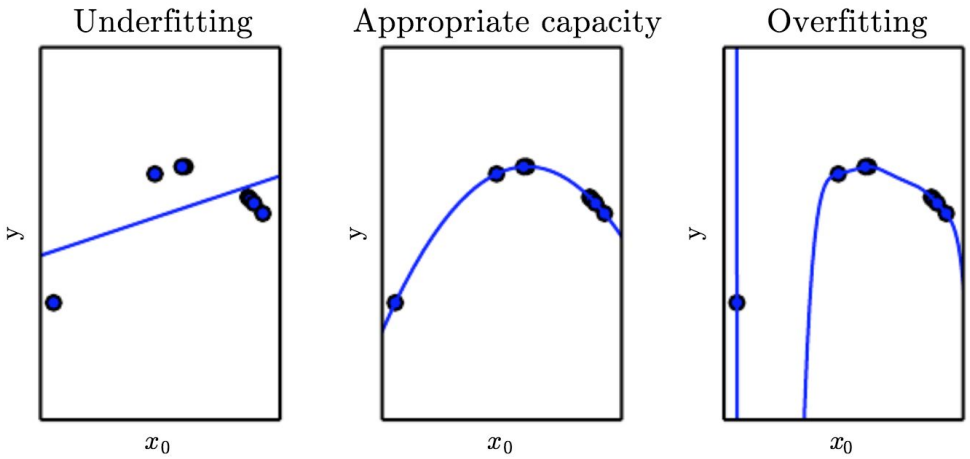
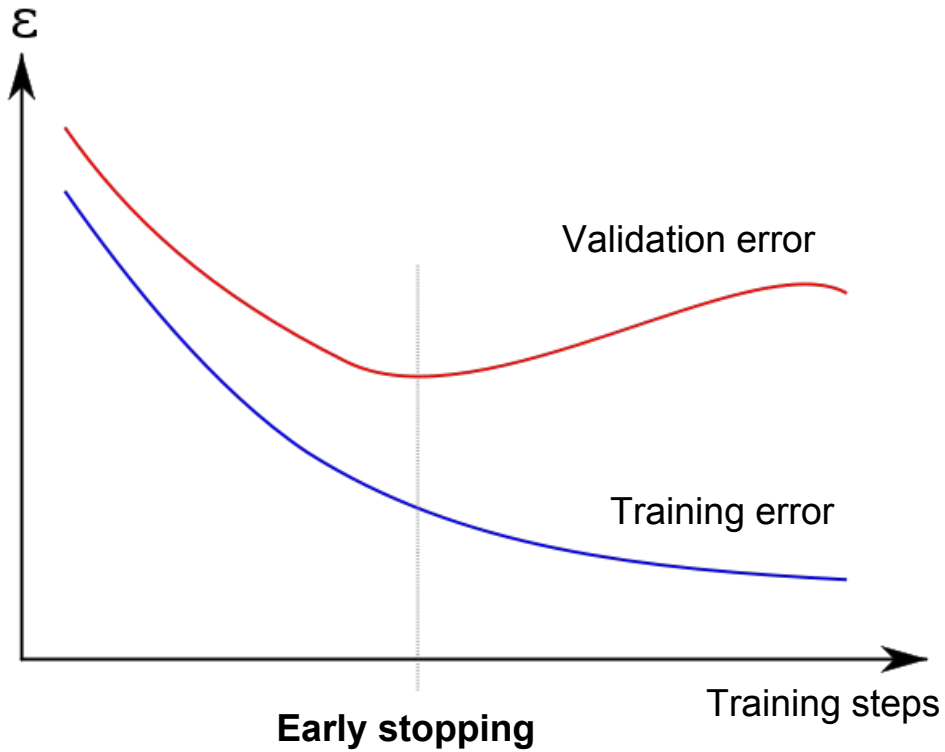



Figure extracted from Deep Learning by Ian Goodfellow and Yoshua Bengio and Aaron Courville, MIT Press, 2016

# Early stopping

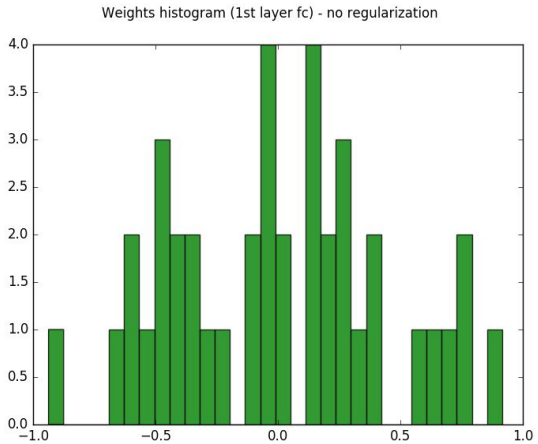


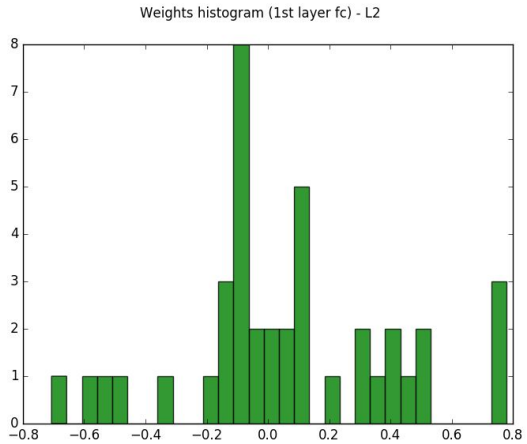
# Loss regularization

- Limit the values of parameters in the network
  - L2 or L1 regularization

$$\mathcal{L}_{new} = \mathcal{L} + \frac{\lambda}{2} W^2$$

$$\mathcal{L}_{new} = \mathcal{L} + \frac{\lambda}{2} |W|$$



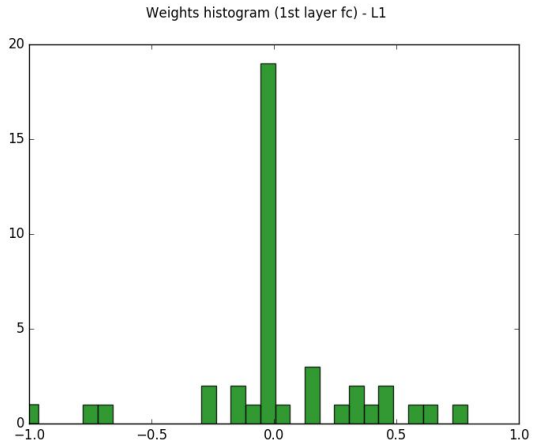




Figure extracted from Cristina Scheau, Regularization in Deep Learning, 2016

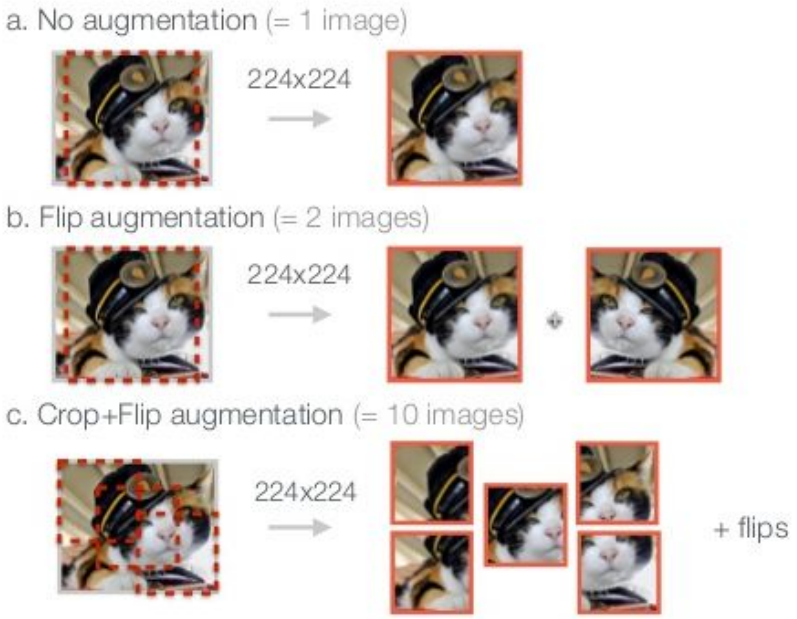
# Data augmentation (1)

- Alterate input samples artificially to increase the data size
- On-the-fly while training
  - Inject Noise
  - Transformations
  - ...


Alex Krizhevsky, ImageNet Classification with Deep Convolutional Neural Networks, NIPS 2012

# Data augmentation (2)

- Image
  - Random crops
  - Translations
  - Flips
  - Color changes
- Audio
  - Tempo perturbation, speed
- Video
  - Temporal displacement



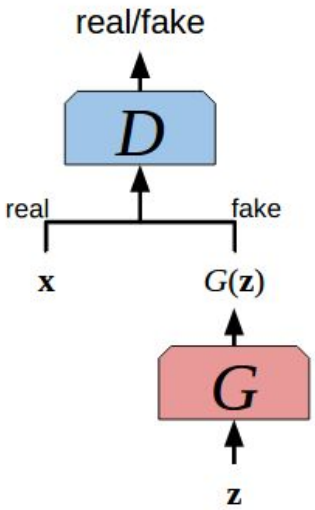
# Data augmentation (3)

- Synthetic data: Generate new input samples


A. Palazzi, Learning to Map Vehicles into Bird's Eye View, ICIAP 2017
DeepGTAV plugin: https://github.com/ai-tor/DeepGTAV

# Data augmentation (4)

- GANs (Generative Adversarial Networks)




P. Ferreira,et.al., Towards data set augmentation with GANs, 2017.
L. Sixt, et.al., RenderGAN: Generating Realistic labeled data, ICLR 2017.

# Dropout (1)

- At each training iteration, randomly remove some nodes in the network along with all of their incoming and outgoing connections (N. Srivastava, 2014)

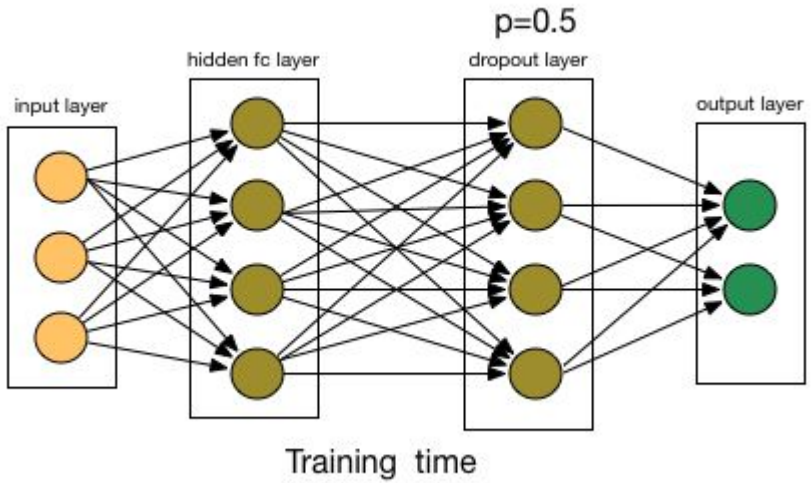



Figure extracted from Cristina Scheau, Regularization in Deep Learning, 2016

# Dropout (2)

- Why dropout works?
  - Nodes become more insensitive to the weights of the other nodes → more robust.
  - Averaging multiple models → **ensemble**.
  - Training a collection of $2^n$ thinned networks with parameters sharing

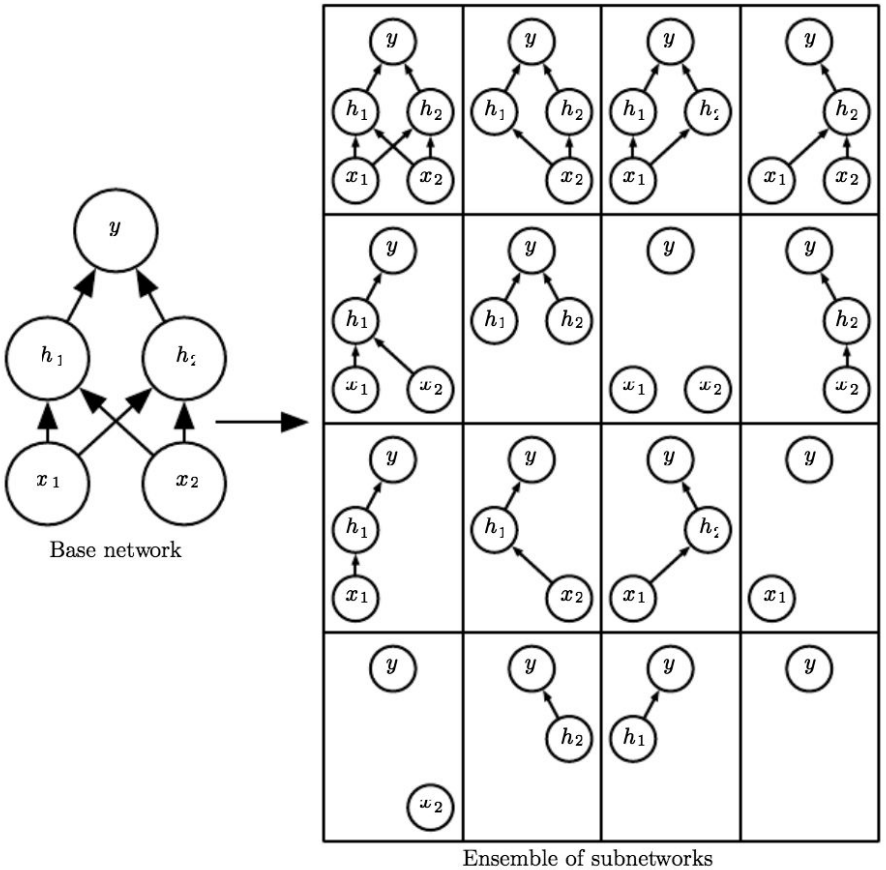
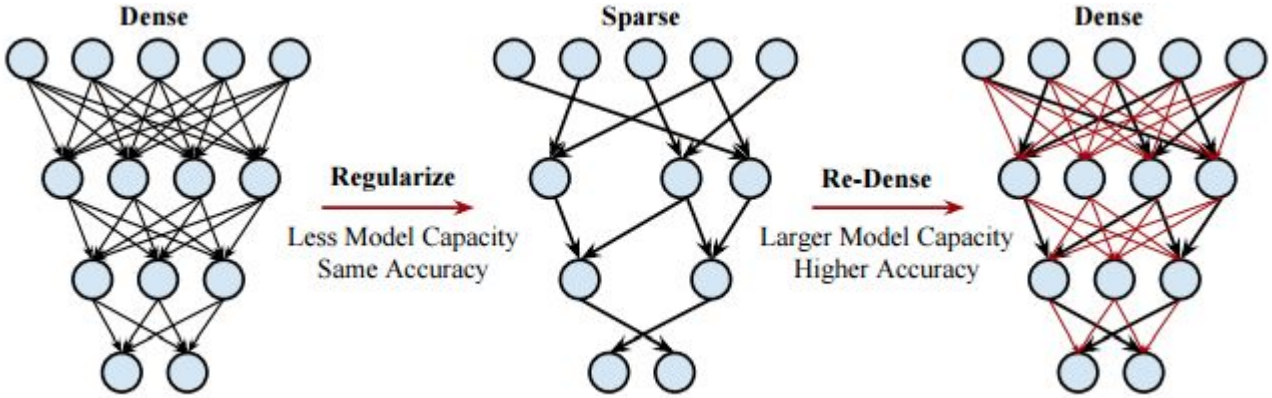



Figure extracted from Deep Learning by Ian Goodfellow and Yoshua Bengio and Aaron Courville, MIT Press, 2016

# Dropout (3)

- Dense-sparse-dense training (S. Han 2016)
  a. Initial regular training
  b. Drop connections where weights are under a particular threshold.
  c. Retrain sparse network to learn weights of important connections.
  d. Make network dense again and retrain using small learning rate, a step which adds back capacity.



# Parameter sharing

## Multi-task Learning

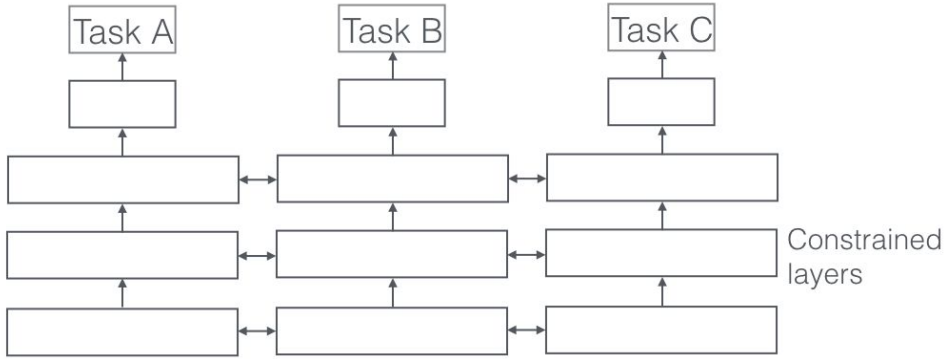
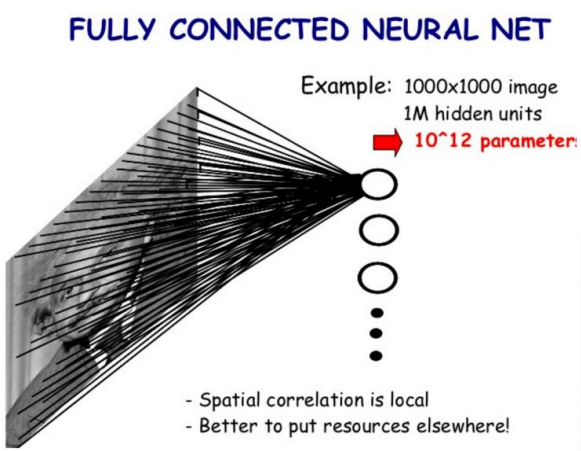
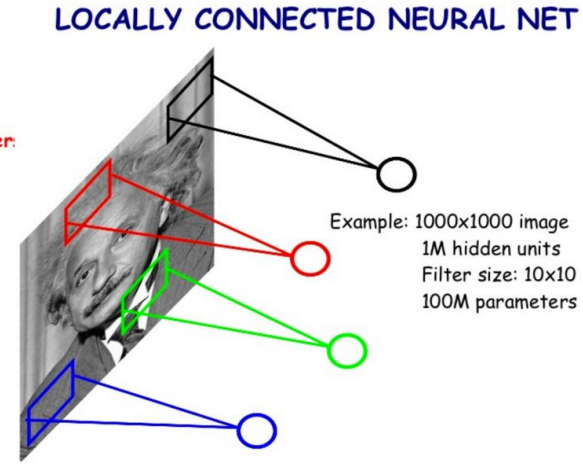



Figure extracted from Sebastian Ruder, An Overview of Multi-Task Learning in Deep Neural Networks, 2017



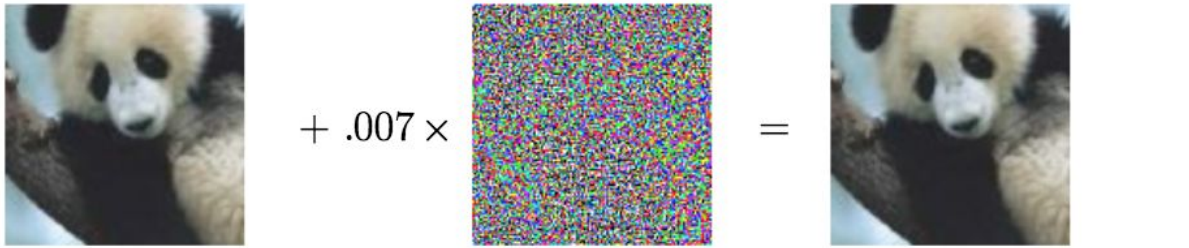




CNNs

Figure extracted from Leonardo Araujo, Artificial Intelligence, 2017

# Adversarial training

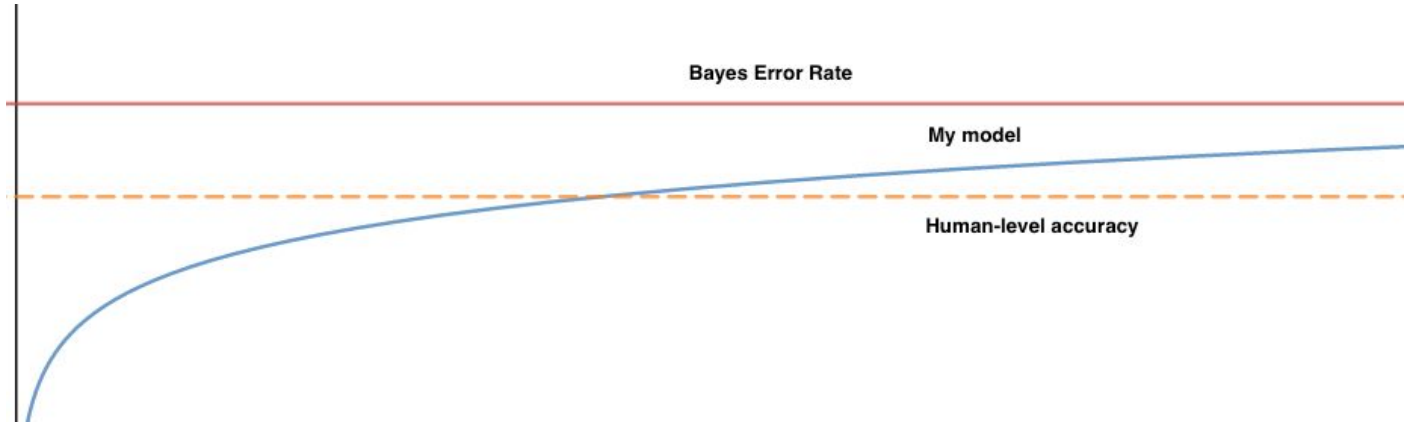
- Search for **adversarial examples** that network misclassifies
  - Human observer cannot tell the difference
  - However, the network can make highly different predictions.




I. Goodfellow, et.al., Explaining and Harnessing Adversarial Examples, ICLR, 2015

# Strategy for machine learning (1)

Human-level performance can serve as a very reliable proxy which can be leveraged to determine your next move when training your model.



# Strategy for machine learning (2)


Human level error . . 1%

Training error . . . 9%

Avoidable bias

# Strategy for machine learning (3)


Human level error . . 1%

Training error . . . 1.1%

Validation error . . 10%

Overfitting training

# Strategy for machine learning (4)


Human level error . . 1%

Training error . . . 1.1%

Validation error . . 2%

Test error . . . . . 11%

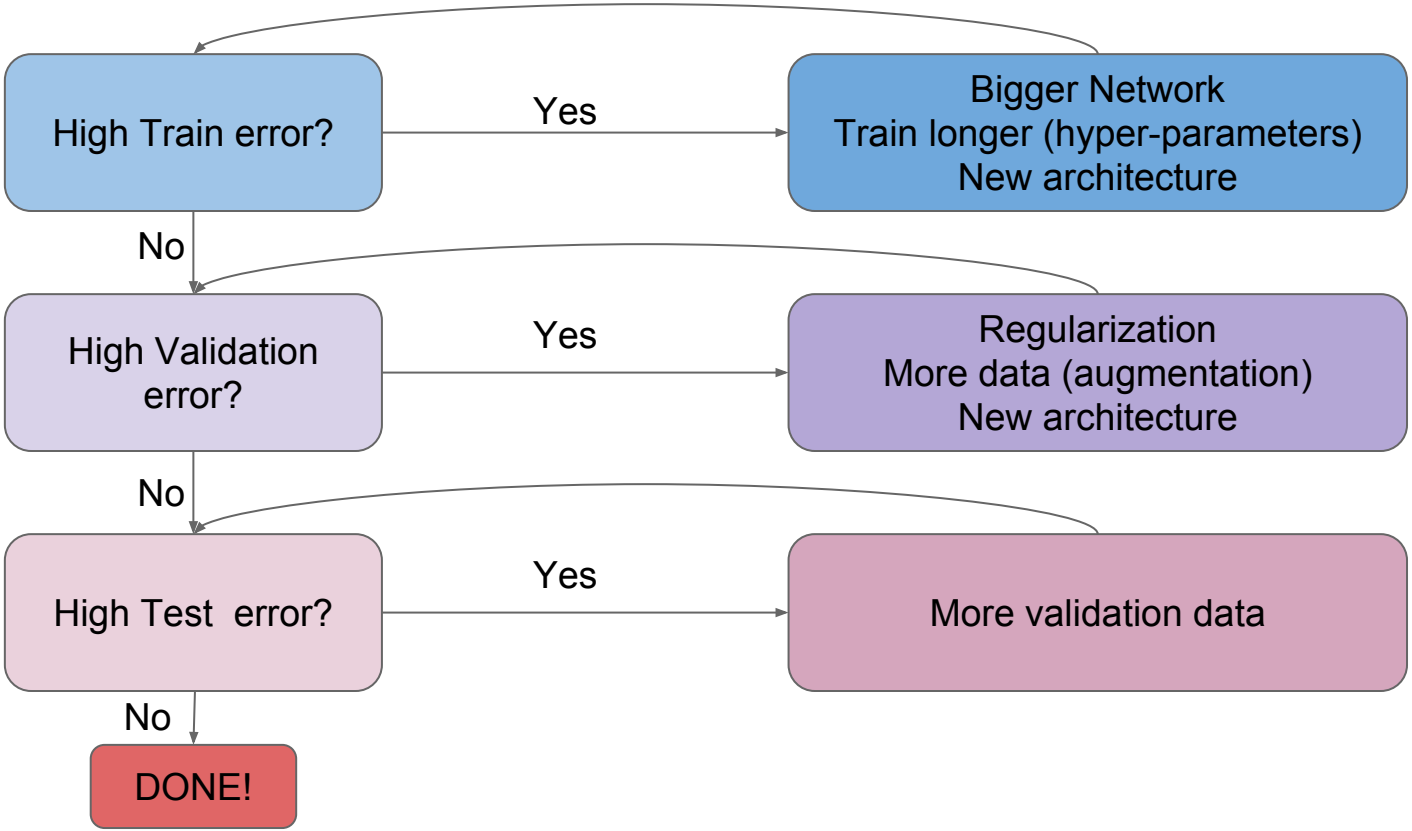
Overfitting validation

# Strategy for machine learning (5)

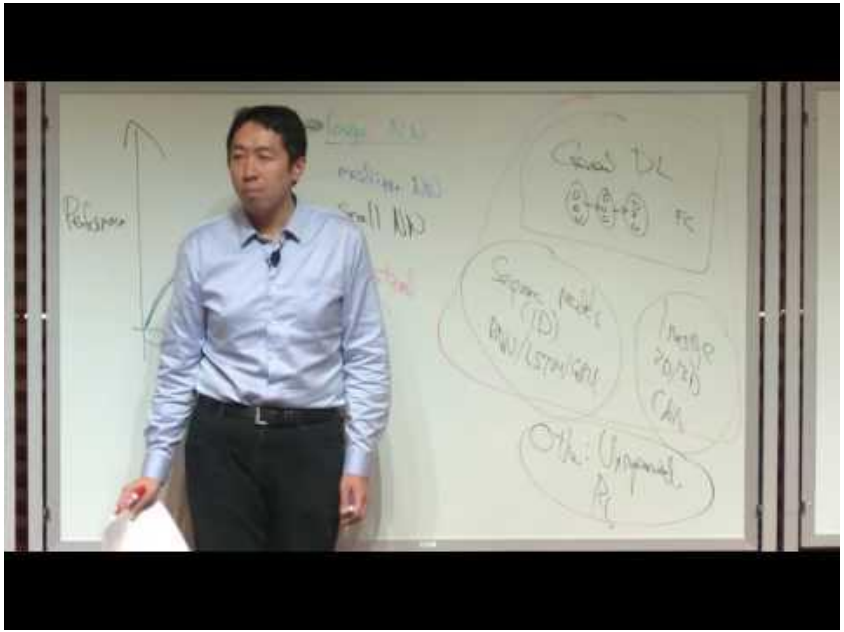
| TRAINING<br>60% | VALIDATION<br>20% | TEST<br>20% |
|---|---|---|

```
Human level error . . 1%
Training error  . . . 1.1%
Validation error  . . 1.2%
Test error  . . . . . 1.2%
```

# Strategy for machine learning (5)



# References



Nuts and Bolts of Applying Deep Learning  by Andrew Ng

https://www.youtube.com/watch?v=F1ka6a13S9I

# Thanks! Questions?


UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Department of Signal Theory
and Communications

*Image Processing Group*

https://imatge.upc.edu/web/people/javier-ruiz-hidalgo