ML Bootcamp

Ahmed Baruwa
Electronic and Electrical Engineering Dept.

Personal Details

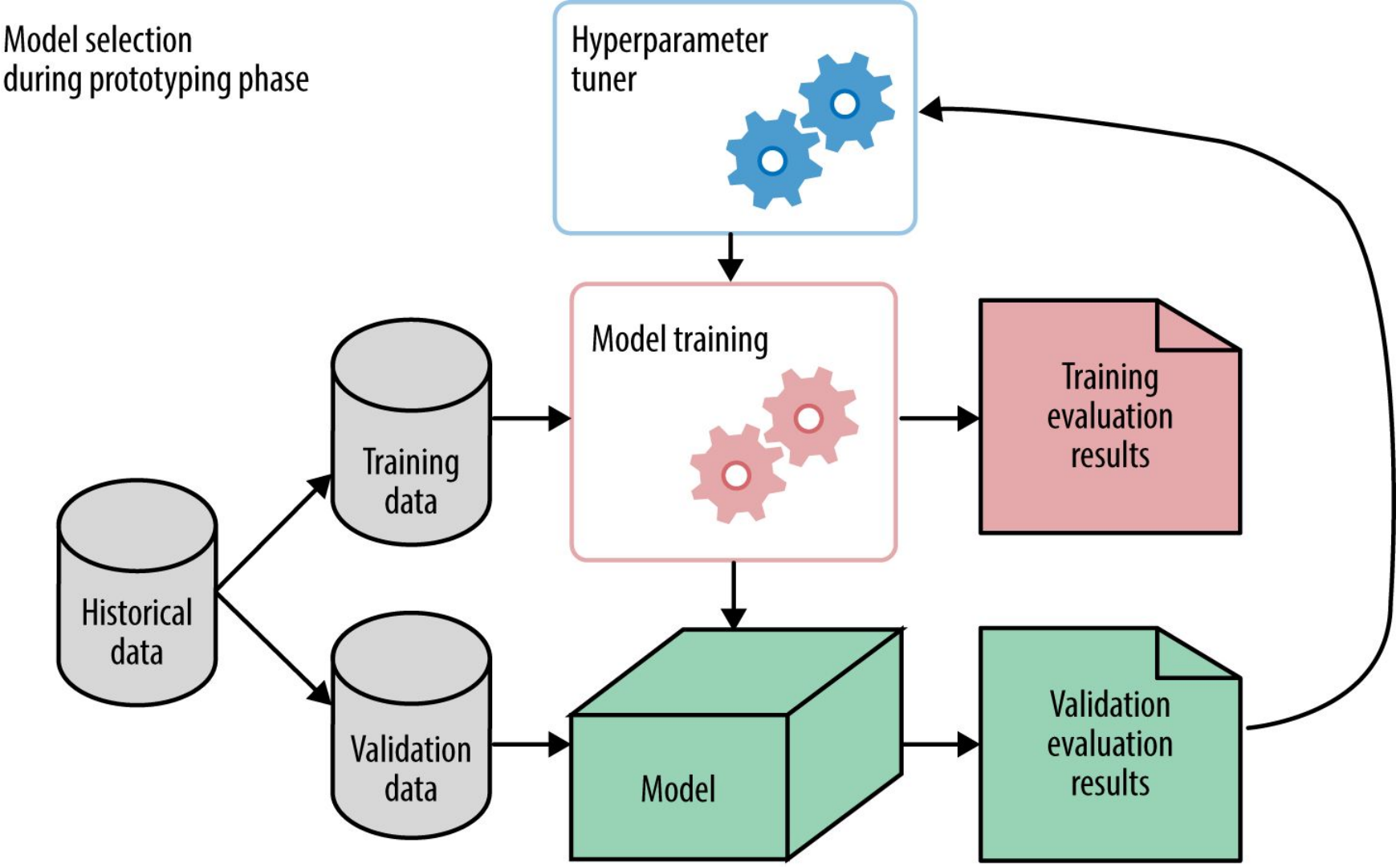
Mail- aobaruwa@yahoo.com

linkedIn profile - linkedin.com/in/ahmedbaruwa

Phone - (+234) 810 470 5408

Github - aobaruwa@github.com

(All materials for this session can be found in the github link)


Model selection
during prototyping phase
Hyperparameter
tuner
Model training
Training
evaluation
results
Training
data
Historical
data
Validation
data
Model
Validation
evaluation
results

Model Validation, Evaluation & Optimization

A model

A model is a black box that takes input data (numeric, text, image, audio, earthquake shock wave ...) and magically throws out an output.

What goes on in the model is what Machine Learning is all about.

Machine Learning

According to Wikipedia >>

ML is the scientific study of algorithms and statistical models that computer systems use to effectively perform a specific task without being explicitly programmed, relying on models and inference based on experience instead!

What Machine Learning is most amazing at

By creating (learning) models from data...

Machine learning helps us perform tasks we have no idea how to program in a procedural way!

Tasks like : Recognizing images, Recognizing speech, Chess program, Self driving cars, language translation

How machines learn from Experience

There are three common ways by which machines learn from experience (data):

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning

How machines Learn from Experience

- **Supervised learning** algorithms experience dataset containing many features as well as label (or target) associated with each example.

- In essence, given several examples of a random vector x, and associated labels (e.g images and their labels), learn to estimate the conditional distribution p(y|x)

How machines Learn from Experience

- **Unsupervised learning** algorithms experience dataset containing many features and are required to learn useful properties from the dataset.

- In essence, given several examples of a random vector x, implicitly or explicitly learn the probability distribution, p(x) (e.g Facebook's clustering algorithm for new friend suggestions)

How machines Learn from Experience

- **Reinforcement Learning**: Training and testing phases are intermixed. Learner collects information by actively interacting with the environment and sometime also affecting the environment, to receive immediate reward for each action. The goal is to maximize the reward over time. This learning scenario is related to dynamic programming. (e.g Chess program, Pacman Agent, Robot Learning to walk)

Learning Problems/Tasks

Examples of learning problems/tasks.

- **Classification** - Assign a category to each item.
- **Regression** - Predict a value for each item.
- **Ranking** - Order items according to some criterion. E.g. Web search.
- **Clustering** - Partition items into homogeneous regions.
- **Dimensionality reduction or manifold learning** - Transform an initial representation of items into a lower dimension of these items while retaining some properties.

Machine Learning Algorithms

A lot of Machine Learning Algorithms can be classified under the following broad categories:

- **Linear Models**
- **Tree based methods**
- **Nearest Neighbors**
- **Neural networks**

Linear Models

Linear models work on the premise

of splitting space into

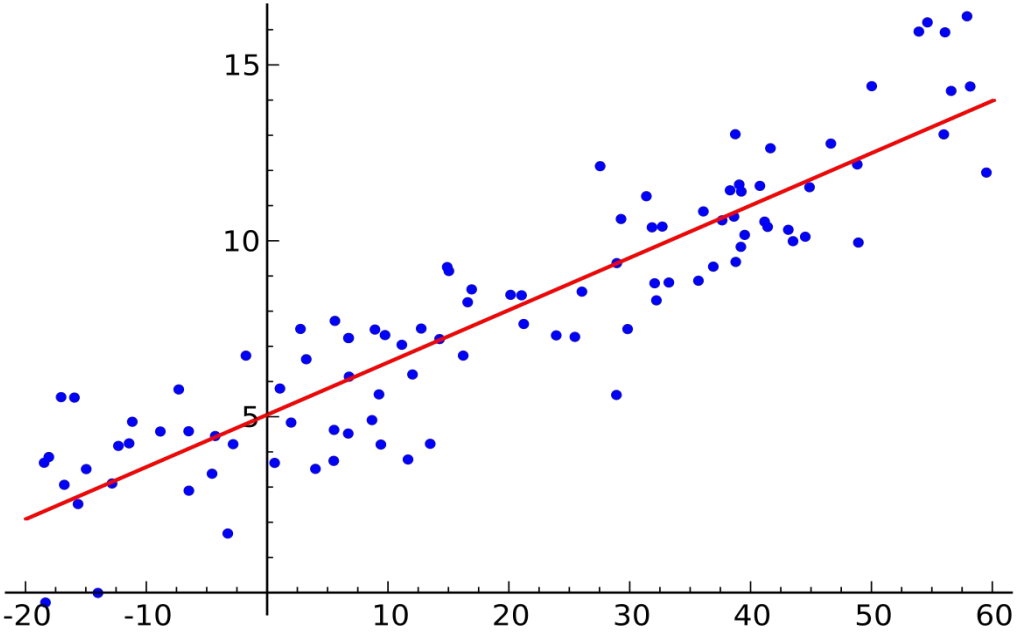
two subspaces by creating

a **linear decision boundary.**

Examples- linear regression,

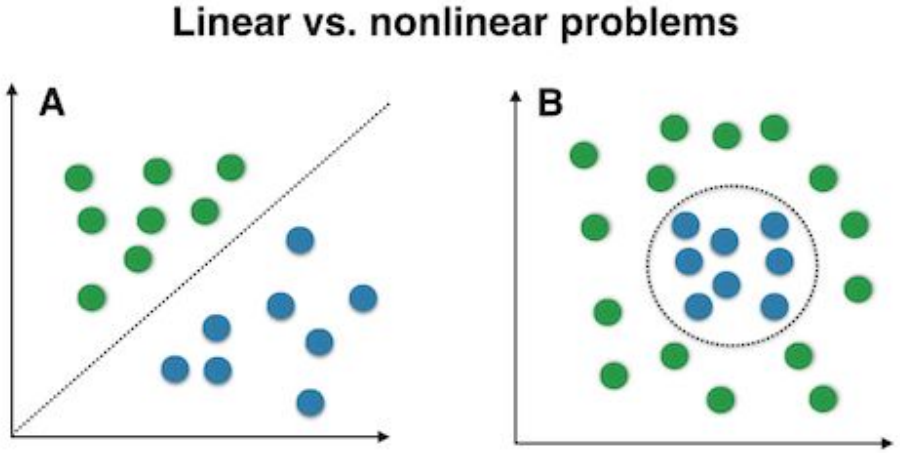
Logistic regression, Support vector

Machines etc.



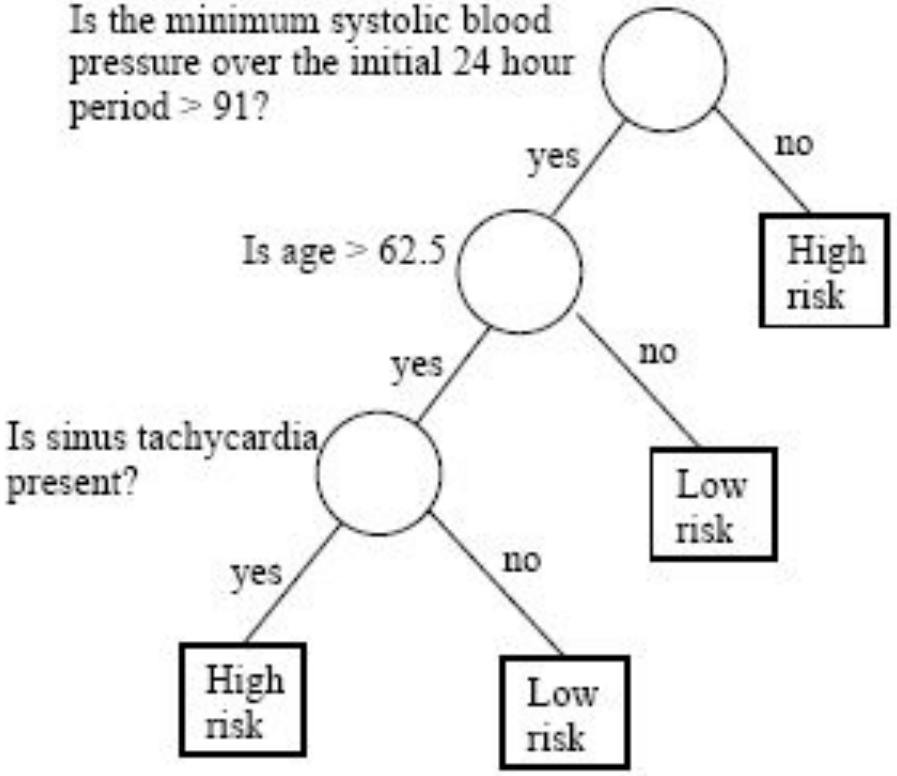
Linear Methods

Linear models have limitations- they fail in modelling complex (non linear types of data.

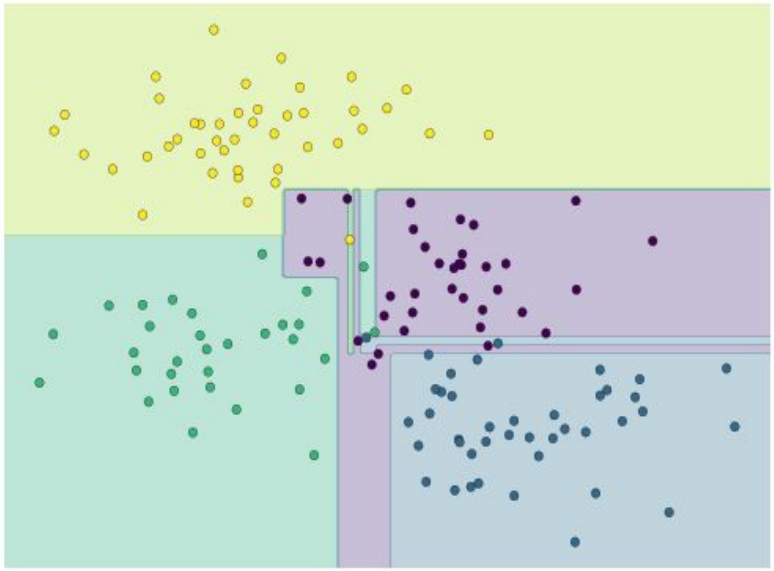


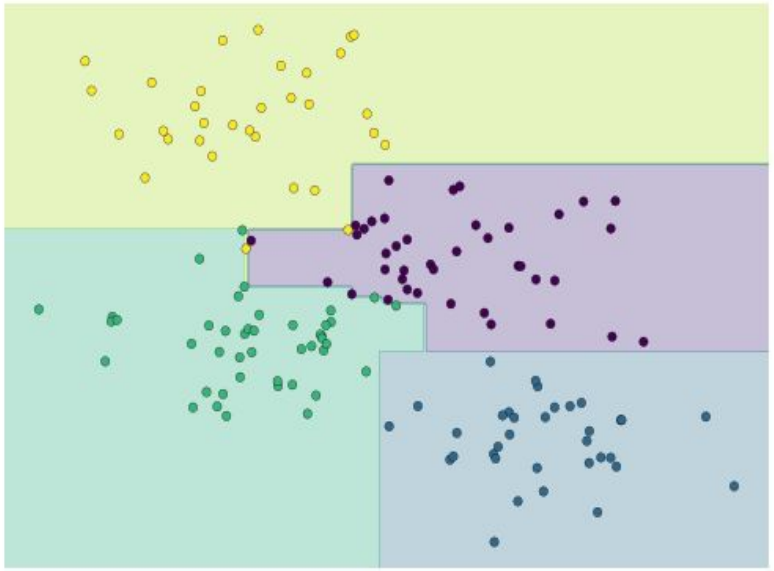
Tree-based Methods

Tree based methods use decision trees as the basic building block for building more complex models.



Tree-based methods





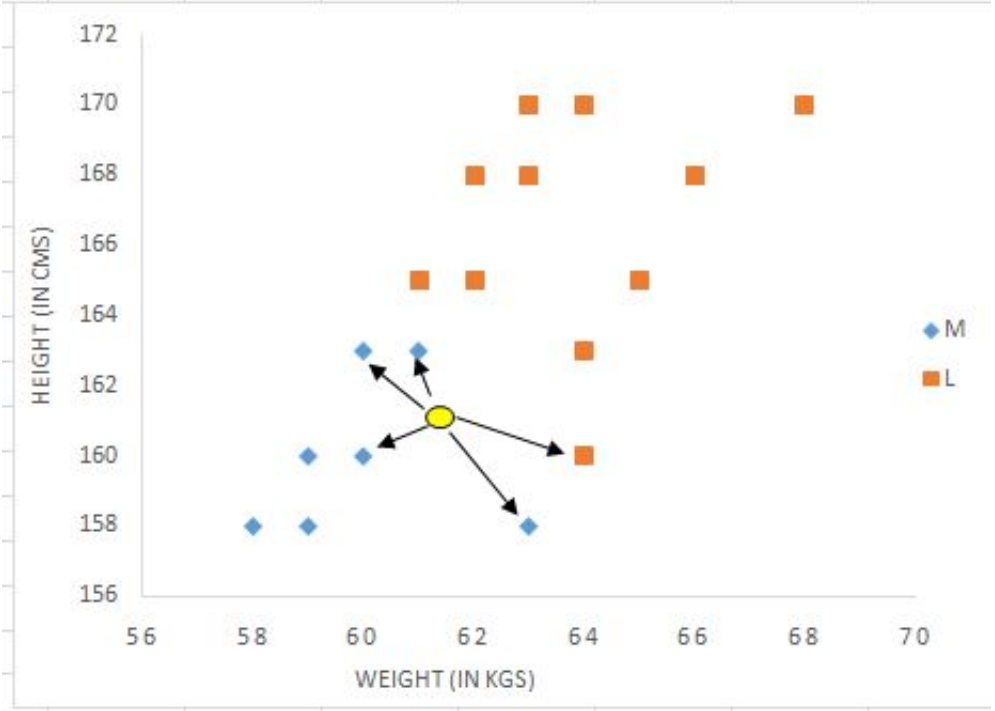
Nearest Neighbors

An unsupervised learning class

of algorithms e.g KDTree,

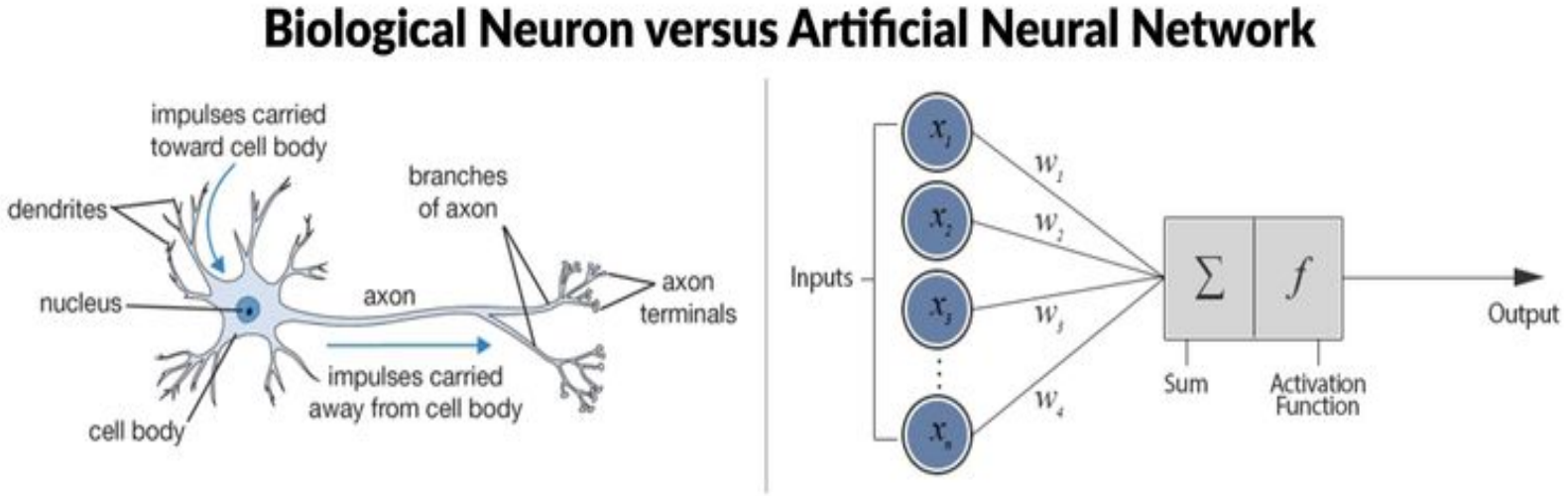
K-Nearest neighbors,

Nearest centroid etc.



Neural networks

A **bio-inspired** class of supervised learning algorithms that is by far one of the most efficient models for handling non-linear data.



Neural Networks

- Started gaining momentum when the task of **computer vision** was to be solved.
- Are the basic building block of a subfield of ML called **Deep learning** - what companies like Google use for speech recognition, Web search, Self driving cars, Machine Translation; Facebook for Tagging faces in pictures (10-years challenge!); Tesla autonomous vehicles heavily depends on computer vision for navigation.

Facebook's 10 years challenge (Joke)

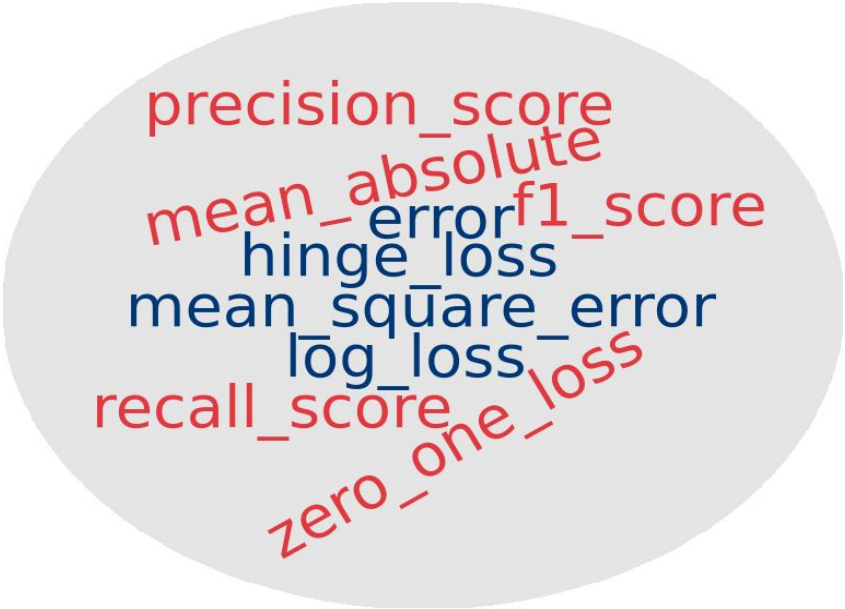
Evaluating the performance of a model

Associated with every ML algorithm is a **loss function** (a measure of the un-correctness of the output given an input) also known as performance metrics.

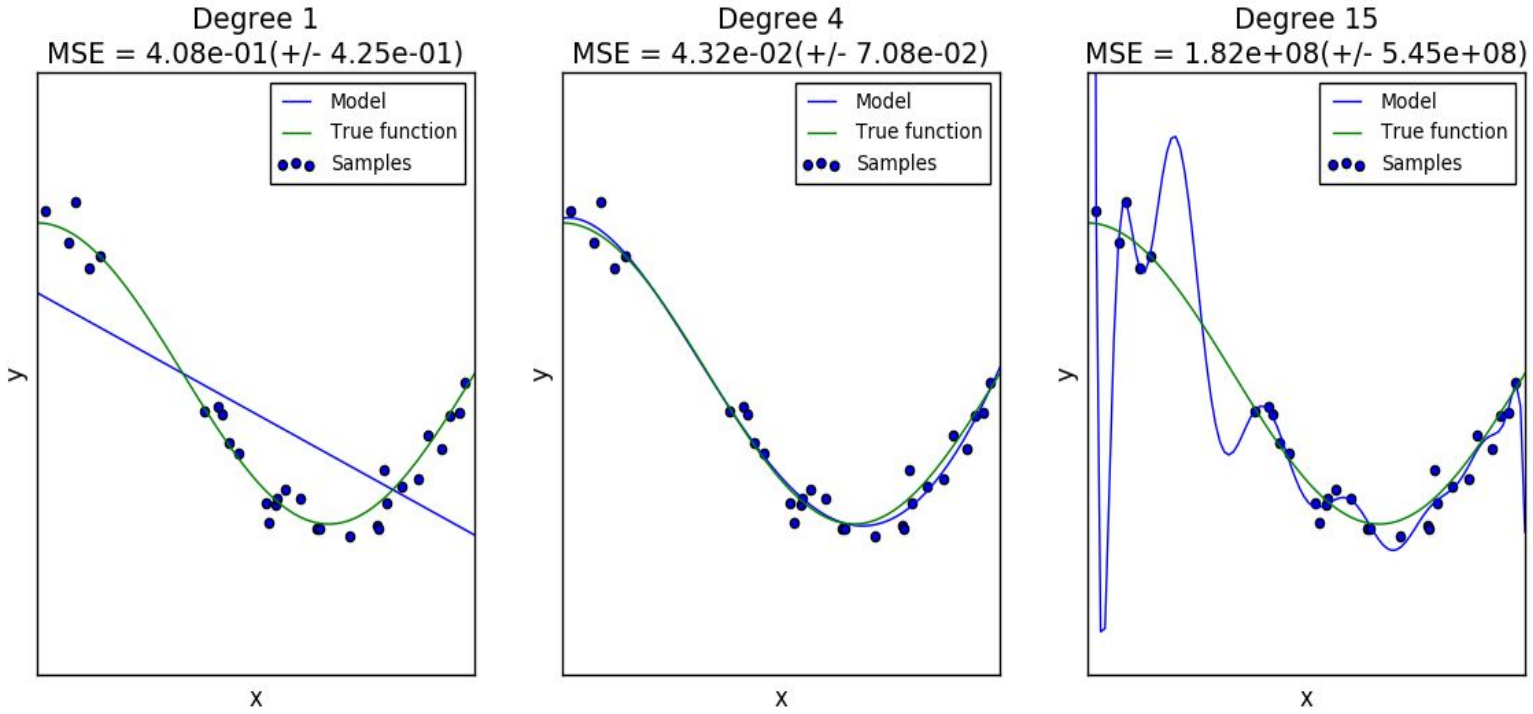
The aim of the algorithm is to minimize as much as possible this loss (iteratively!) using the training data.

The performance of the algorithm can’t be measured with the same training data it was fitted with!

Performance metrics



Overfitting and Underfitting



Model Validation

- The performance of the algorithm can't be measured with the same training data it was fitted with!
- To prevent underfitting and overfitting, the dataset needs to be splitted into (commonly three partitions) - **training set, cross-validation set and test set**
- The model will be : fitted with the training set

  Evaluated with the cross-val. set

  Finally deployed on the test set

The Scikit Learn Machine learning Library