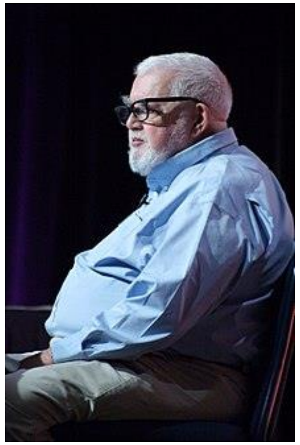
AI and machine learning



John McCarthy

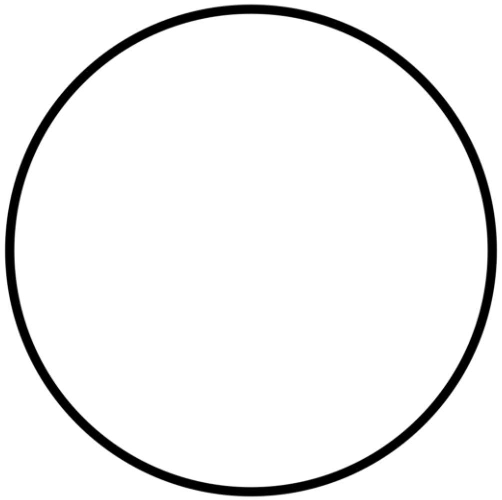
AI was introduced as an academic discipline in 1950s to create machines that can think.

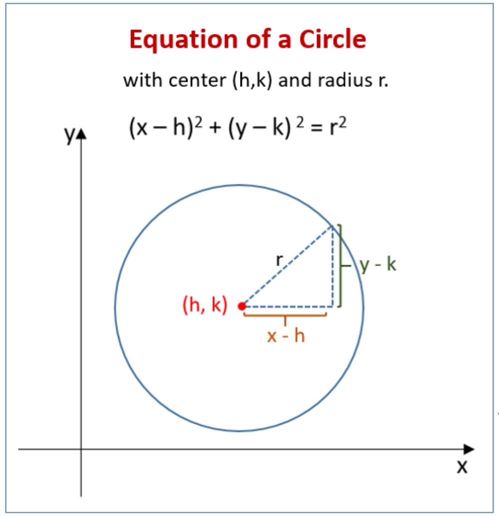
John McCarthy, widely recognized as one of the godfathers of AI, defined it as the science and engineering of making intelligent machines.

Old-Fashioned AI were mainly based on rules. (eg. Chess-playing system).

Machine learning is a subset of AI. That is, all machine learning counts as AI, but not all AI counts as machine learning.

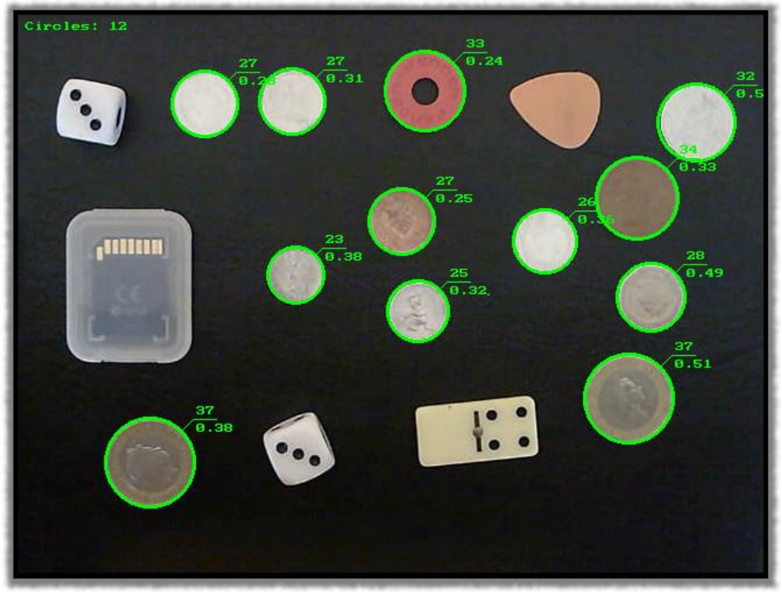
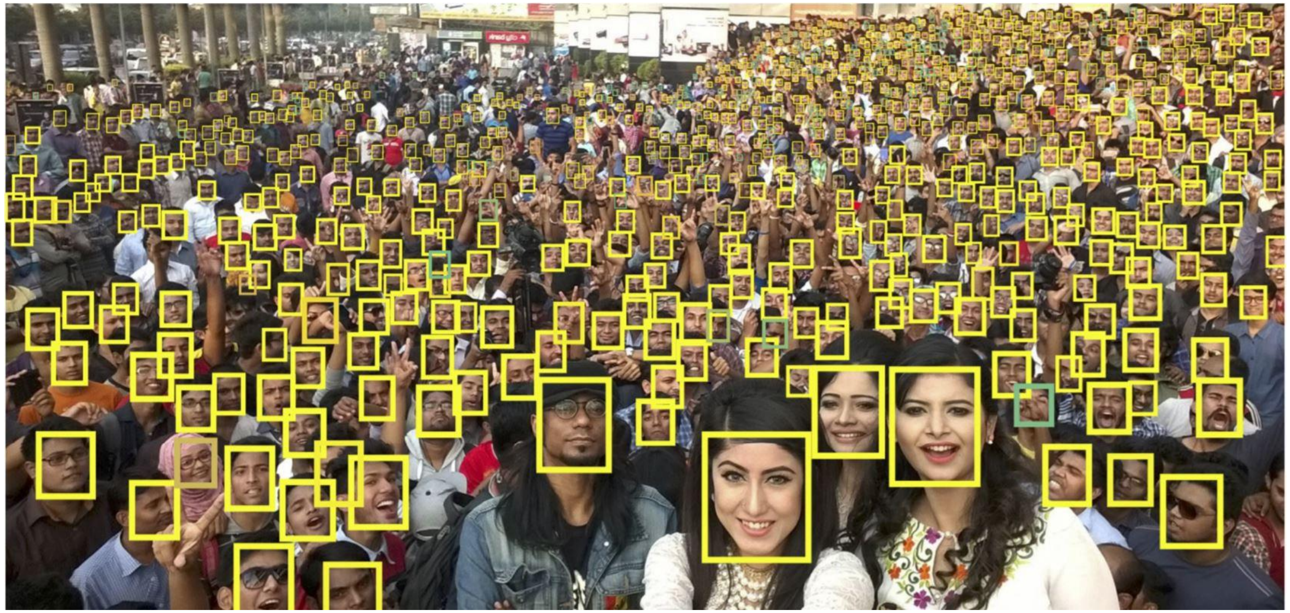
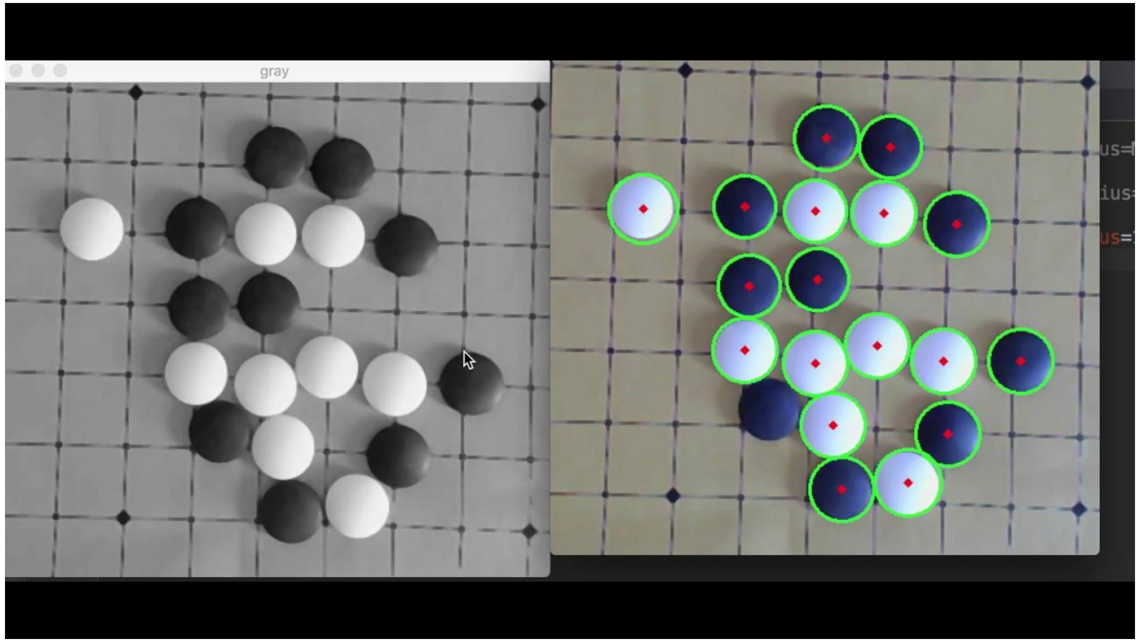
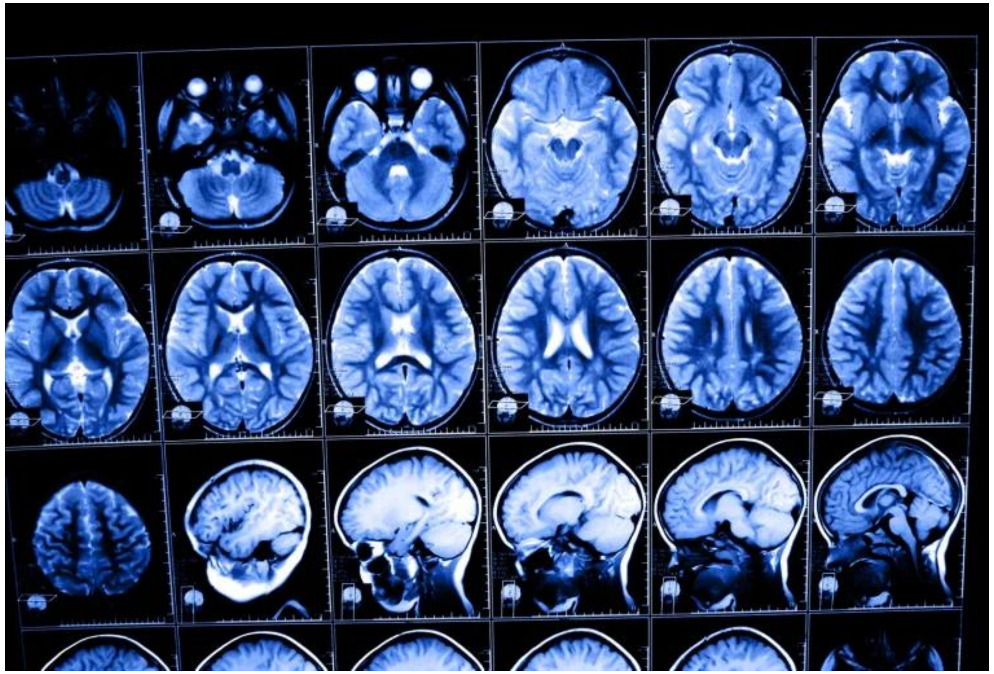
Definition of tree ?





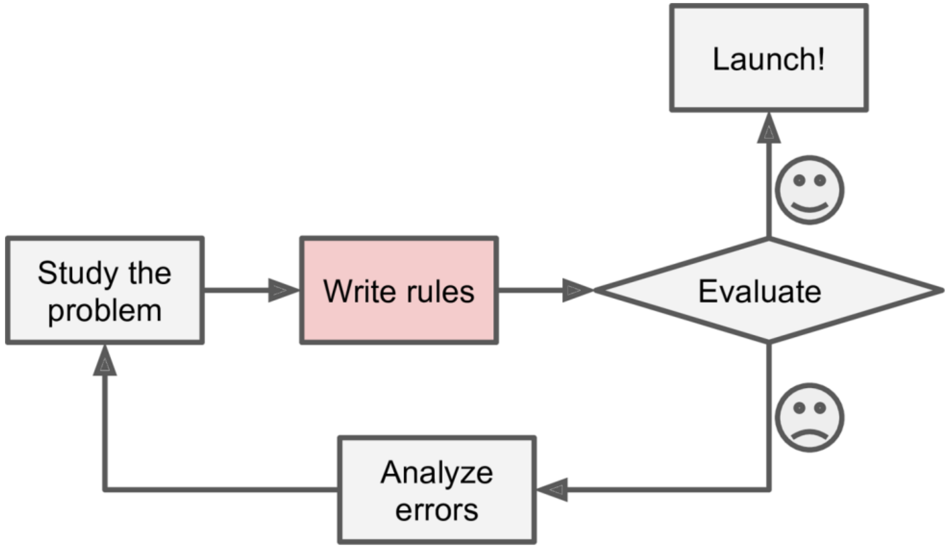
Equation of a Tree ?

We learned trees by looking at trees, not by studying its mathematical definition. In other words, we learned from data

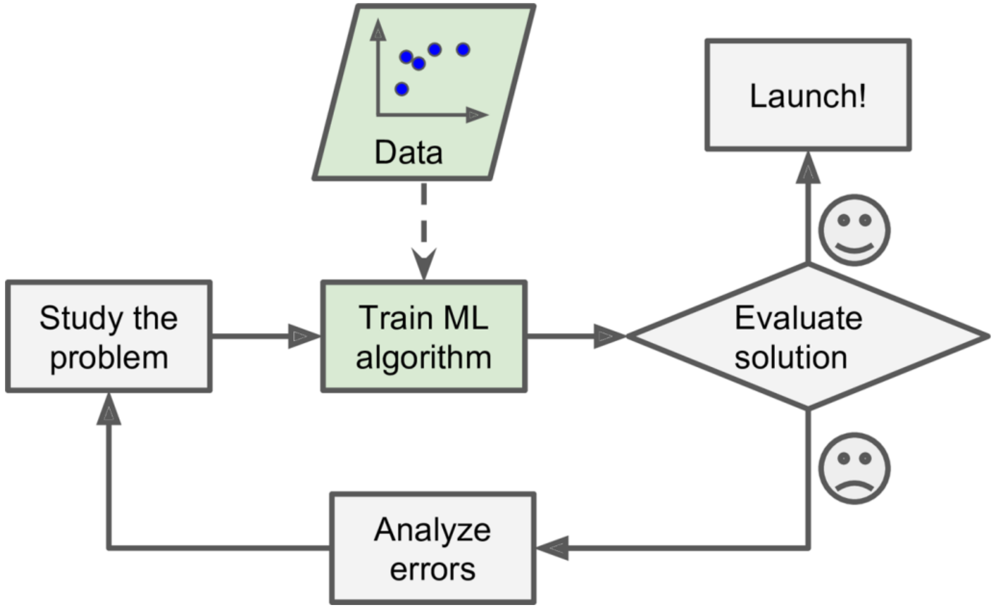

Circles: 12
27
0.31
33
0.24
32
34
0.33
27
0.25
23
0.38
25
28
0.49
37
0.51
37
0.38


gray


Machine learning methods learns from data.

- Learning from data is used in situations where we don't have an analytic solution, but we do have data that we can use to construct an empirical solution.
- Learning from data is one of the most widely used techniques in science, engineering, and economics, among other fields.
- Machine learning is the science and art of programming computers so they can learn from data.



Traditional Approach



Machine Learning Approach

https://www.oreilly.com/library/view/hands-on-machine-learning/9781491962282/ch01.html

Learning versus Design

While learning is based on data, design approaches are based on specifications.

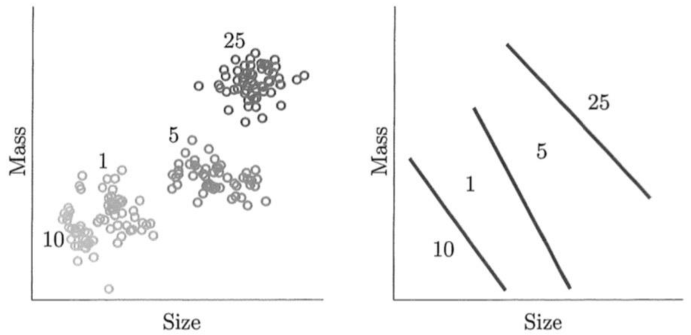
Example:-

Consider the problem of recognizing coins of different denominations, which is relevant to vending machines. We want the machine to recognize 50ps, 1 rupee, 2 rupee and 5 rupee.

In learning approach, we are given a sample of coins from each of the four denominations and we use these coins as our dataset. We treat the size and mass as the input vector, and the denomination as the output.

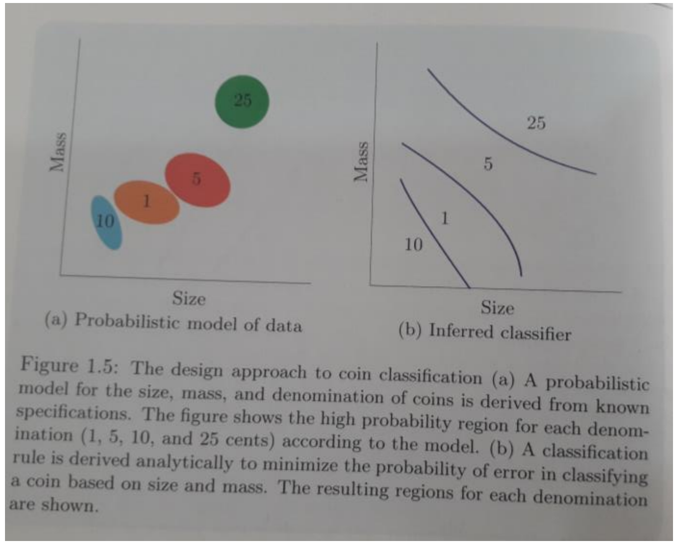
The learning algorithm searches for a hypothesis that classifies the data set well. If we want to classify a new coin, the machine measures its size and mass, and then classifies it according to the learned hypothesis.

Design approach : Compute the joint probability distribution of size mass and coin denomination. Once we have that joint distribution, we can construct the optimal decision rule to classify coins based on size and mass.



(a) Coin data (b) Learned classifier

Figure 1.4: The learning approach to coin classification (a) Training data of pennies, nickels, dimes, and quarters (1, 5, 10, and 25 cents) are represented in a size-mass space where they fall into clusters. (b) A classification rule is learned from the data set by separating the four clusters. A new coin will be classified according to the region in the size-mass plane that it falls into.

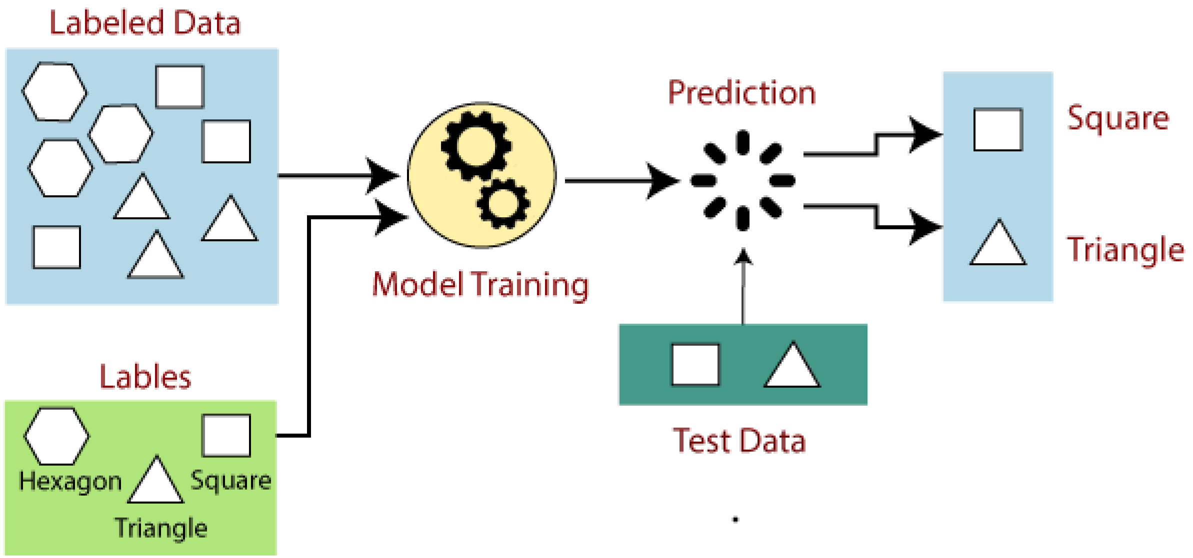


Types of Learning

- **Supervised learning** is the machine learning task of learning a function that maps an input to an output based on example input-output pairs.

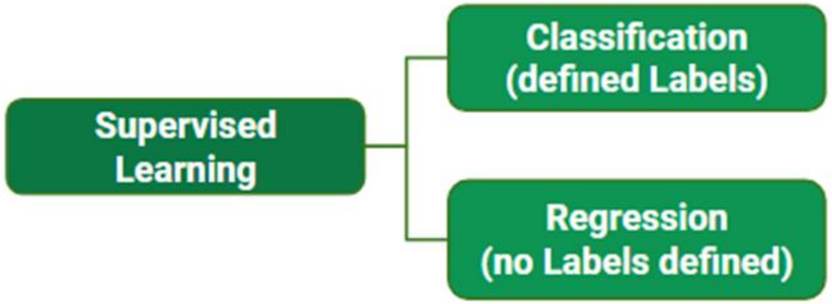
- It infers a function from *labeled* [*training data*]() consisting of a set of *training examples*.

- In supervised learning, each example is a *pair* consisting of an input object (typically a vector) and a desired output value (also called the *supervisory signal*).



Some important supervised learning methods

- Linear regression
- Logistic regression
- Support Vector Machines (SVM)
- Decision Trees and Random Forests
- Neural Networks

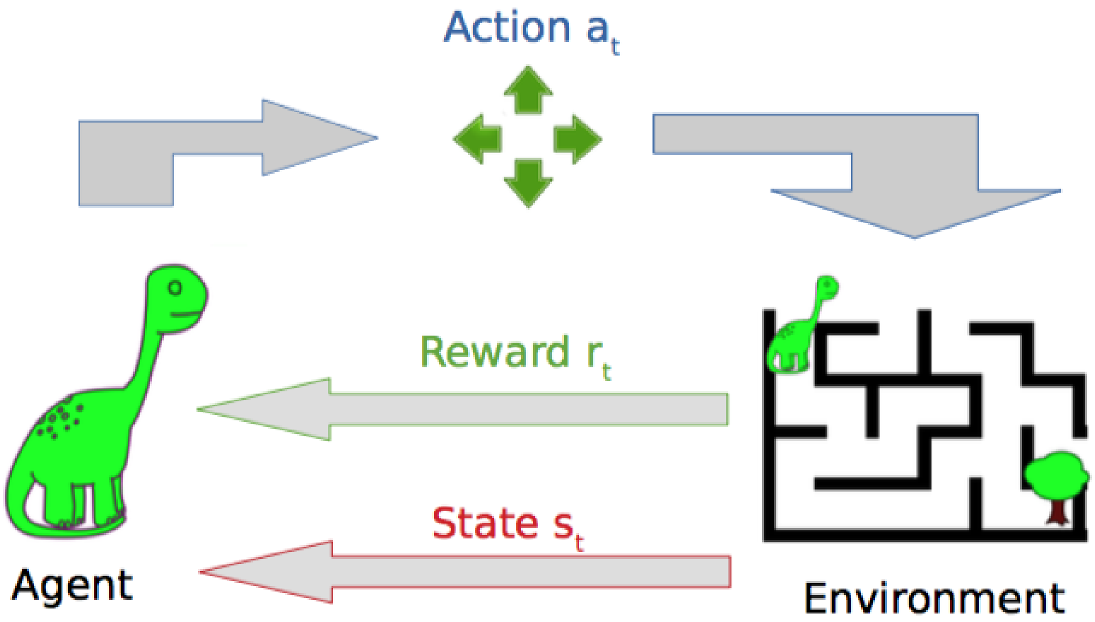


Reinforcement Learning

- When the training data does not explicitly contain the correct output for each input, we are no longer in a supervised setting.

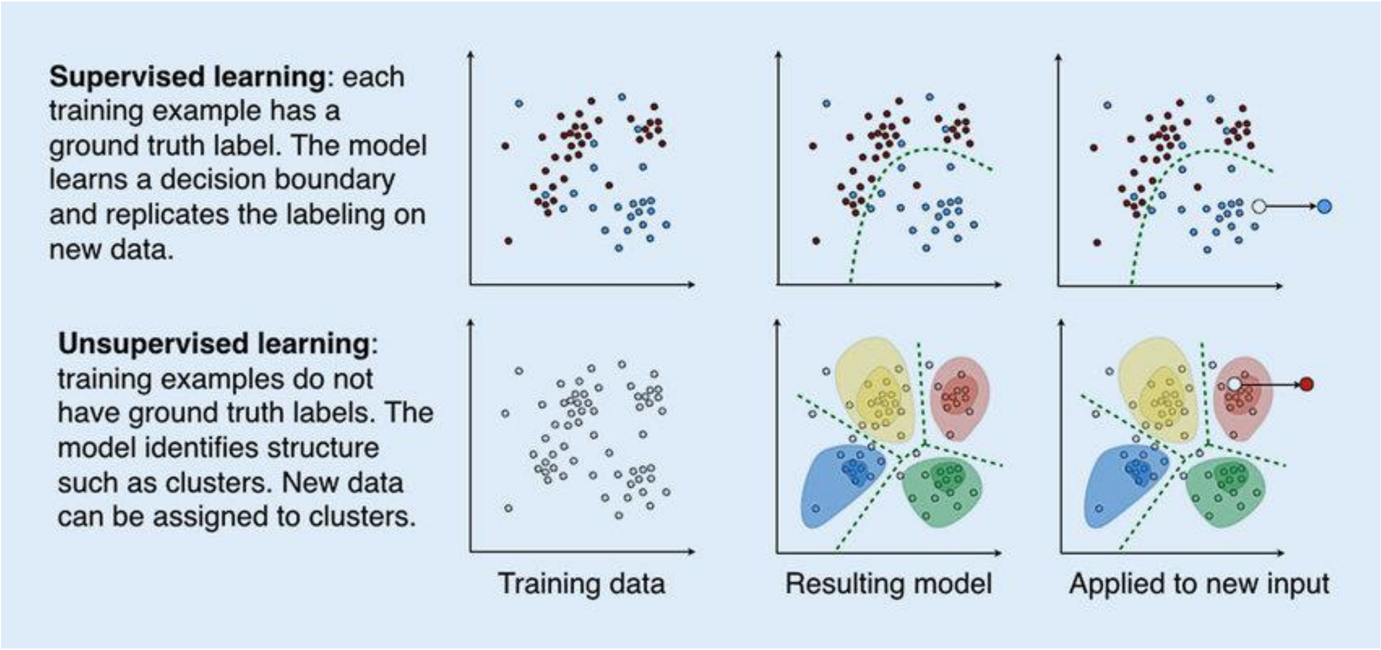
- In reinforcement learning, the training example does not contain the target output, but instead contains some possible output together with a measure of how good that output is.

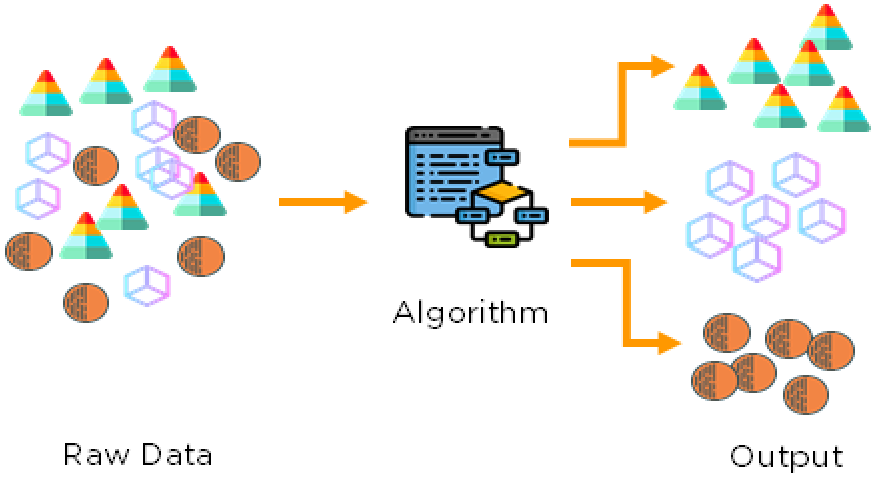
- In contrast to supervised learning where the training examples were of the form (input, correct output), the examples in reinforcement learning are of the form (input, some output, grade for this output)



Unsupervised Learning

- In the unsupervised setting, the training data does not contain any output information at all.
- In contrast to supervised learning, it's not always easy to come up with metrics for how well an unsupervised learning algorithm is doing.





Some important unsupervised learning methods

- Clustering
- Visualization and dimensionality reduction

Main challenges of machine learning

Insufficient quantity of training data

- Many machine learning algorithms require large amounts of data before they begin to give useful results.
- Even for very simple problems you typically need thousands of examples, and for complex problems such as image or speech recognition you may need millions of examples (unless you can reuse parts of an existing model).

The Unreasonable Effectiveness of Data

In a famous paper published in 2001, Microsoft researchers Michele Banko and Eric Brill showed that very different Machine Learning algorithms, including fairly simple ones, performed almost identically well on a complex problem of natural language disambiguation[8] once they were given enough data (as you can see in Figure 1-20).

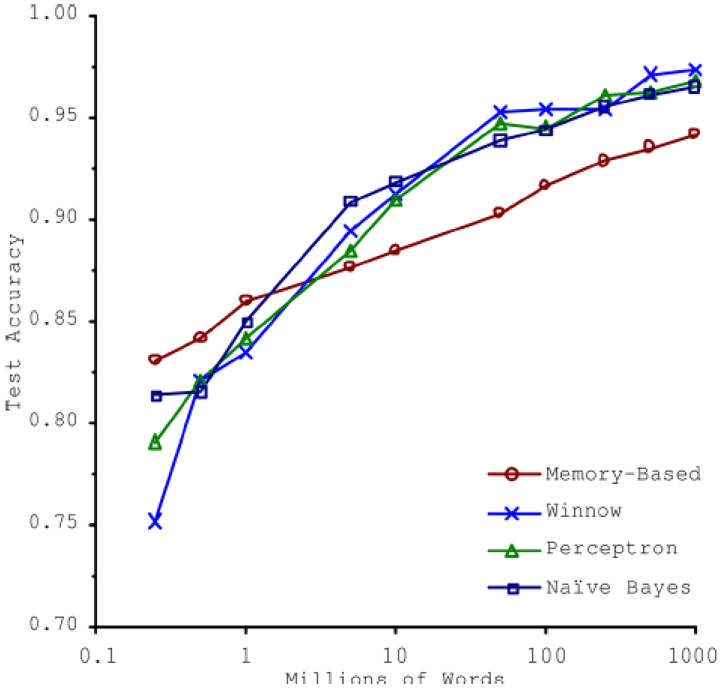


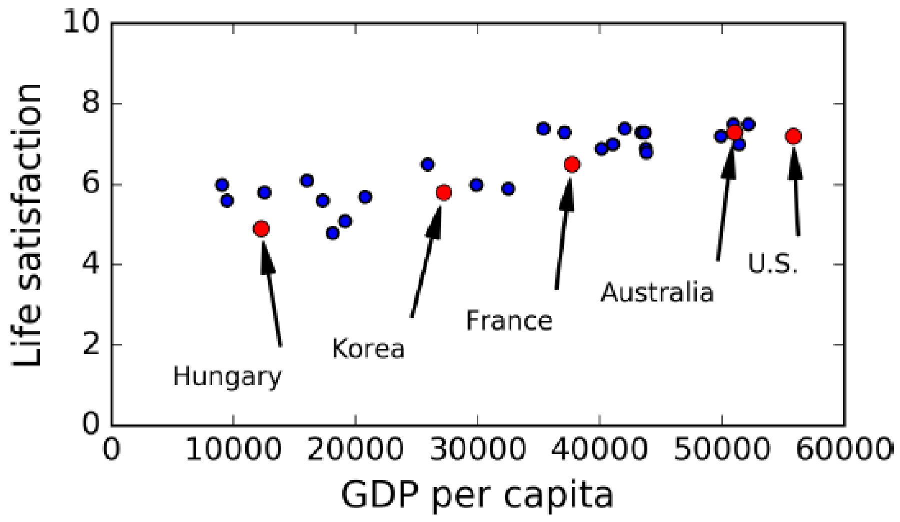
Figure 1-20. The importance of data versus algorithms[9]

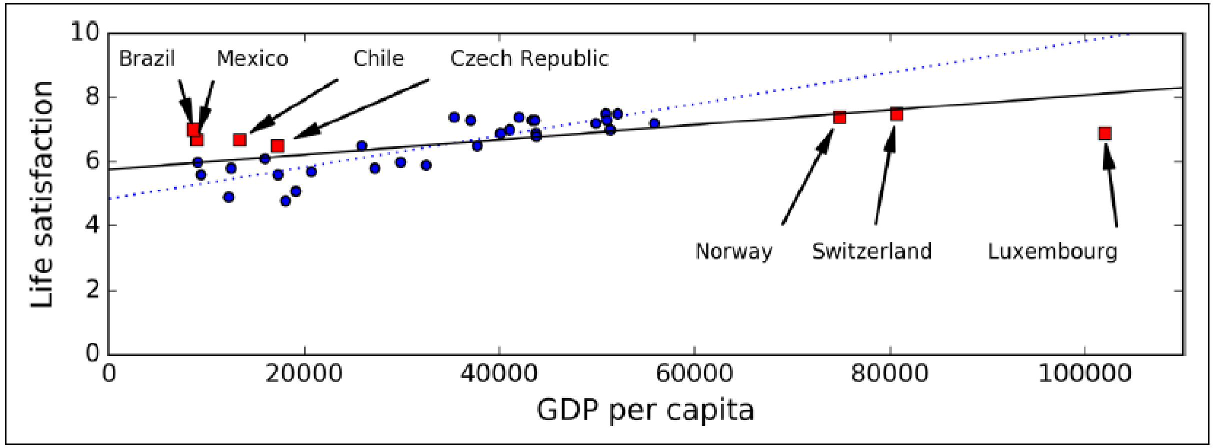
Nonrepresentative Training data

- In order to generalize well, it is crucial that your training data be representative of the new cases you want to generalize to.

- It is crucial to use a training set that is representative of the cases you want to generalize to.

- If the sample is too small, you will have *sampling noise* (i.e., nonrepresentative data as a result of chance), but even very large samples can be nonrepresentative if the sampling method is flawed. This is called *sampling bias*.



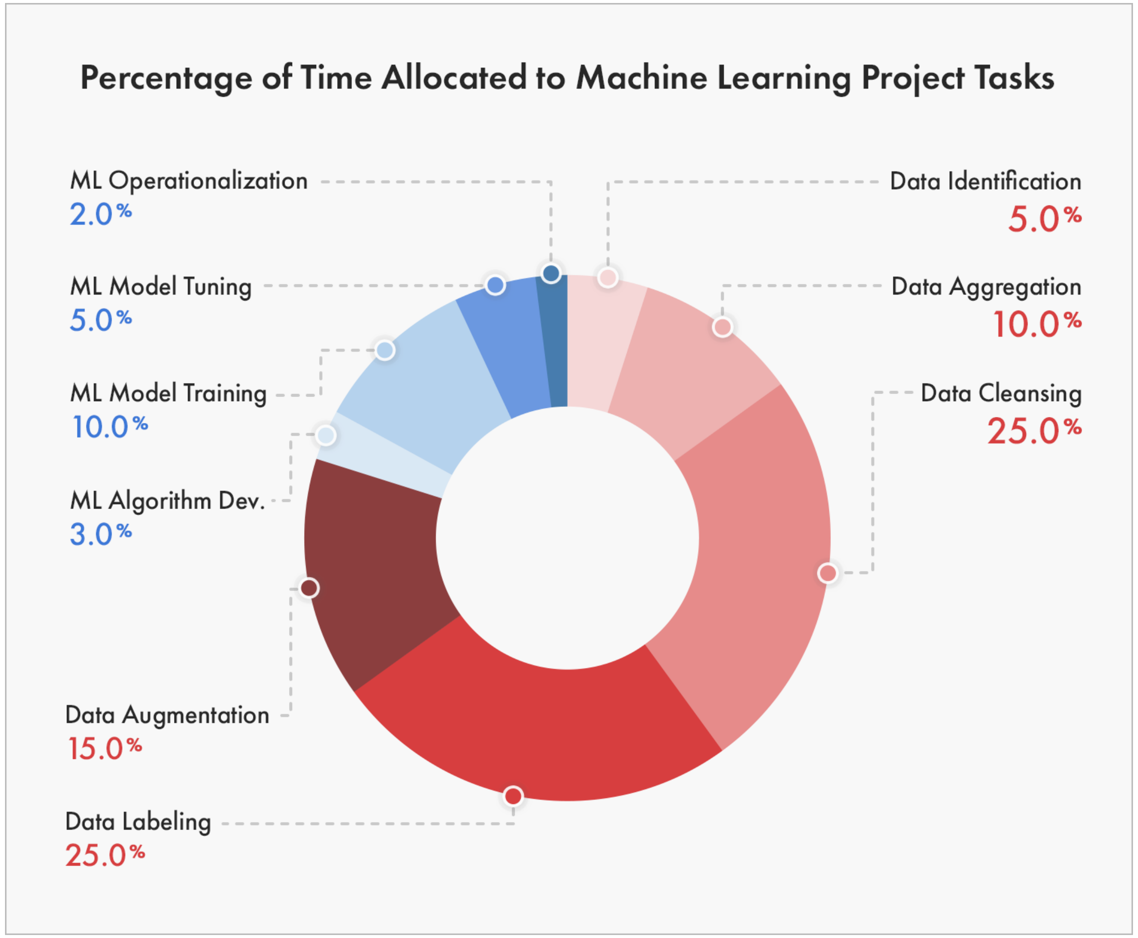


Poor-Quality data

- If your training data is full of errors, outliers, and noise (e.g., due to poor quality measurements), it will make it harder for the system to detect the underlying patterns, so your system is less likely to perform well.

- Clean the data before using it for training/testing

- If some instances are clearly outliers, it may help to simply discard them or try to fix the errors manually.

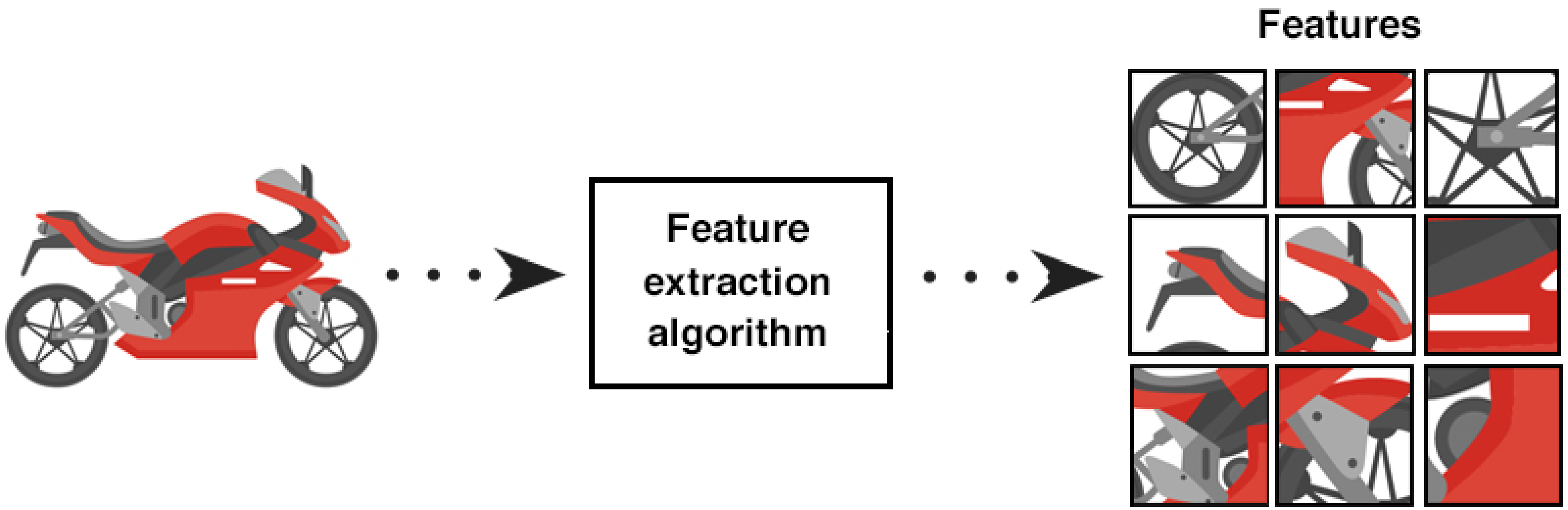


Irrelevant Features

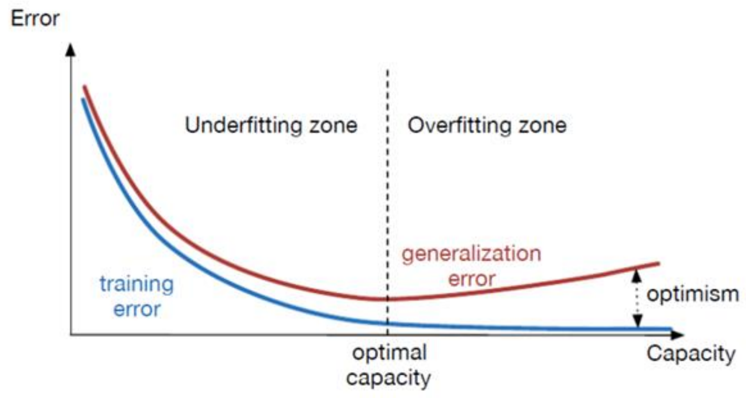
A critical part of the success of a Machine Learning project is coming up with a good set of features to train on.

This process, called *feature engineering*, involves:

- *Feature selection*: selecting the most useful features to train on among existing features.
- *Feature extraction*: combining existing features to produce a more useful one (as we saw earlier, dimensionality reduction algorithms can help).
- Creating new features by gathering new data

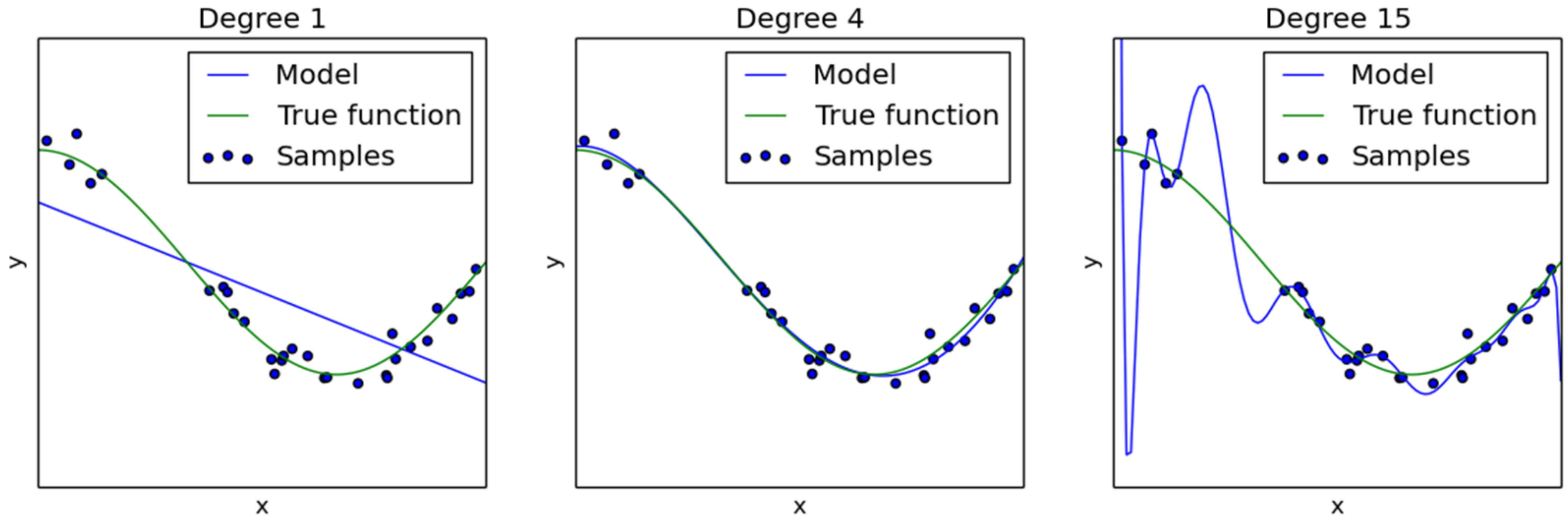


Overfitting and underfitting

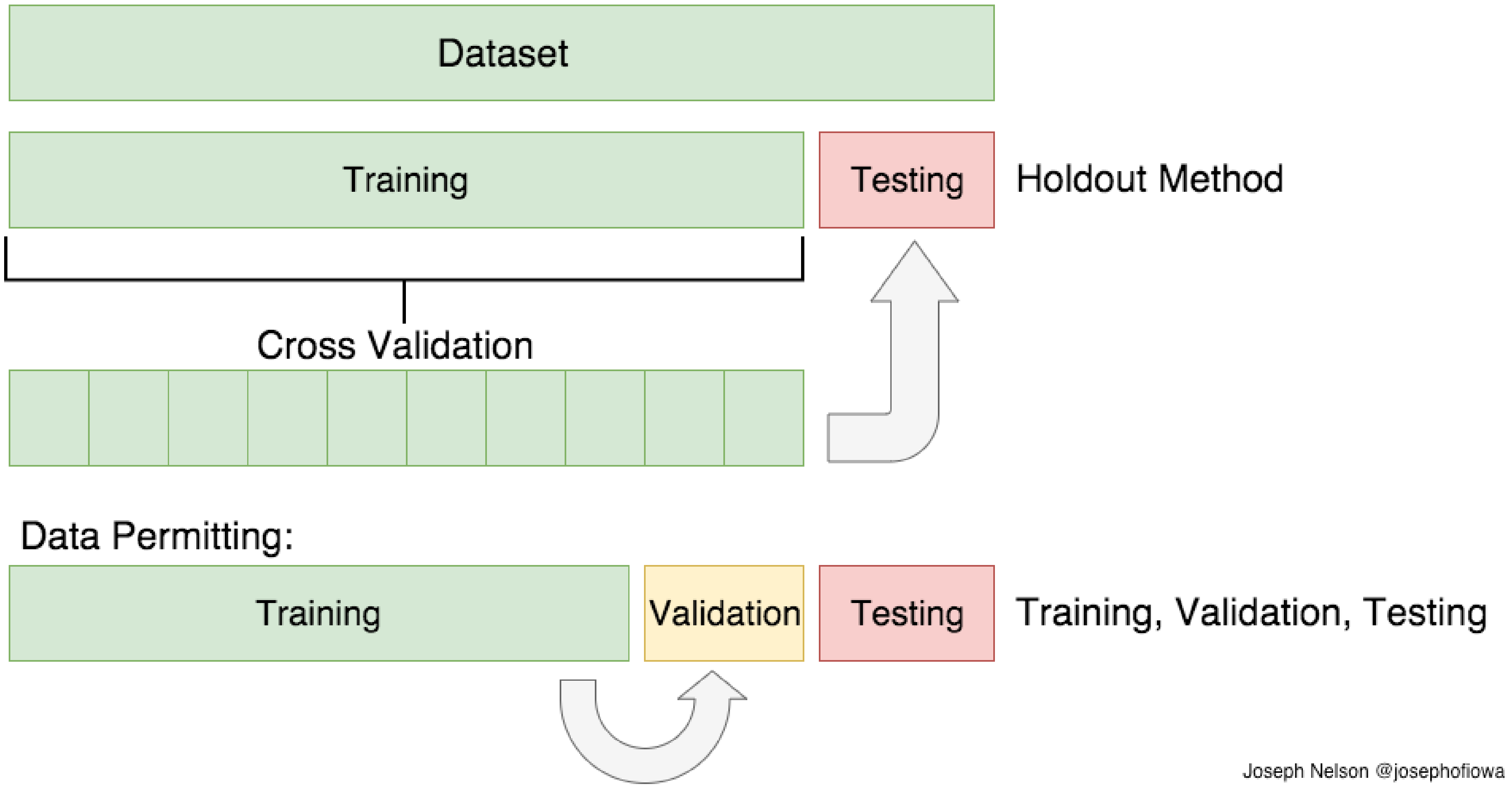


Overfitting : The model performs well on the training data, but does not generalize well.

Underfitting : The model is too simple to learn the underlying structure of the data.



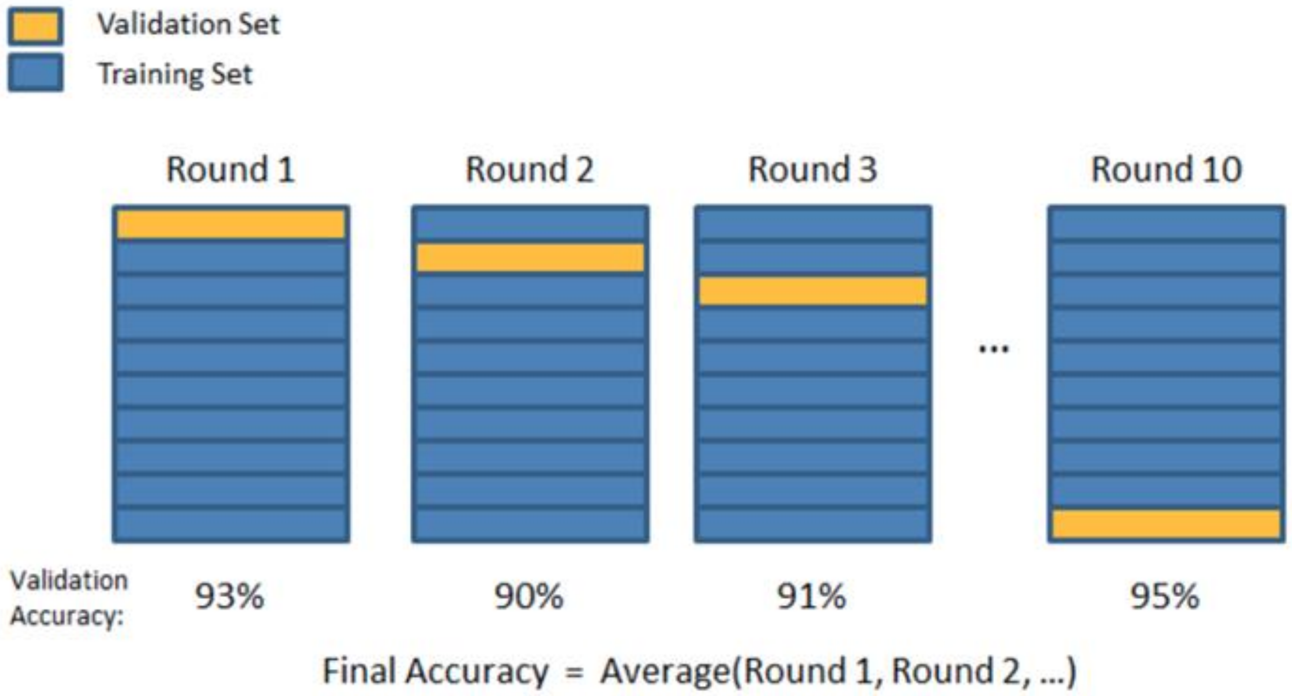
Training, Validation and Test data



https://towardsdatascience.com/train-test-split-and-cross-validation-in-python-80b61beca4b6

K-fold cross validation

- In K-Folds Cross Validation we split our data into k different subsets (or folds).
- We use k-1 subsets to train our data and leave the last subset (or the last fold) as test data.
- We then average the model against each of the folds and then finalize our model. After that we test it against the test set.



https://towardsdatascience.com/train-test-split-and-cross-validation-in-python-80b61beca4b6