Machine Learning & Predictive Analytics

Class 1

Arnab Bose, Ph.D.

MSc Analytics

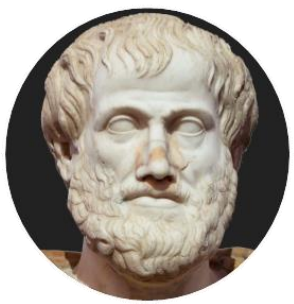
University of Chicago

Artificial Intelligence Genesis

Aristotle
Epagoge
Theory of Induction

understand causation

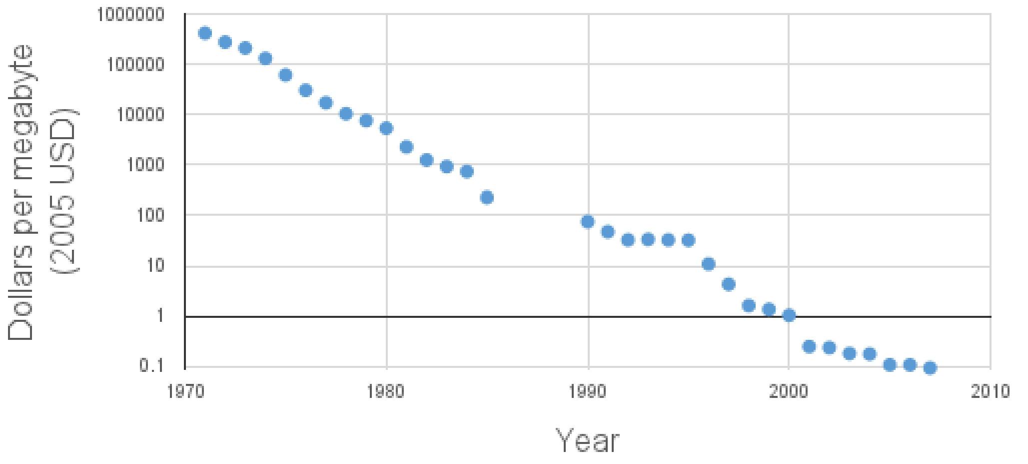
AI
Data Ocean
Disk storage
Processing power



Circa 350 BC

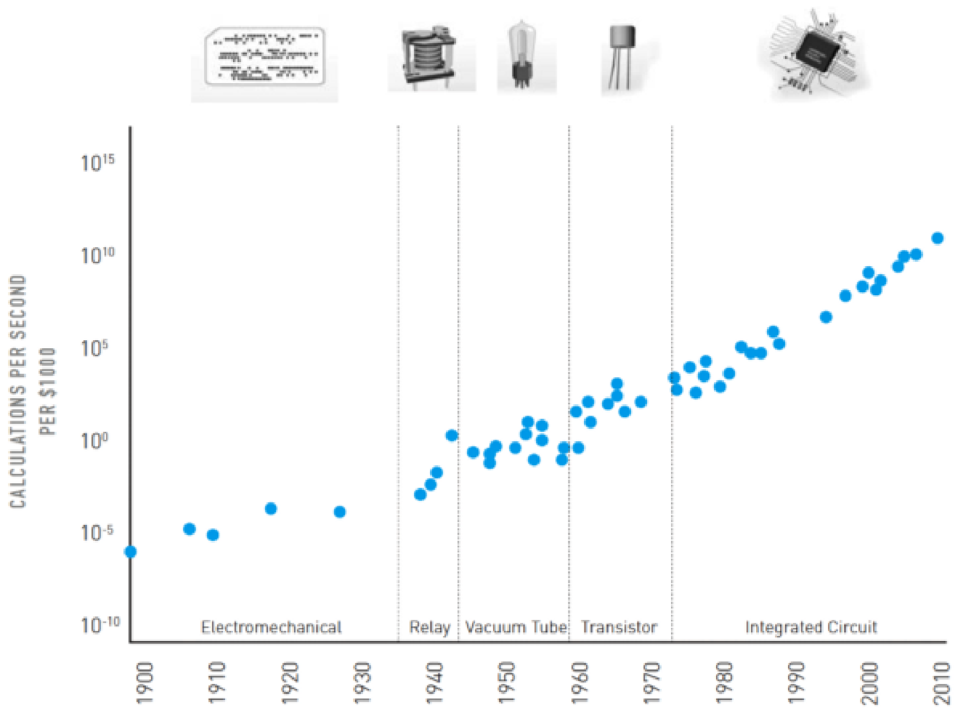
Modern Day

Data Storage



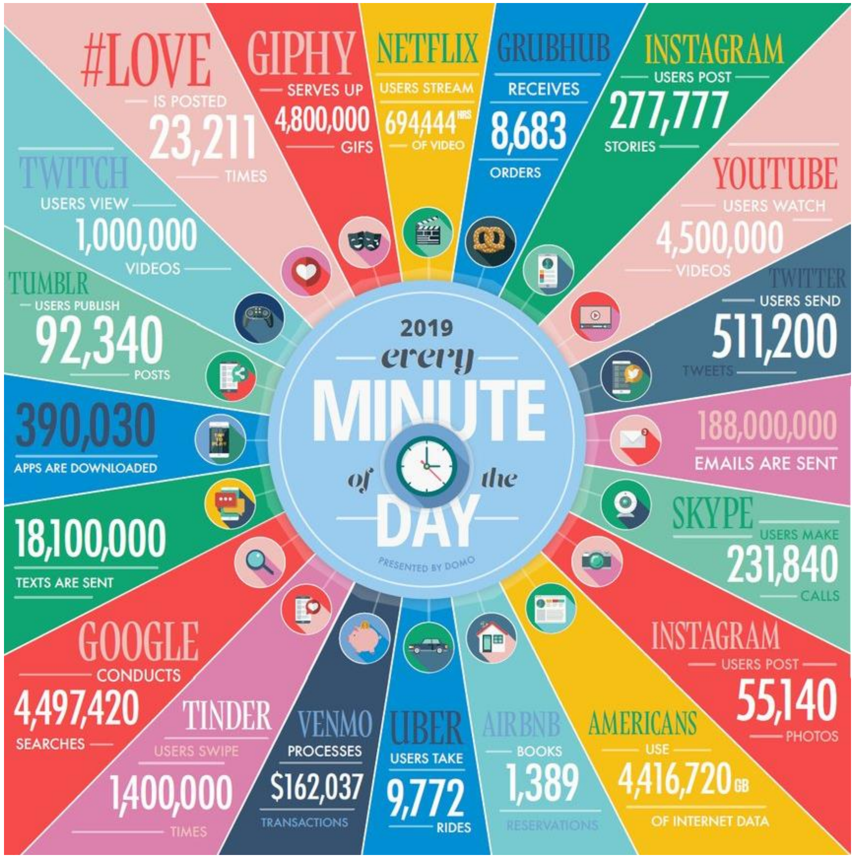
https://intelligence.org/2014/05/12/exponential-and-non-exponential/dram-chart-2/

Processing Power



https://medium.com/@nivo0o0/when-exponential-technological-progress-becomes-our-reality-74acafd65e26

Data Ocean



https://twitter.com/telefonicab2b/status/1153988229672124417

Artificial Intelligence (AI)

- Human Intelligence Exhibited by Machines

You've seen these machines endlessly in movies —
- As friend — C-3PO
- As foe — The Terminator.

General AI machines have remained in the movies and science fiction novels for good reason; we can't pull it off, at least not yet.

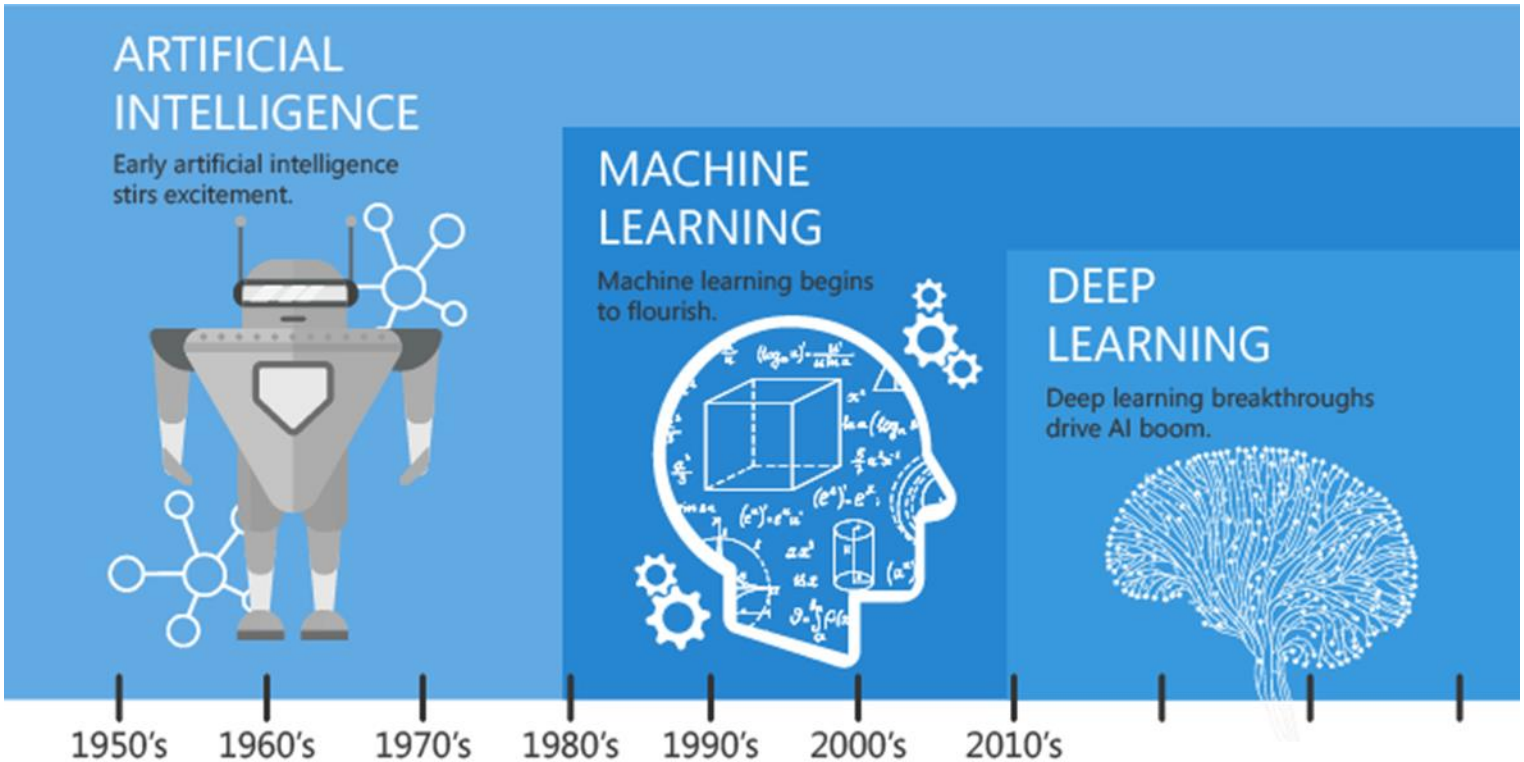
What we can do falls into the concept of "Narrow AI."

Technologies that are able to perform specific tasks as well as, or better than, we humans can.

Examples of narrow AI are things such as:

- Image classification on a service like Pinterest
- Face recognition on Facebook.

Artificial Intelligence (AI) / Machine Learning (ML) / Deep Learning (DL)



Since an early flush of optimism in the 1950's, smaller subsets of artificial intelligence - first machine learning, then deep learning, a subset of machine learning - have created ever larger disruptions.

AI Systems – Expert vs Learning

1. Expert systems rely on hardcoded knowledge
    1. Inspired by logic systems
    2. Difficulty in handling unexpected/unseen patterns

2. Learning systems learn from first principles using raw data
    1. Inspired by neuroscience
    2. Can be used for new unseen tasks

What is Machine Learning

1. Meaningful data transformations from input to output data.

2. Transformations: represent or encode the data (RGB or HSV for color pixel).

3. Learning is automatic search for better data representations.

4. Search through a predefined space of possibilities using guidance from feedback signal.

5. *A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks T, as measured by P, improves with experience E*

   - Tom Mitchell, Machine Learning, McGraw Hill, 1997

Experience E, Task T, Performance P

1. Chess

   T: playing chess

   P: % of games won

   E: playing practice games against itself.

2. Driving

   T: driving a vehicle

   P: avg distance before error

   E: sequence of images and steering commands recoded during manual driving.

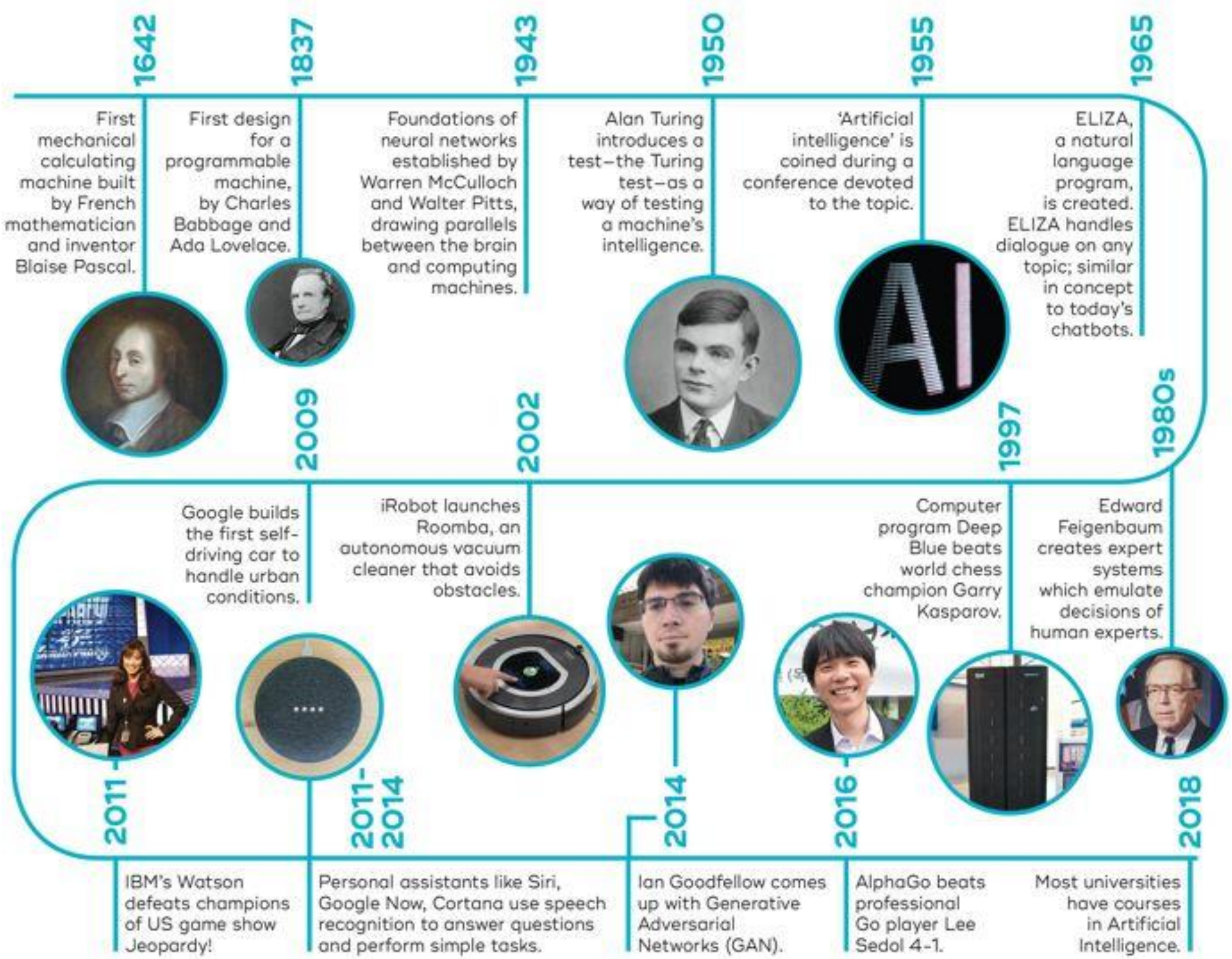
3. Handwriting Recognition

   T: recognizing and classifying handwritten words in images

   P: % of correctly classified words

   E: DB of handwritten words with given classifications.

History of Machine Learning



http://tinkeringchild.com/ideas-for-exploring-machine-learning-in-the-primary-years/

Types of Learning – Part 1

Supervised
- Linear Regression
- Logistic Regression
- K-NN (k-Nearest Neighbors)
- Support Vector Machines (SVMs)
- Decision Tress and Random Forests
- Neural Networks

Unsupervised
- Clustering
- Visualization and Dimensionality Reduction
- Association Rule
- Neural Networks

Semi-Supervised
- Document classifier
- Recommender system

Self-Supervised
- Word embeddings
- Autoencoders

Supervised Learning

1. The majority of practical machine learning uses supervised learning.

2. Supervised learning is where you have input variables (x) and an output variable (Y) and you use an algorithm to learn the mapping function from the input to the output.

$$y = f(x)$$

3. The goal is to approximate the mapping function so well that when you have new input data x that you can predict the output variables y for that data.

4. It is called supervised learning because the process of an algorithm learning from the training dataset can be thought of as a teacher supervising the learning process.

5. We know the correct answers, the algorithm iteratively makes predictions on the training data and is corrected by the teacher. Learning stops when the algorithm achieves an acceptable level of performance.

Unsupervised Learning

1. Unsupervised learning is where you only have input data x and no corresponding output variables.
2. The goal for unsupervised learning is to model the underlying structure or distribution in the data in order to learn more about the data.
3. These are called unsupervised learning because unlike supervised learning above there is no correct answers and there is no teacher. Algorithms are left to their own devises to discover and present the interesting structure in the data.
4. Unsupervised Learning problems can be further grouped into Clustering and Association Problems.
5. Clustering: A clustering problem is where you want to discover the inherent groupings in the data, such as grouping customers by purchasing behavior.
6. Association: An association rule learning problem is where you want to discover rules that describe large portions of your data, such as people that buy A also tend to buy B.

Semi-supervised Learning

1. Semi-supervised learning is halfway between supervised and unsupervised learning.
2. Traditional classification methods use labelled data to build classifiers.
3. The labelled training sets used as input in Supervised learning is very certain and properly defined.
4. However, they are limited, expensive and takes a lot of time to generate them.
5. On the other hand, unlabeled data is cheap and is readily available in large volumes.
6. Hence, semi-supervised learning is learning from a combination of both labelled and unlabeled data;
7. Where we make use of a combination of small amount of labelled data and large amount of unlabeled data to increase the accuracy of our classifiers.

Self-supervised Learning

1. Form of unsupervised learning where the data provides the supervision.

2. Uses labeled training data.

3. The labeling is autonomous and no manual (human) labeling is needed.

4. Well suited for online learning.

Types of Learning – Part 2

ALGORITHMS

Reinforcement Learning

Transfer Learning

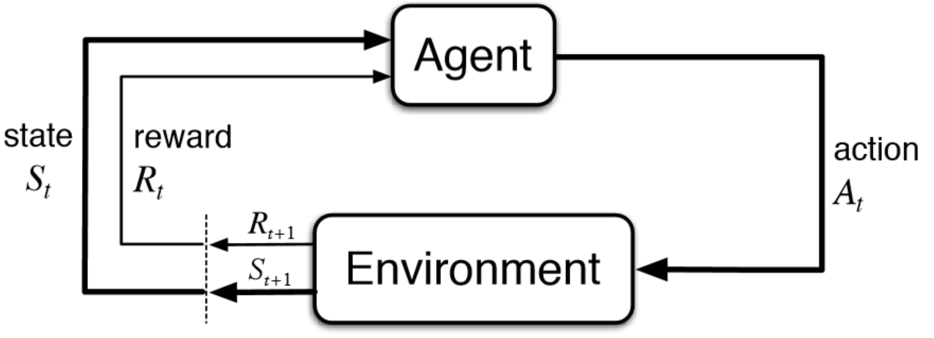
Active Learning

Reinforcement Learning

Reinforcement Learning is learning what to do and how to map situations to actions.

The end result is to maximize the numerical reward signal.

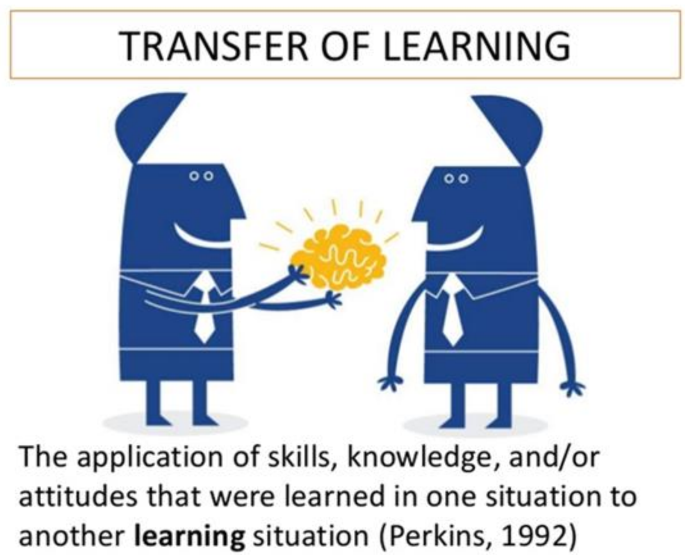
The learner is not told which action to take, but instead must discover which action will yield the maximum reward.



Transfer Learning

1. A machine learning technique where a model trained on one task is re-purposed on a second related task.

2. An optimization that allows rapid progress or improved performance when modeling the second task.

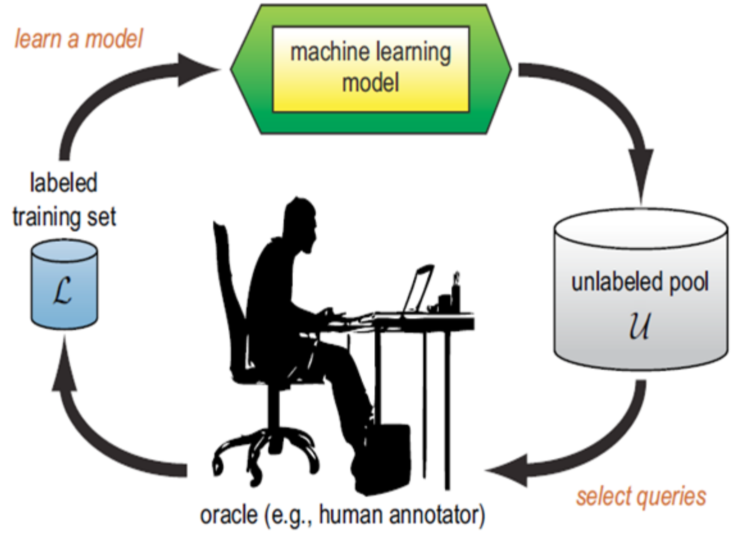


Active Learning

Active learning (sometimes called "query learning" or "optimal experimental design" in the statistics literature) is a subfield of machine learning and, more generally, artificial intelligence.

The key hypothesis is that if the learning algorithm is allowed to choose the data from which it learns—to be "curious," if you will—it will perform better with less training.

Active learning is a special case of semi-supervised learning.



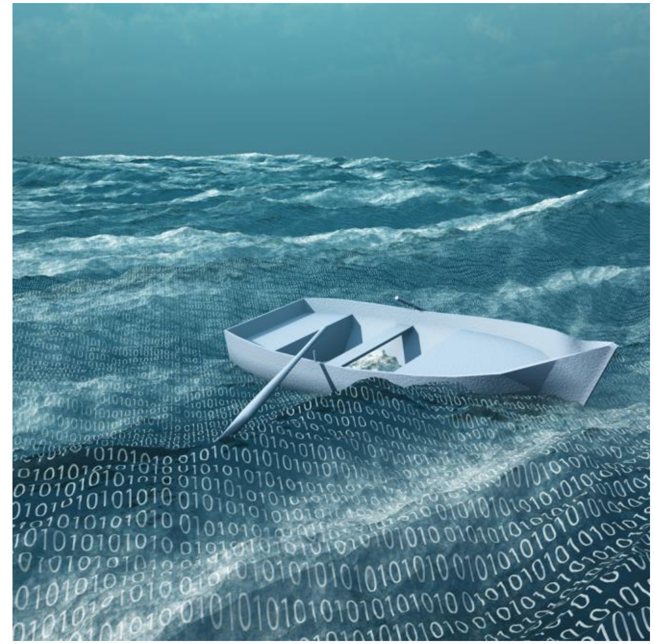
http://burrsettles.com/pub/settles.activelearning.pdf

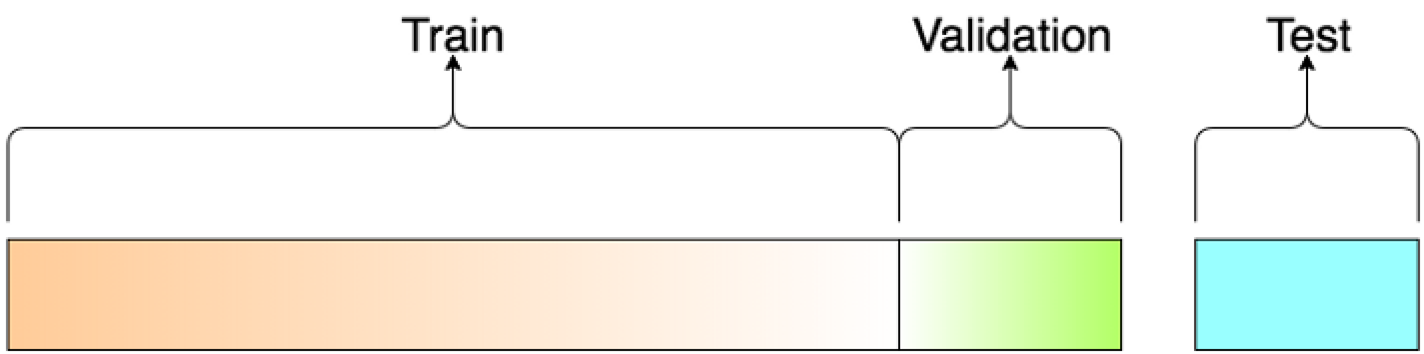
Navigate the Ocean of Data

1. Training – fit model parameters

2. Validation (aka development / dev)– tune model hyperparameters

3. Test – assess the performance of the model





https://towardsdatascience.com/train-validation-and-test-sets-72cb40cba9e7

Data Assumptions

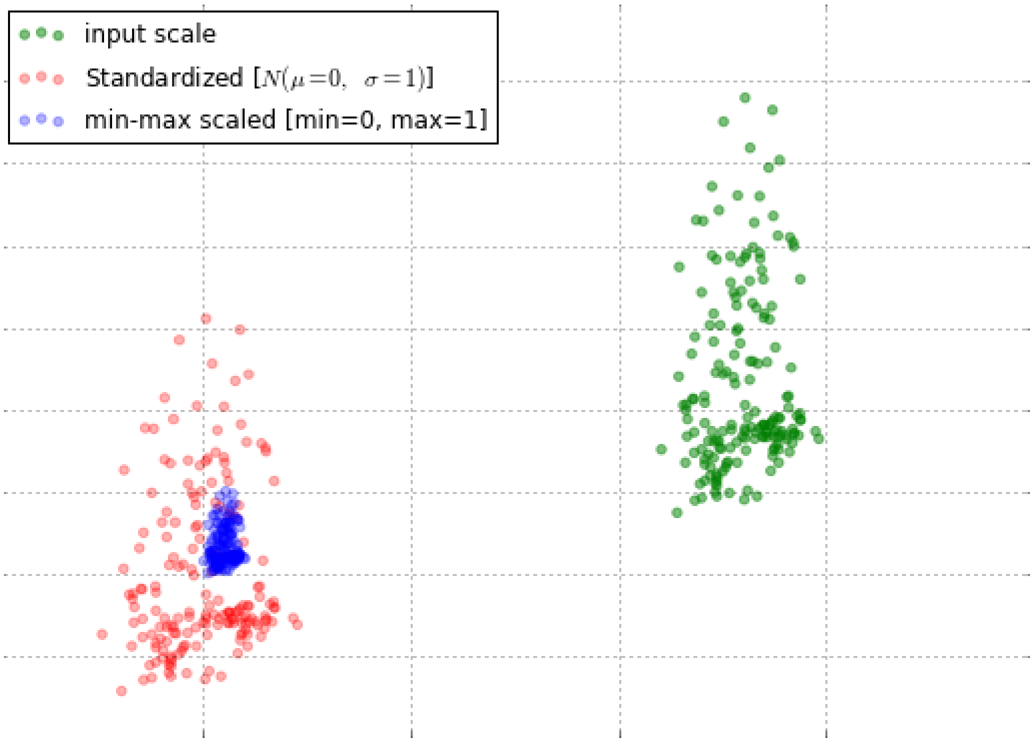
1. Training and test data are from the same probability distribution.

2. Training and test data are iid (independent and identically distributed).

Normalization vs Standardization

$$S = \frac{X - X_{mean}}{\sigma_X}$$

$$N = \frac{X - X_{min}}{X_{max} - X_{min}}$$



https://stackoverflow.com/questions/32108179/linear-regression-normalization-vs-standardization

http://www.faqs.org/faqs/ai-faq/neural-nets/part2/section-16.html

Universal Workflow of ML

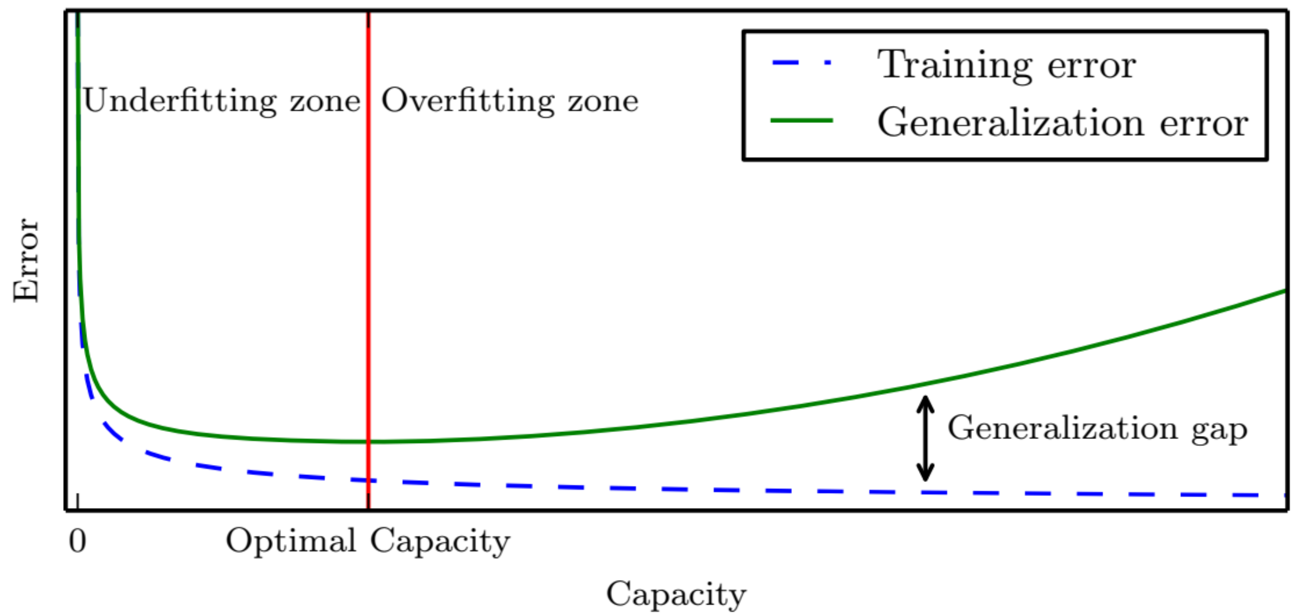
1. Define the problem
2. Assemble dataset
3. Choose a metric to quantify project outcome
4. Decide on how to calculate the metric
5. Prepare dataset
6. Define standard baseline
7. Develop model that beats baseline
8. Ideal model is at the border of overfit and underfit – cross the border to know where it is so overfit model
9. Regularize model and tune hyperparameters

Overfitting and Underfitting

1. Overfitting – model fits very well to the training data, aka detects patterns in the noise also
    1. Detect:
        1. Low training error, high generalization error.
    2. Remedies:
        1. Reduce model capacity by removing features and/or parameters.
        2. Get more training data.
        3. Improve training data quality by reducing noise.

2. Underfitting – model too simple to detect patterns in the data
    1. Detect
        1. High training error.
    2. Remedies:
        1. Increase model capacity by adding more parameters and/or features.
        2. Reduce model constraints.

Error vs Model Capacity



http://www.deeplearningbook.org/lecture_slides.html

Parametric & Nonparametric Models

$$\hat{y} = \hat{f}(x)$$

Estimate the unknown function f as $\hat{f}$

- Parametric Models:
    1. Assume the functional form or shape of f
    2. Apply methodology to train model
    3. Advantage – simple estimation
    4. Disadvantage – $\hat{f}$ may be far from true f

- Nonparametric Models:
    1. No assumption on the functional form or shape of f
    2. Estimate to fit as close as possible to the data
    3. Advantage – can accurately fit a wide range of possible shapes of f
    4. Disadvantage – need large datasets (since there is no fixed # of params to estimate)

Linear Regression with OLS using scikit-learn

$$y = \theta^T X$$

The cost function minimization is a closed-form solution called the Normal Equation:

$$\hat{\theta} = (X^T . X)^{-1} X^T . y$$

- Advantage – equation is linear with size of training set so it can handle large training sets efficiently
- Disadvantage –
  1. computational complexity of inverting a matrix that increases with size of training set
  2. difficult to do online learning with new data arriving regularly (need to recalculate estimates), ie no iterative parameter updates

Textbook Chapters

- Materials covered available in books:
  - DL: Chapters 2 – 4
  - HML: Chapters 1, 4
  - ISL: Chapters 2 – 3
  - DLP: Chapter 4

- Code: https://github.com/ageron/handson-ml2