Introduction to Machine Learning and Artificial Intelligence

Yulia Newton, Ph.D.

CS156, Introduction to Artificial Intelligence

San Jose State University

Spring 2021

What is AI?

- “AI is the ability of a digital computer or computer-controlled robot to perform tasks commonly associated with intelligent beings.” - Britannica

- “AI is intelligence demonstrated by machines, unlike the natural intelligence displayed by humans and animals.” - Wikipedia

- “Artificial intelligence (AI) is wide-ranging branch of computer science concerned with building smart machines capable of performing tasks that typically require human intelligence. AI is an interdisciplinary science with multiple approaches, but advancements in machine learning and deep learning are creating a paradigm shift in virtually every sector of the tech industry.” - builtin.com

Automating intellectual tasks normally performed by humans

- What is an intellectual task?
- Can we make a computer “think” like a human?
- Can we make a computer “intelligent” like a human?

- “Artificial Intelligence” - term coined by John McCarthy in 1955 (1927 - 2011)
  - Influential contributor to the field of computer science, specifically AI, theory of computation and knowledge representation
  - Creator of LISP
  - https://history-computer.com/ModernComputer/Software/LISP.html

Introduction to history of AI

- Science fiction of early 20th century introduced the concept of intelligent machines
- By 1950's a generation of scientists, mathematicians, and engineers grew up familiar with the concept of artificial intelligence (machine intelligence)
- Alan Turing's "Computing Machinery and Intelligence" 1950 publication in Mind (Oxford Academic)
  - Introduced the concept of Turing Test
    - Can machines think?
  - Described how to build intelligent machines and how to test their intelligence
    - Machines can use available information and reasoning to solve problems, just like humans
  - https://academic.oup.com/mind/article/LIX/236/433/986238

Introduction to history of AI (cont'd)

- In mid 20th century computing was expensive
- Herbert Simon, Allen Newell and John Shaw designed the first AI program *Logic Theorist*
  - Mimic problem solving skills of a human
  - Funded by Research and Development (RAND) Corporation
  - Presented at Dartmouth Summer Research Project on Artificial Intelligence (DSRPAI) conference
  - https://history-computer.com/ModernComputer/Software/LogicTheorist.html

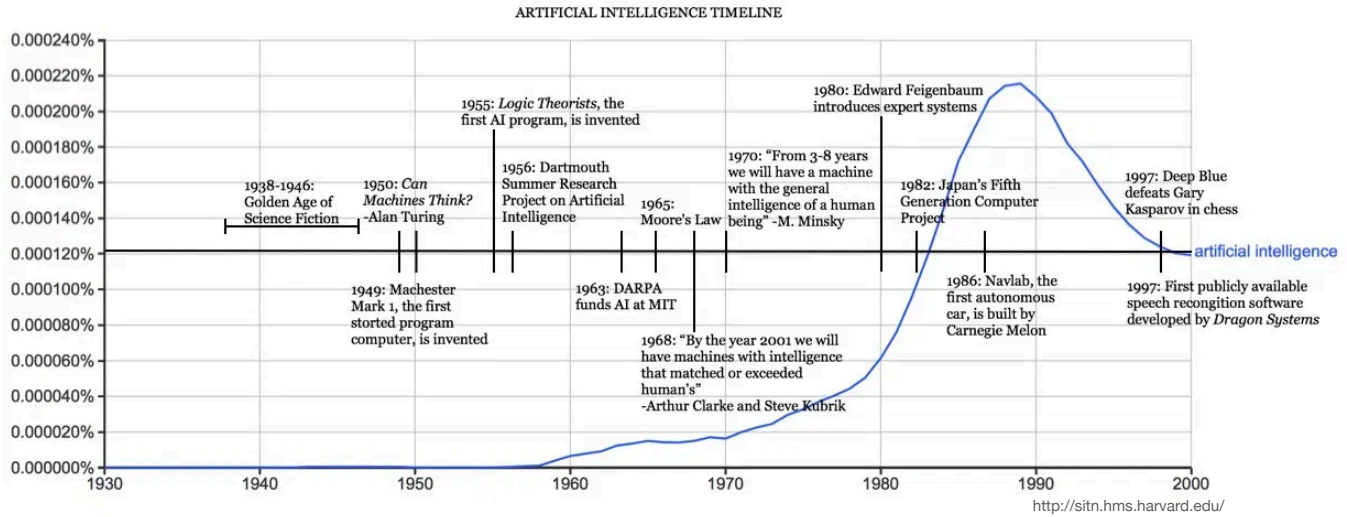
Introduction to history of AI (cont'd)

- Initially, AI was simply a pre-defined set of rules
  - Pre-defined and recorded by humans
  - The rules are hard-coded and the code uses those rules
    - No "learning" of new rules happening
  - Simulating human behavior - following an algorithm

https://en.wikipedia.org

Introduction to history of AI (cont'd)

- Rollercoaster of ups and downs for the field of AI over the years
- AI thrived in 1990's and 2000's
  - Expansion of algorithms, revolutionizing of computer processing and storage, Big Data
  - In 1997 IBM's Deep Blue beat grand master Gary Kasparov at chess and speech recognition software by Dragon Systems implemented on Windows



Examples of AI in everyday life

- AI is in all aspects of our life
  - Healthcare, banking, marketing, customer service, smart cars, smart homes, enterntainment, law enforcement, search and recommender systems, social media, e-commerce, etc.

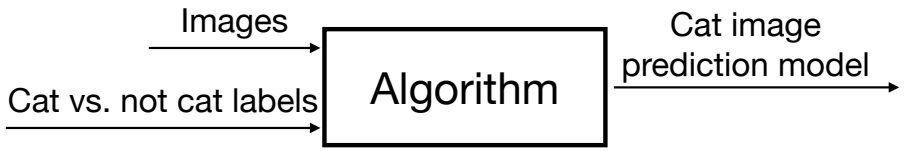
- Examples of AI applications
  - Voice recognition (Siri, Alexa, etc.)
  - Self-driving cars (Tesla, etc.)
  - Content-on-demand service (Netflix, Pandora, etc.)
  - Autocorrect, text suggestions
  - Etc.

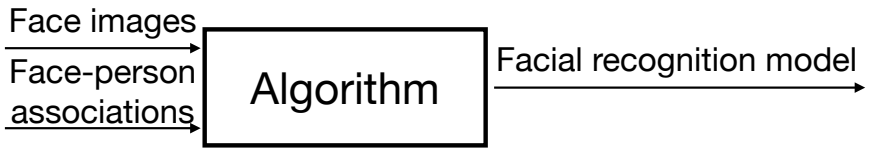
Machine learning

- Instead of a human hard-coding pre-specified rules for the algorithm to follow, the machine must figure out ("learn") the rules for the human
  - Generate the rules from the provided data

Machine learning (cont'd)

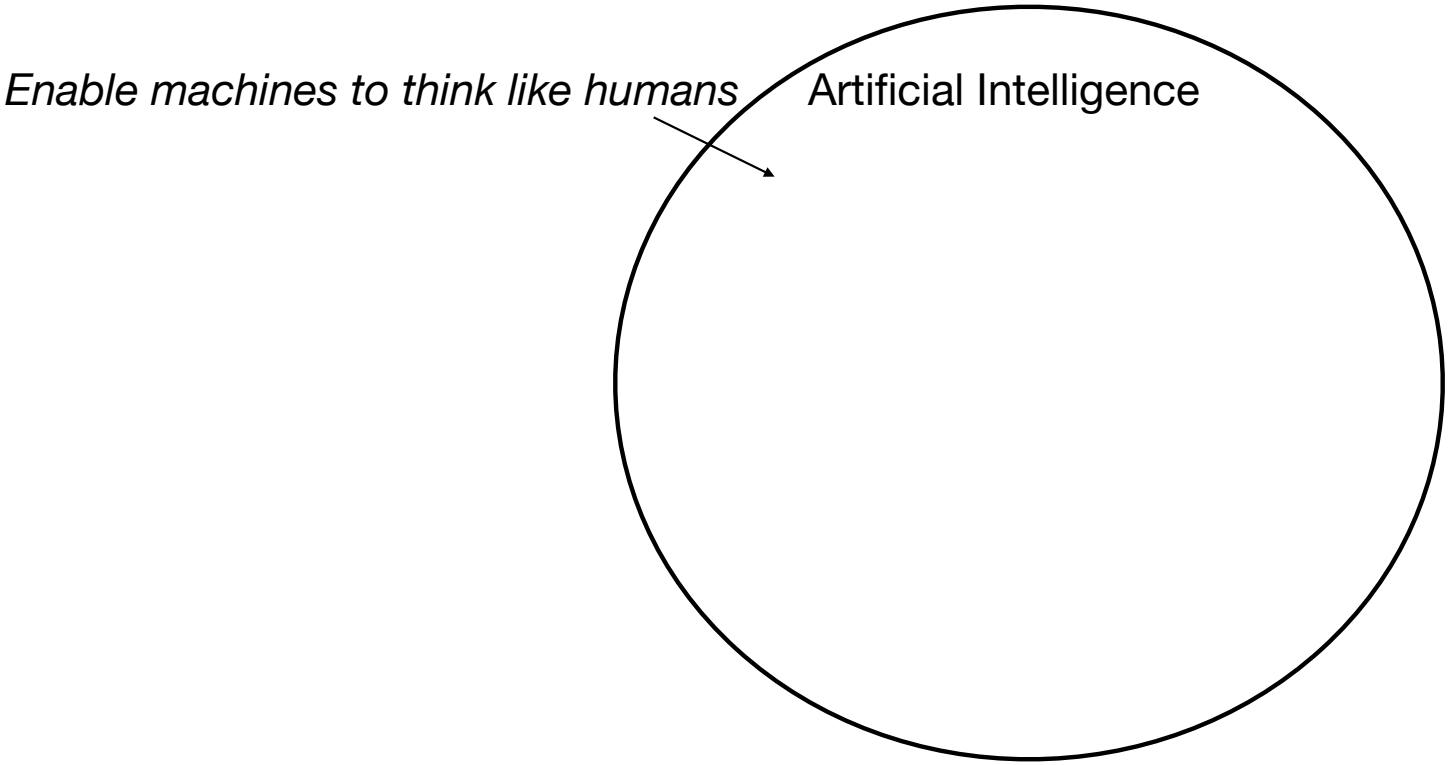
- Objective is to obtain a predictive model that will be able to infer answers on new unseen data
  - We already know answers for data we have





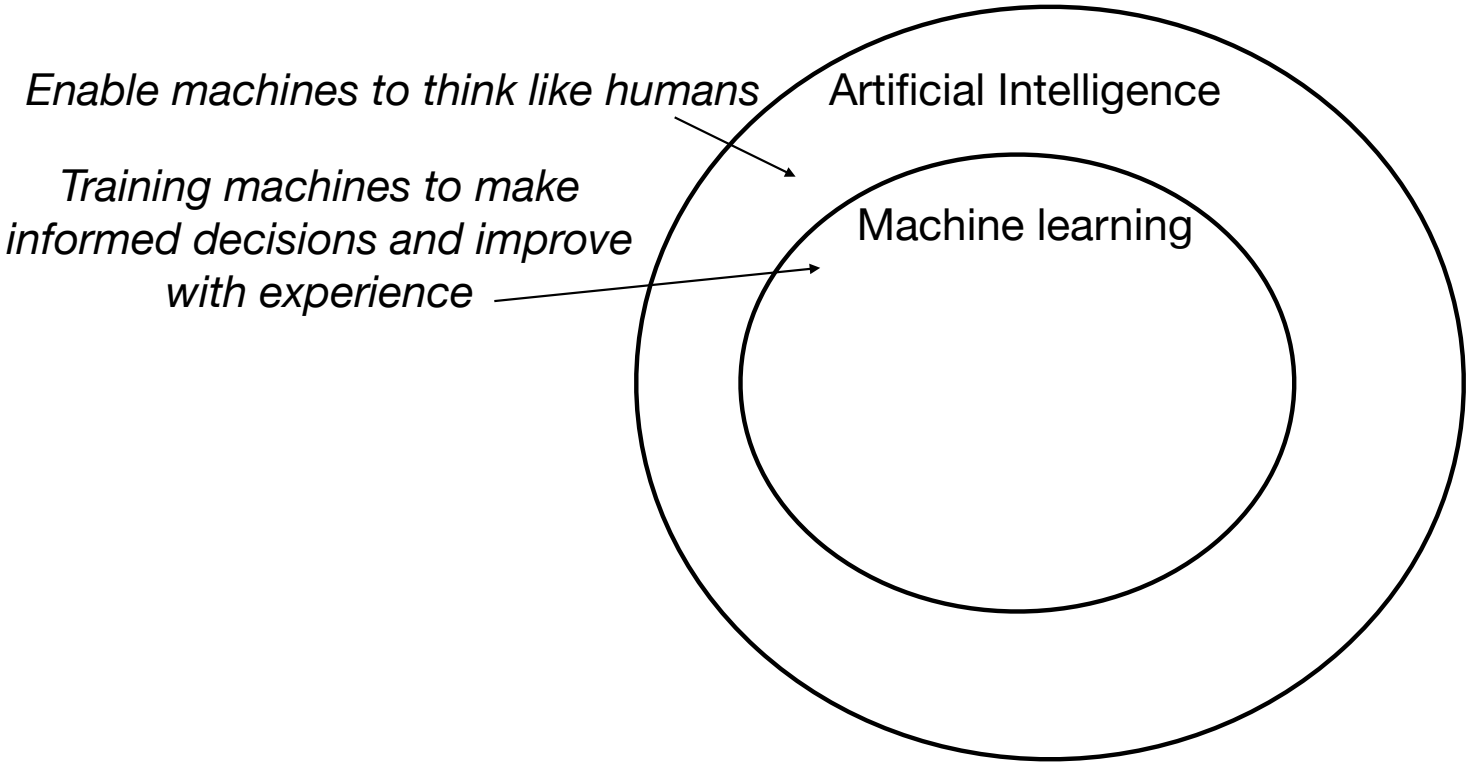
AI vs. machine learning vs. deep learning

- Machine learning is a part of artificial intelligence



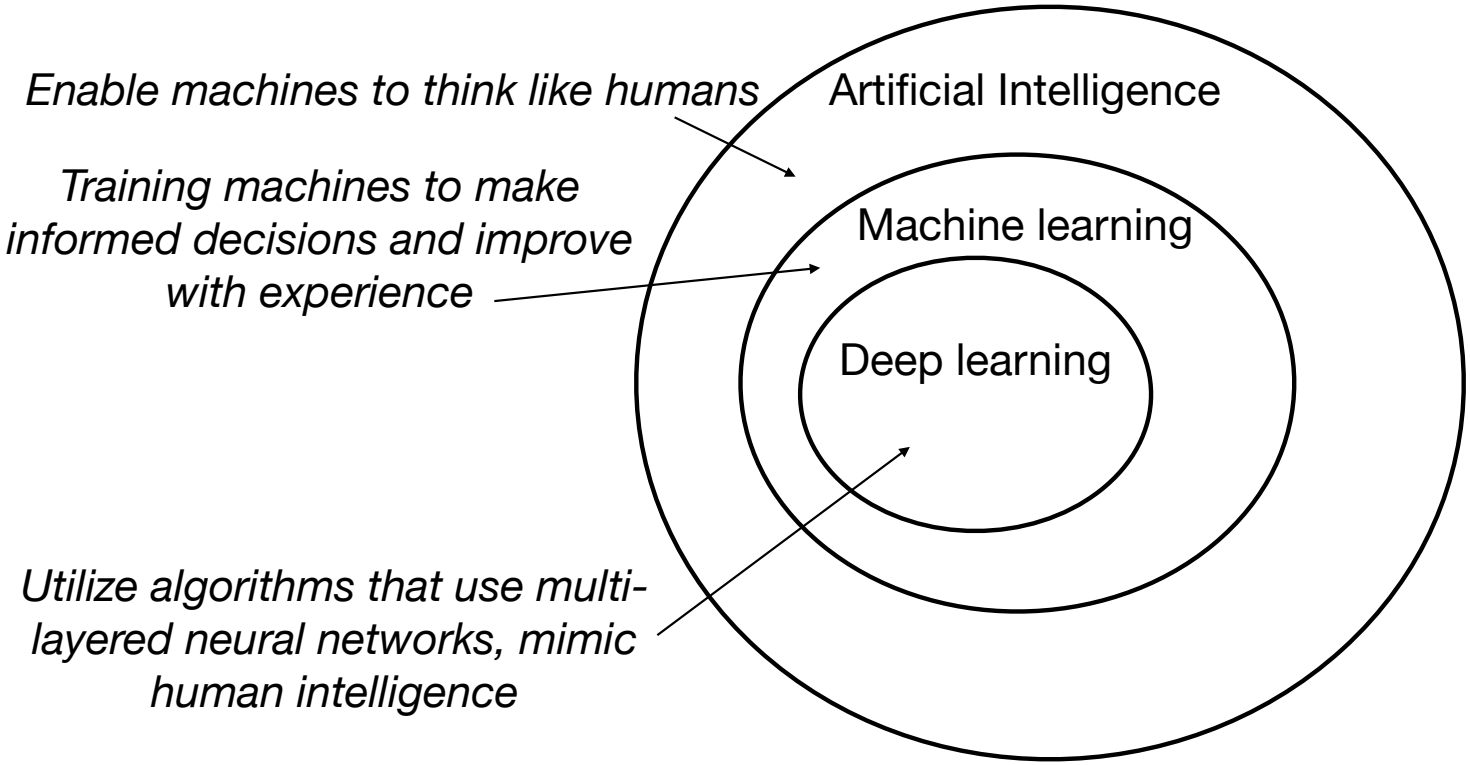
AI vs. machine learning vs. deep learning

- Machine learning is a part of artificial intelligence

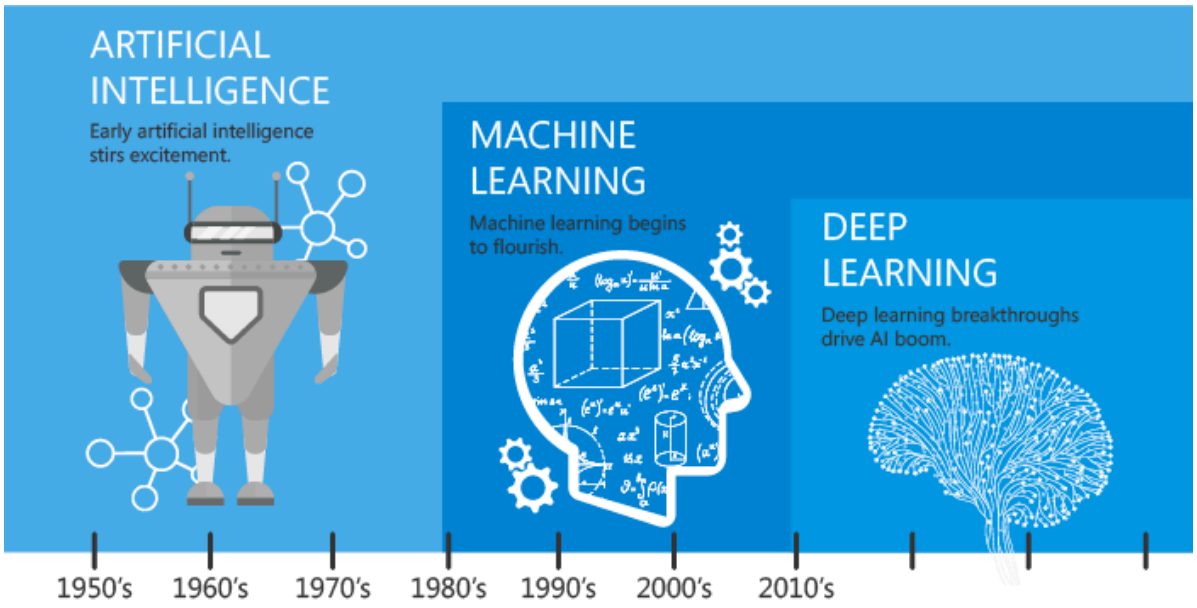


AI vs. machine learning vs. deep learning

- Machine learning is a part of artificial intelligence



Timeline of the field of AI



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.

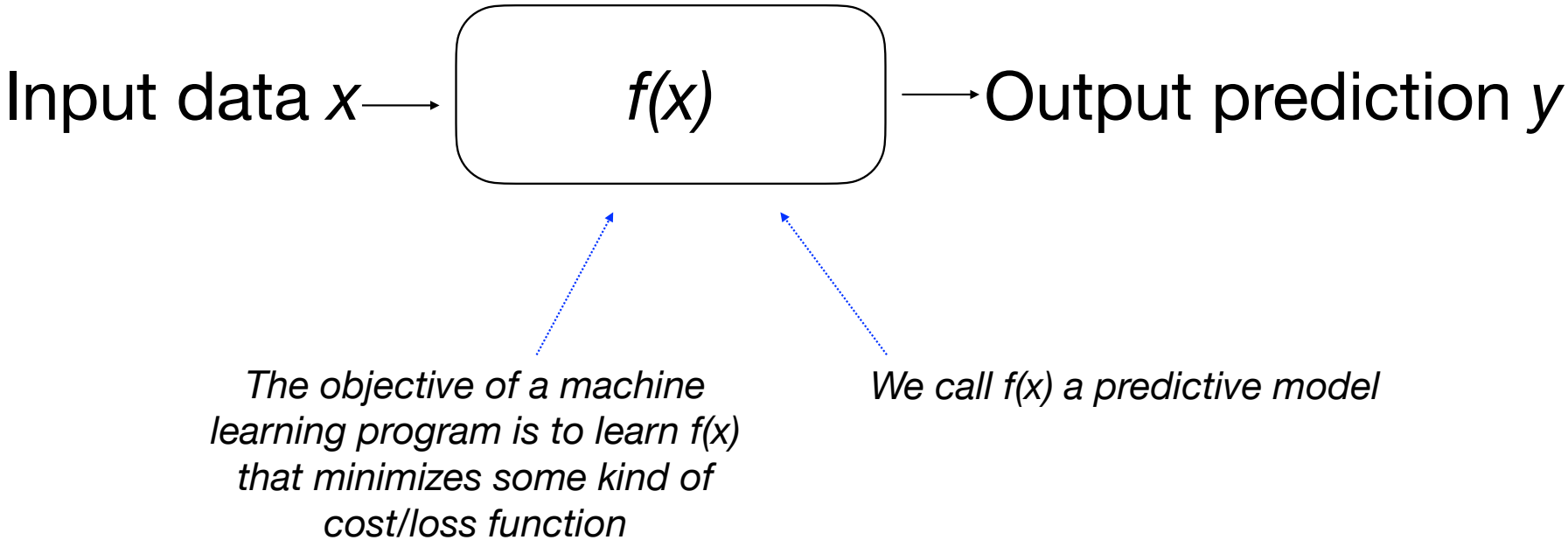
towardsdatascience.com

Predictive modeling

- Using machine learning programming we build models
- Machine learning solves problems where predictions must be made
- Hence, we call machine learning programming “predictive modeling”

What do we mean by rules?

- Rules = unknown function *f(x)*
- Typical prediction task:



How do we learn *f(x)*?

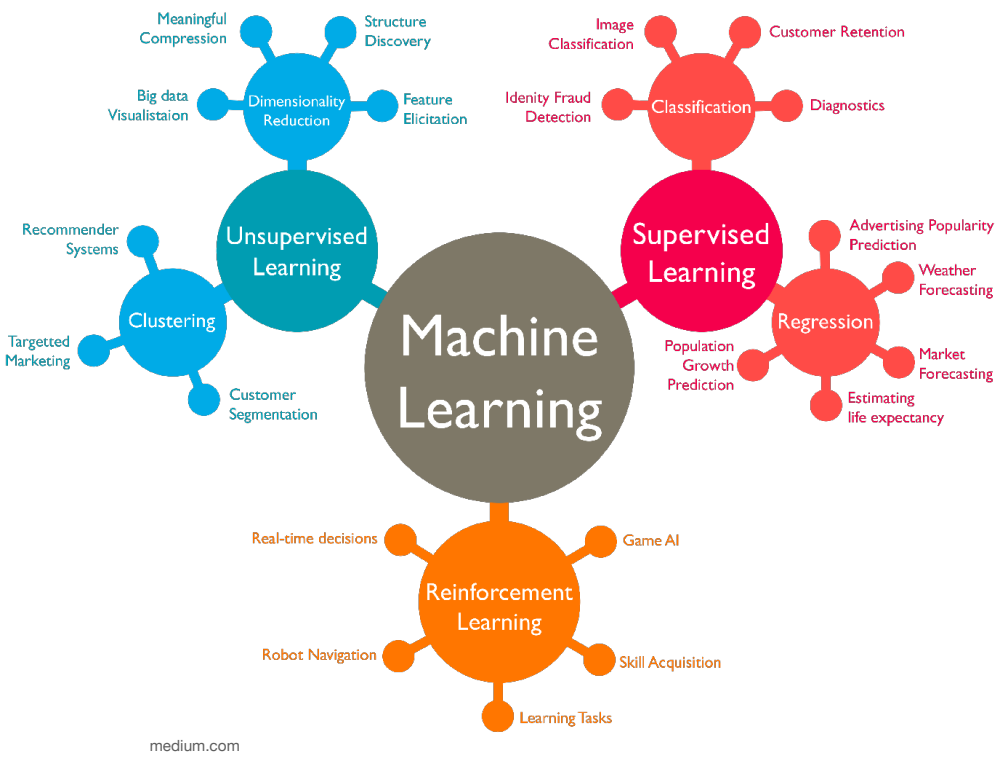
- We train the model on previously seen data
- We evaluate the model on set-aside unseen data

- What could go wrong?
  - The data we trained the model on is not representative of the unseen/future/real data
  - Training data has incorrect prediction labels
  - Training data is highly unbalanced in regards to class distribution
  - Training data contains highly covariate features
    - Redundancy in predictive power
  - Training data needs normalization/projection/transformation in order to be useful
  - Too many dimensions in relation to the number of observations
  - Provided data does not contain the patterns that lead to the desired answer
  - Bad data in = bad model out

Types of problems in machine learning

- **Classification** (supervised learning)
  - Learning a set of categorical labels from a set of observations
    - Two-class problems (binary problems)
    - Multi-class problems
- **Regression** (supervised learning)
  - Learning continuous numeric values from a set of observations
- **Clustering** (unsupervised learning)
  - Finding patterns in data in the absence of a response variable/label

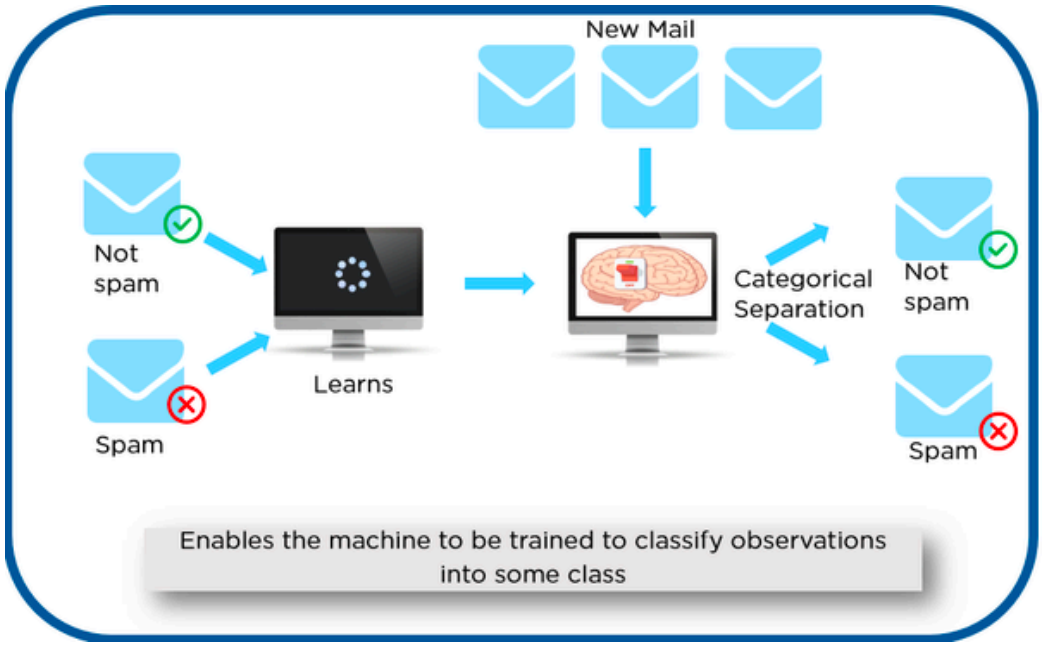
Types of problems in machine learning (cont'd)



Classification problems

- Supervised predictive modeling
- Output variable is a category (“apple”, “banana”, “mango”, etc.)
  - Each category is referred to as “class”
- Observations are grouped by a specific predetermined criteria
  - Predicting whether an image is of a cat
    - Cat vs. not (two class problem)
    - Cat vs. Dog vs. Horse vs. Whale (multi-class problem)
  - Predicting whether a patient has breast cancer from DNA mutations
    - Breast cancer vs. normal breast (two class problem)
    - Breast cancer vs. stomach cancer vs. esophageal cancer vs. brain cancer (multi-class problem)
  - Predicting sentiment of a customer service survey
    - Positive vs. negative (two class problem)
    - Fully satisfied vs. dissatisfied with the web site vs. dissatisfied with delivery times vs. dissatisfied with customer service representative (multi class problem)
- Criteria is known in advance and is used for learning *f(x)*

Classification problems (spam email example)

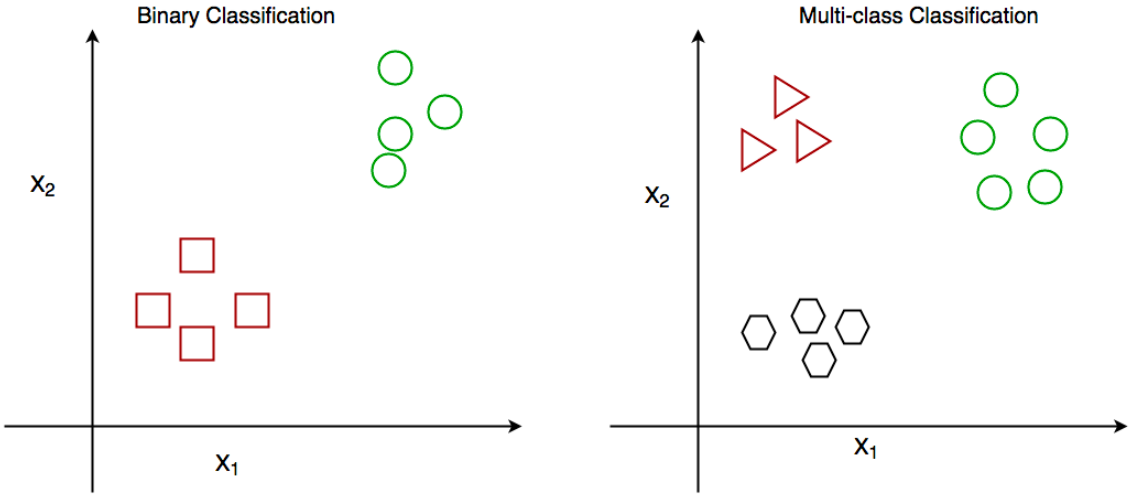


https://towardsdatascience.com

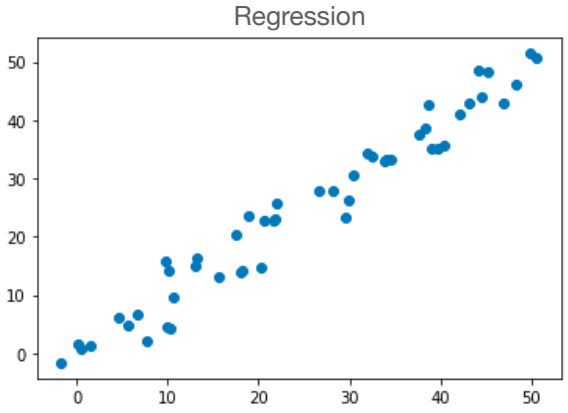
Regression problems

- Supervised predictive modeling
- Instead of a categorical label, the output variable is a real continuous value
  - Predicting individual's salary based on how much they spend in discretionary spending
  - Predicting stock price based on previous price fluctuations
  - Predicting price of a couch based on its dimensions and materials used in making it
  - Predicting patient's temperature based on blood pressure and heart rate

Classification vs. regression

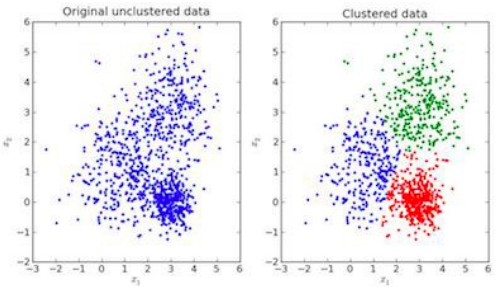


www.geeksforgeeks.org

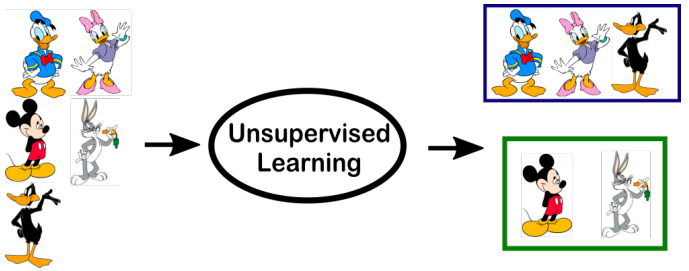


Unsupervised learning

- Finding groups and patterns in observations based on the input data
- Clustering data in high-dimensional or reduced space
- Data projection
- Types of clustering:
  - Hierarchical clustering
  - K-means clustering
  - K-NN (k nearest neighbors)
  - Principal Component Analysis
  - Singular Value Decomposition
  - Independent Component Analysis



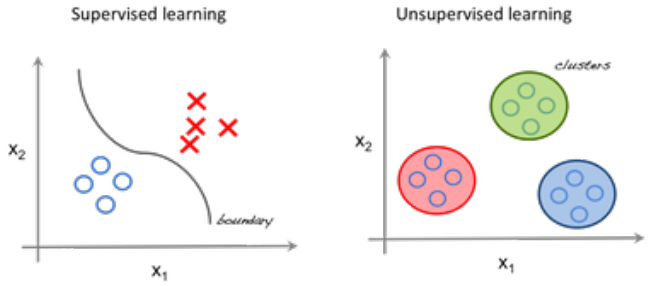
https://school.geekwall.in



https://towardsdatascience.com

Supervised vs. unsupervised learning

- Supervised - a model is trained using labeled data
  - Objective is to obtain a model that predicts categorical or continuous numeric labels with as much accuracy as possible
- Unsupervised - algorithms are used against unlabeled data
  - Objective is to discover patterns in the input data

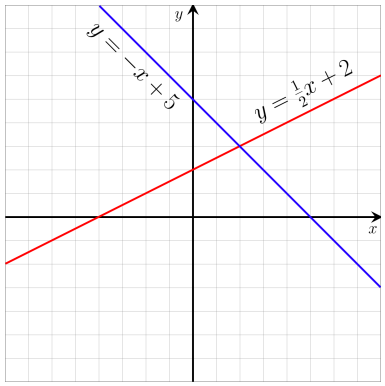


https://school.geekwall.in

Predictor vs. response variables

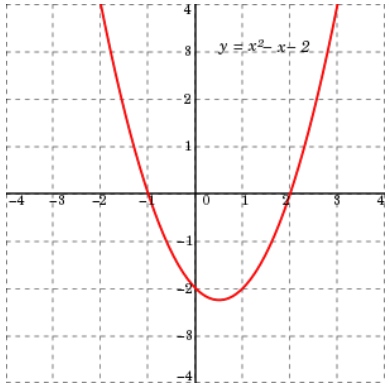
- Input data vs. answers
- Predictor variables are independent variables used to predict the dependent/ outcome variable
  - In equation ***y = f(x)***
    - ***x*** is an independent/predictor variable and ***y*** is the dependent/response variable
- In the case of a classification problem, the response/output variable is the label

Linear *f(x)*:

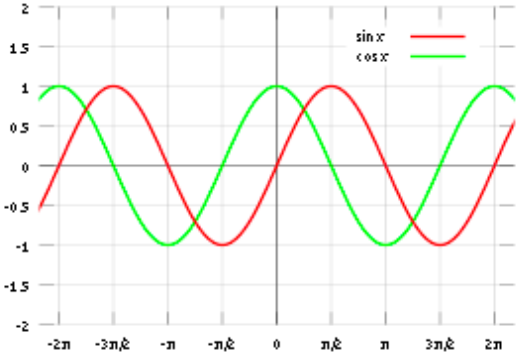


https://en.wikipedia.org

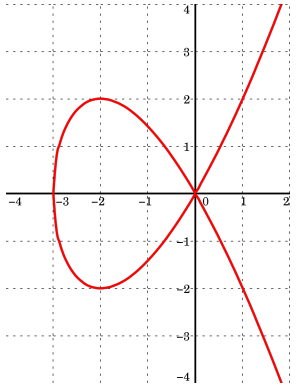
Quadratic *f(x)*:



Periodic *f(x)*:

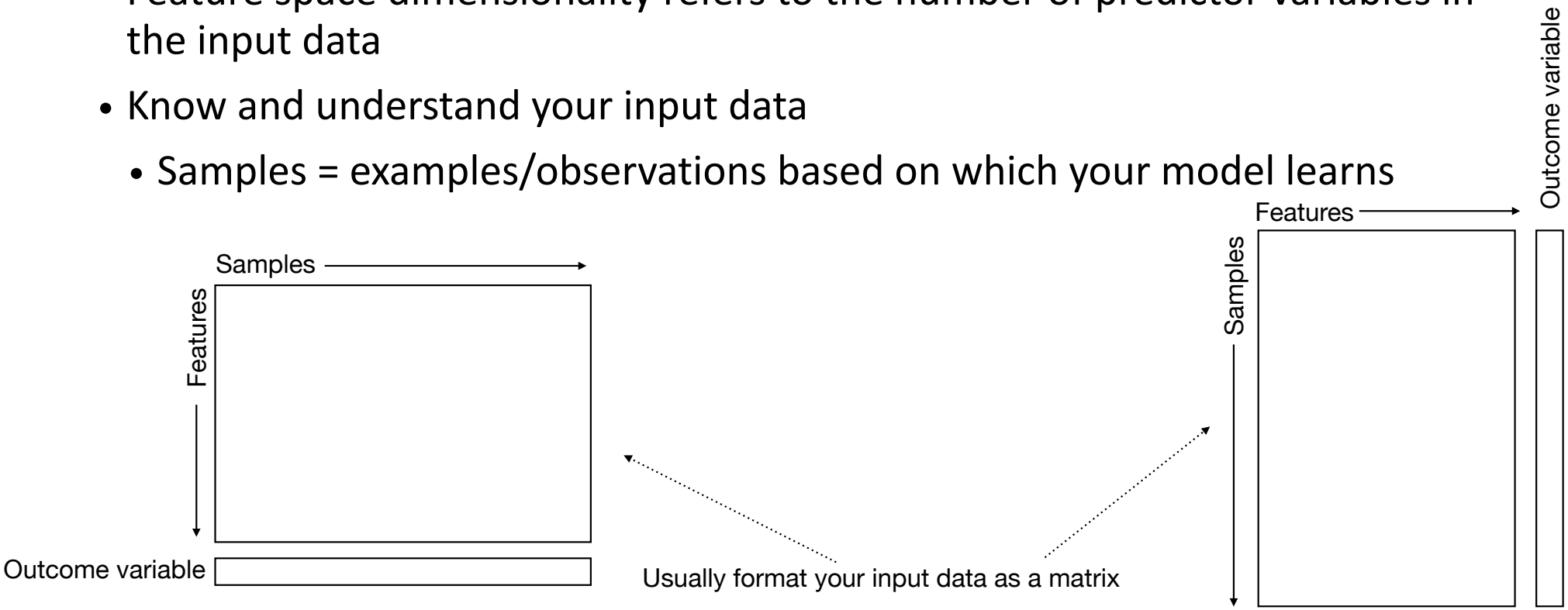


Unknown *f(x)*:



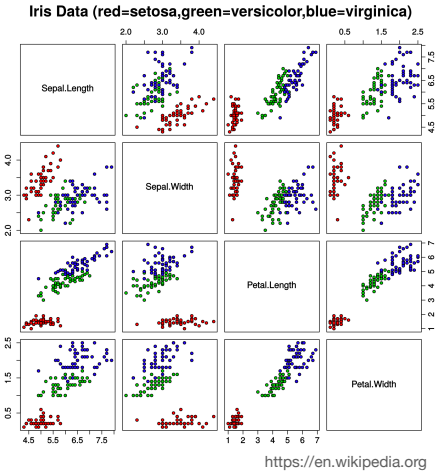
Feature space

- All the input variables used to predict the output/response variable comprise the feature space of the input data
- Feature space dimensionality refers to the number of predictor variables in the input data
- Know and understand your input data
  - Samples = examples/observations based on which your model learns

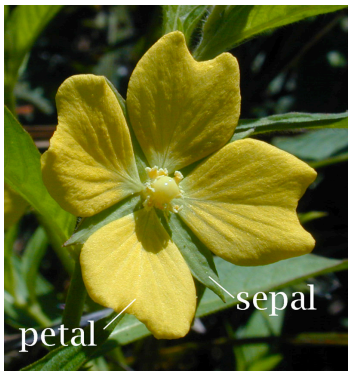


Iris dataset

- Fisher's *Iris* flower dataset
  - Introduced by Ronald Fisher in 1936 paper "The use of multiple measurements in taxonomic problems" in Annals of Eugenics
  - Describes morphological features of three species of Iris plant (*Iris setosa*, *Iris virginica* and *Iris versicolor*)
    - 50 samples from each group
    - 4 input features (continuous numeric), 1 output label (categorical, species)



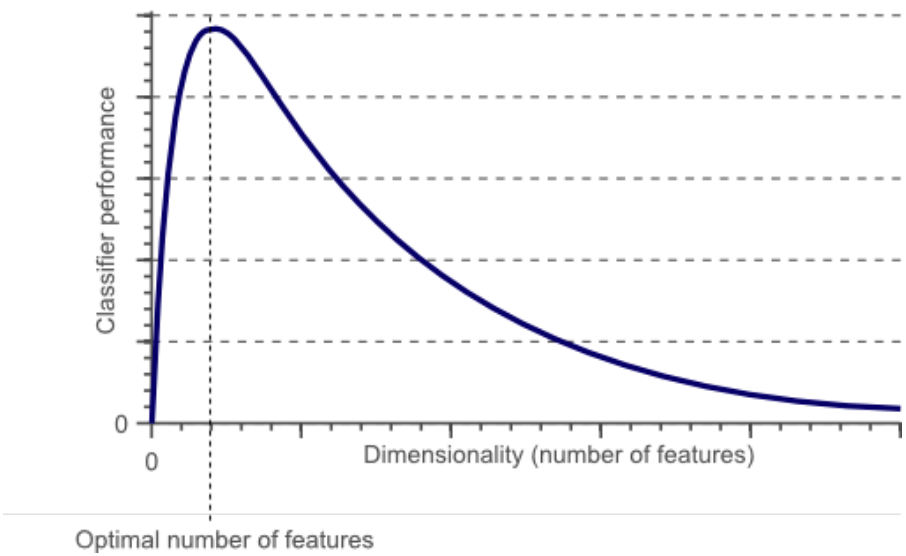
https://en.wikipedia.org



Curse of dimensionality

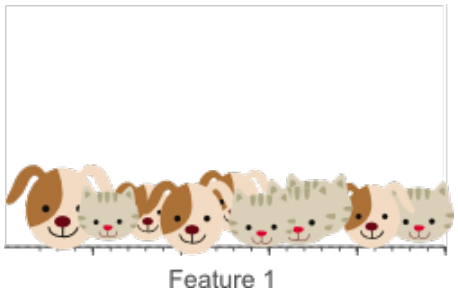
- Phenomenon that occurs when learning on data in a very high-dimensional space, especially with very few samples to learn from
  - Predictive power of a classifier or a regressor first increases with the increase of the number of dimensions and then diminishes
- Can also refer to sparsity of data
- Machine learning methods do best when there are enough examples to learn from
  - How does the model perform on the new/unseen/future data?
- High dimensionality prevents data organization into patterns
- Typical rule: 5 training samples for each dimension in the input data
  - Some domains just don't have enough samples (e.g. bioinformatics and biomedical research)
    - There are ~20,000 genes with individual measurements and the number of samples are usually in hundreds (in a good dataset)

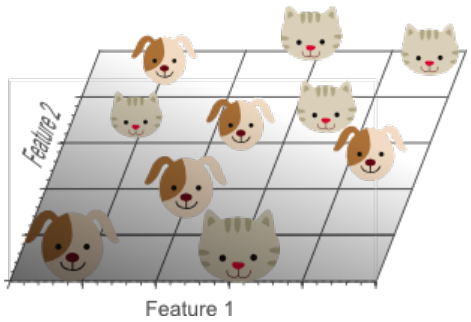
Curse of dimensionality (cont'd)

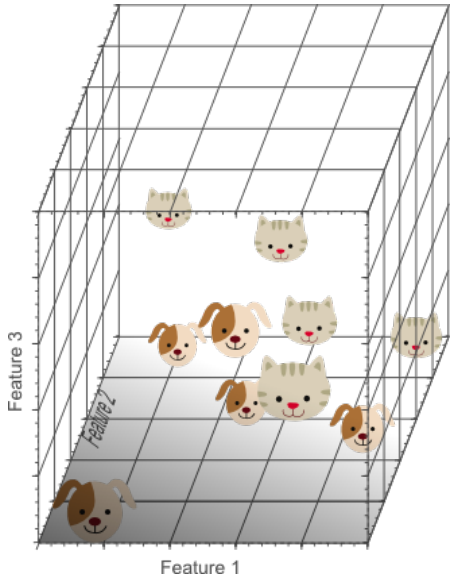


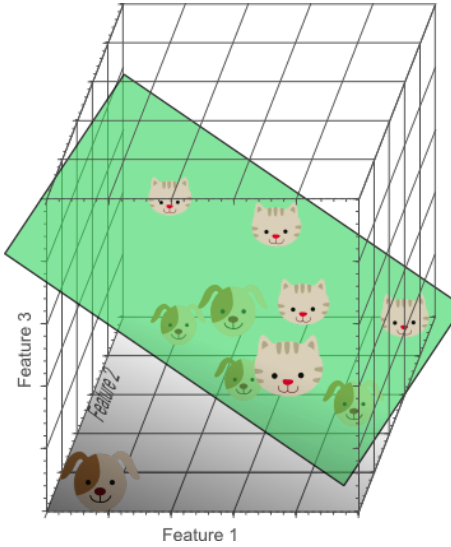
www.visiondummy.com

Curse of dimensionality (cont'd)





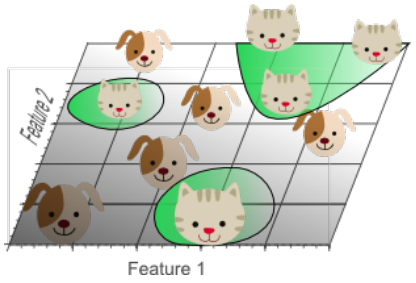




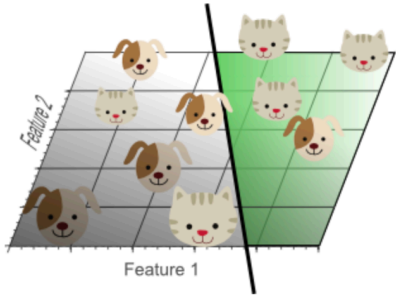
www.visiondummy.com

Dogs and cats are not easily separable in 1-D or 2-D but adding the 3rd dimension creates linear separability

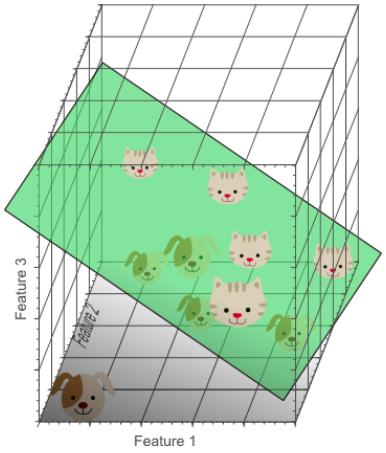
Curse of dimensionality (cont'd)



This model is over-fitted to our input data and is not representative of another random sample

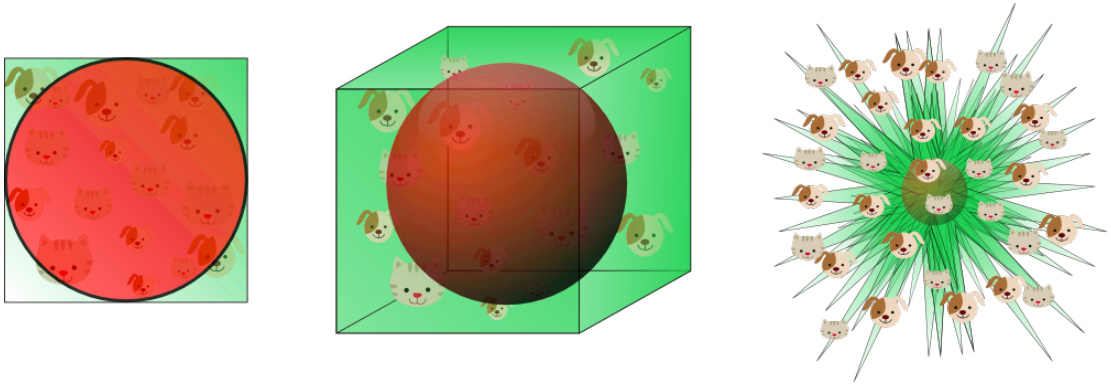


This model is not over-fitted but also does not produce good accuracy



This model separates classes well without obvious over-fitting. The more features we use the more likely we will be able to separate our data well but also risk over-fitting the model

Curse of dimensionality (cont'd)



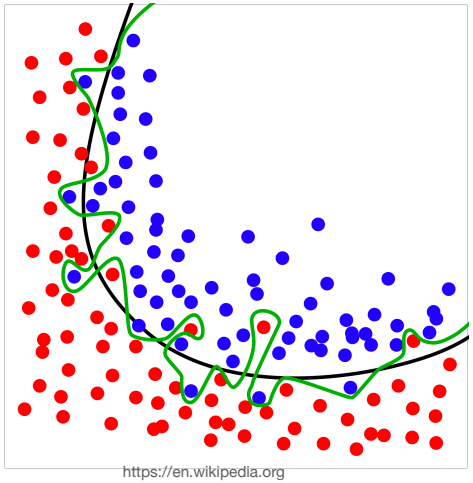
www.visiondummy.com

With the increase of dimensionality more observations are going to occupy their own non-overlapping with other observations space and it becomes easier to find the separation function that will fit that particular dataset perfectly

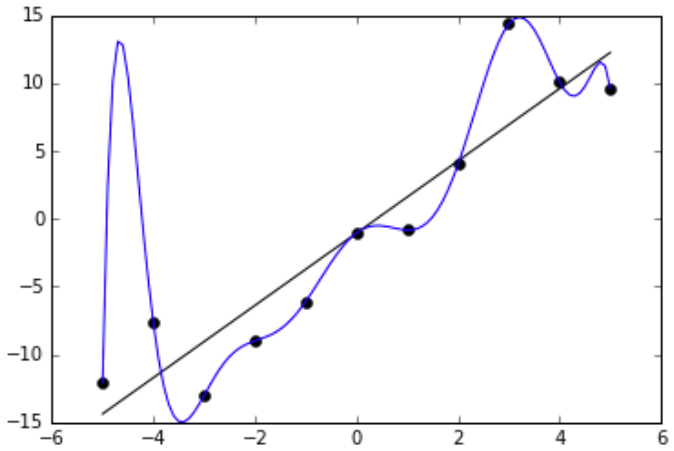
Model overfitting

- High-dimensionality often leads to model overfitting
  - Too many dimensions to consider for the algorithm, not enough observations to learn a good predictive function
- Overfitting - a phenomenon in machine learning and statistics when a model is too well fitted to the particular dataset
  - Model predicts with high accuracy on given data but does not perform well on new/unseen data
- Goodness of fit - statistical term that refers to how well the model fits the data

Model overfitting (cont'd)



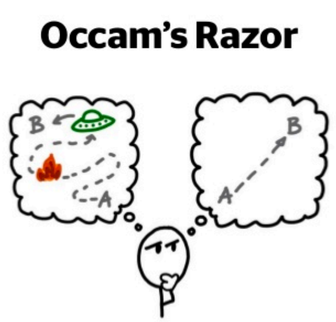
https://en.wikipedia.org



https://en.wikipedia.org

Occam's razor

- Law of parsimony
- A principle in problem solving, which states that the simplest explanation is usually the right one
  - Originated from philosopher William of Ockham



"When faced with two equally good hypotheses, always choose the simpler."

automaticaddison.com

Ockham chooses a razor

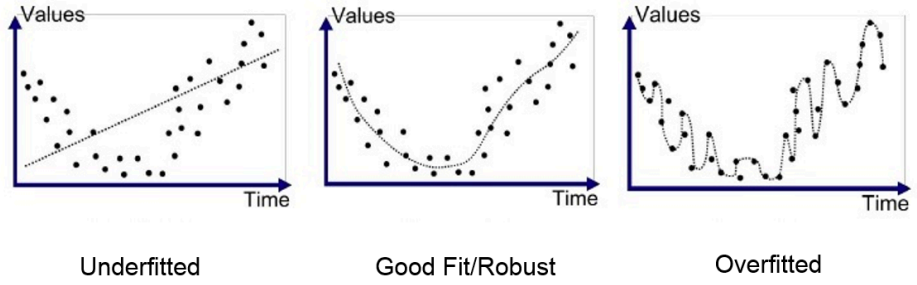
www.dictionary.com

You can also under-fit a model

- Under-fitted model does not properly capture the rules/patterns produced by the data
- An under-fit model does not perform well with new/unseen data

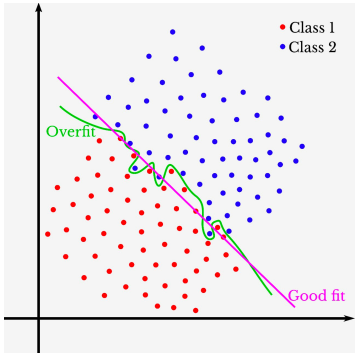
Over- vs. under-fitting a model

Regression problems:

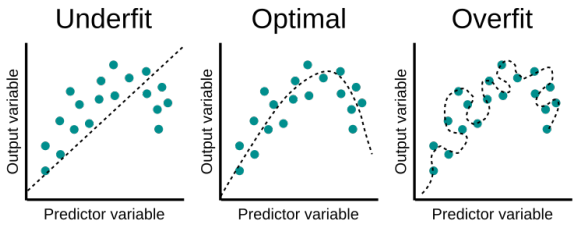


medium.com

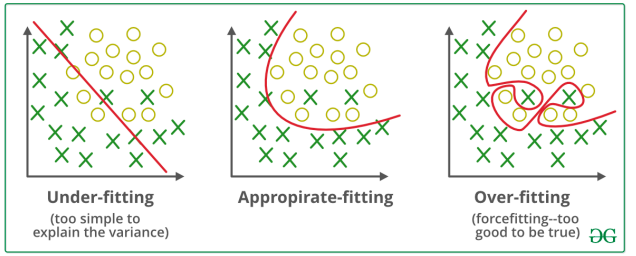
Classification problems:



towardsdatascience.com



www.educative.io

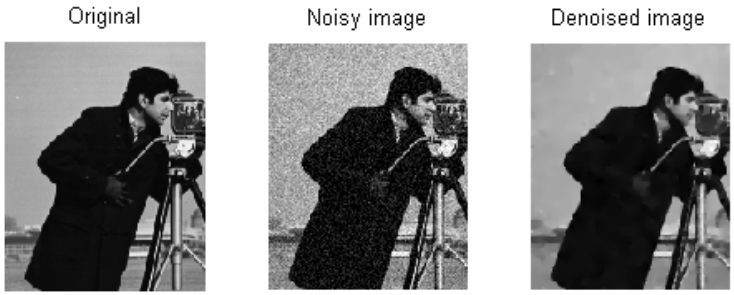


www.geeksforgeeks.org

Noisy data

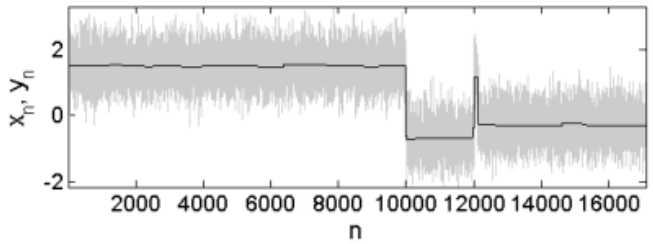
- Noise in the data refers to presence of values that make it more difficult for the algorithm to find *f(x)* that predicts *y* with acceptable accuracy
- Noise can come from a number of sources (bad data, faulty equipment recording, errors in data entry, missing data, not enough variance in the data to separate different classes, etc.)
- Noise can contribute to model overfitting

Noise in an image data:



https://en.wikipedia.org

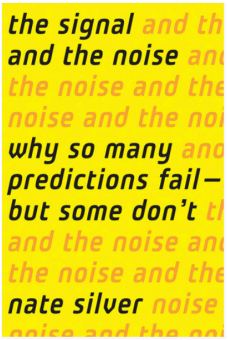
Noise in a signal data:



Signal vs. noise

- Signal - true patterns in the data
- Noise - irrelevant and random information
- Noise interferes with signal and makes it difficult to recognize it
- An algorithm can “memorize” the noise in the input data rather than find and learn the signal
- The more complex the model is the more likely it is to “memorize” rather than “learn”

- “*The Signal and the Noise: Why Most Predictions Fail – but Some Don’t*” 2012 book by statistician Nate Silver
  - Case studies from baseball, elections, climate change, the 2008 financial crash, poker, and weather forecasting (https://en.wikipedia.org/wiki/The_Signal_and_the_Noise)



https://en.wikipedia.org

Feature space reduction

- Also referred to as dimensionality reduction
- Transformation/projection/rotation of the data from high-dimensional feature space to low-dimensional feature space
- Reduces the number of input/independent variables
- Very commonly involved in the fields of signal processing, speech recognition, neuroinformatics, bioinformatics, and many others
- Can be used as a part of data pre-processing for dimensionality reduction, noise reduction, data visualization, and unsupervised learning

Major approaches in feature space reduction

- Feature selection
- Feature extraction

- Can be applied to both supervised and unsupervised learning

- Advantages:
  - Simpler model (remember the Occam's razor principle)
  - Shorter training times
  - Less resource usage
  - Helps to avoid the curse of dimensionality
  - Reduces overfitting of the model

How does feature selection work?

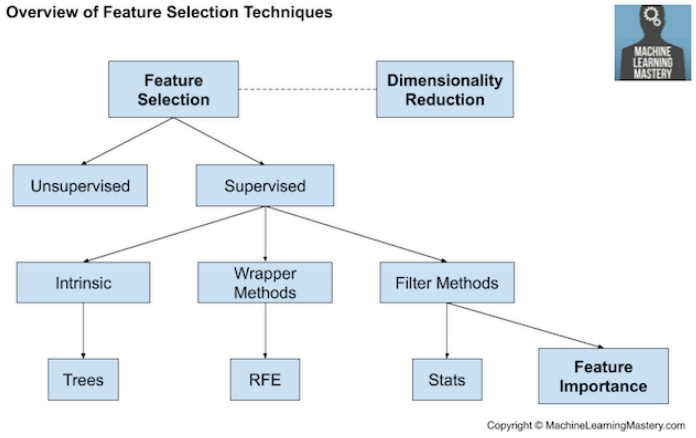
- Selecting a subset of relevant features
  - Select from the features already present in your input dataset
- Eliminating highly covariate features reduces redundancy in predictive power
  - Redundant and irrelevant are two different things
- For supervised learning, not all features might be predictive of the output variable
- For unsupervised learning, not all features might be able to show patterns in the data

Feature selection approaches

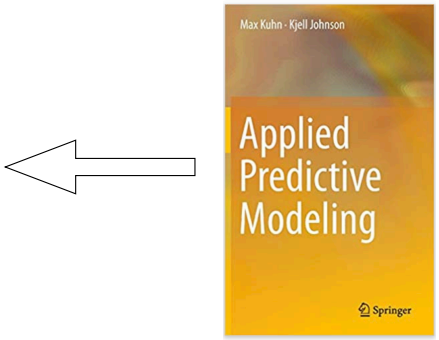
- You can think of feature selection approaches in terms of supervised vs. unsupervised
  - Is the output variable taken into account when selecting features?
- For supervised feature selection, wrapper vs. filter methods
  - Wrapper approaches create many models with subsets of input features
  - Filter methods use statistical techniques to identify those features most predictive of the label
- Intrinsic/embedded feature selection
  - Built into the algorithm itself

- Feature scoring (supervised, filter)
- Eliminate features with low variance (unsupervised)
  - Top X variant features
- Find features that highly correlate and remove redundant variables
- Some algorithms contain built-in feature selection
  - Penalized/regularized regression model (e.g. Lasso regression)
  - Random forest (supervised, wrapper)

Feature selection at a bird's eye view

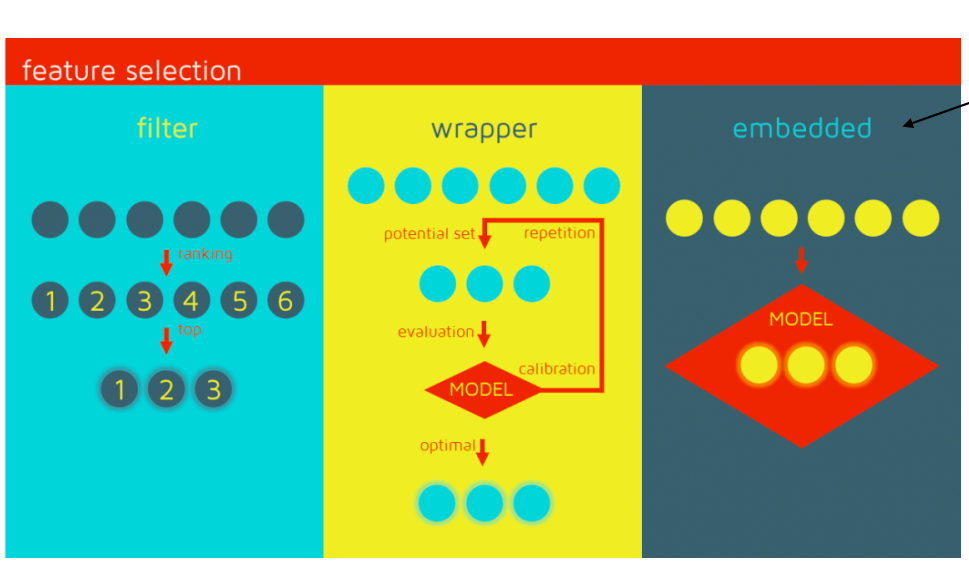


machinelearningmastery.com



ISBN-13: 978-1461468486
ISBN-10: 1461468485

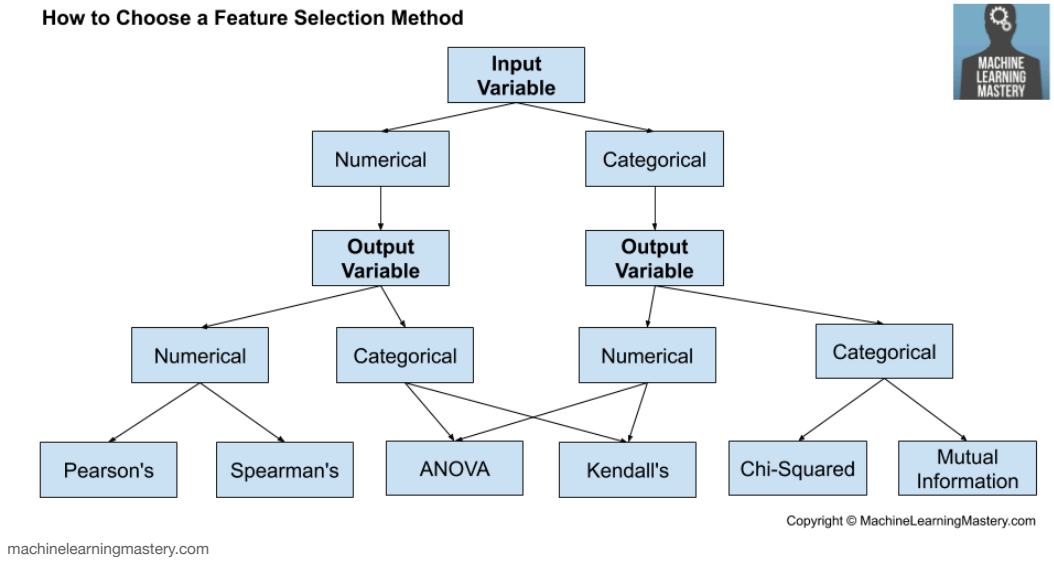
Supervised feature selection



Intrinsic

quantdare.com

How to use the appropriate statistical tests for your feature selection?



Feature extraction

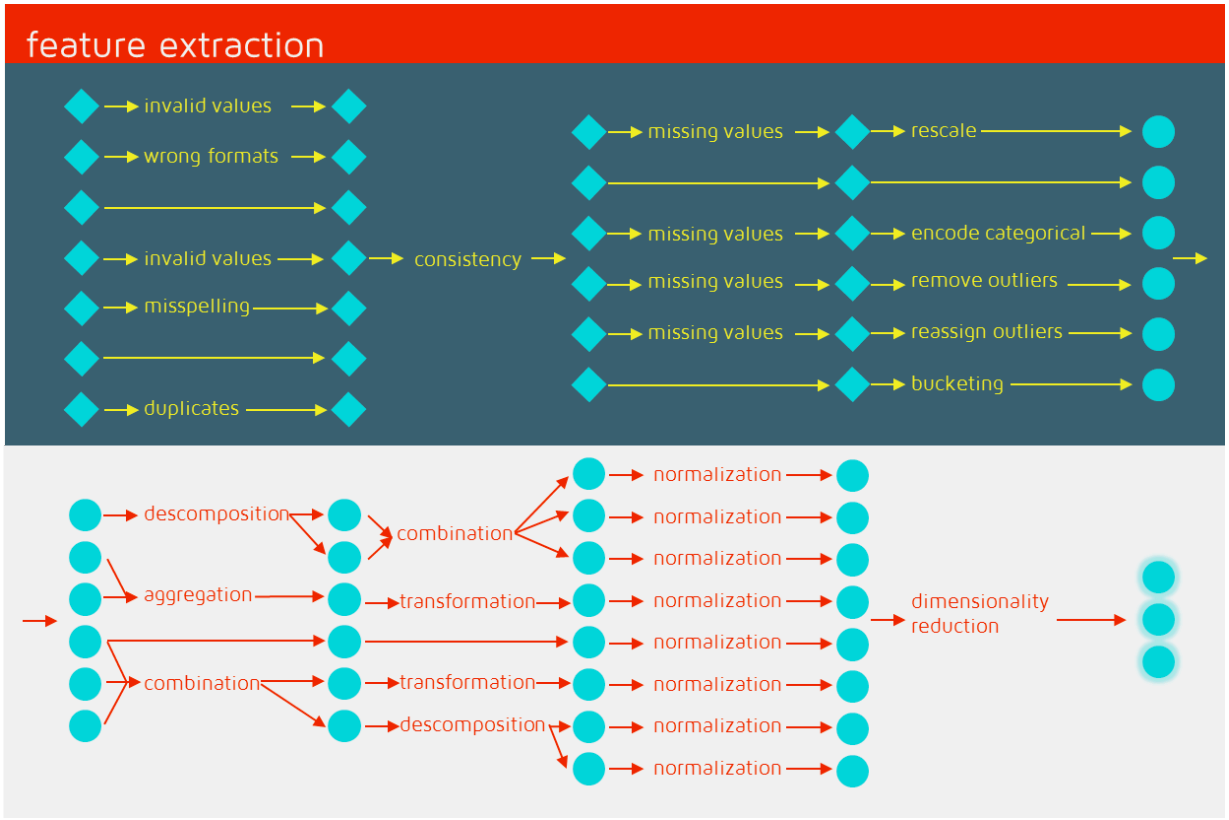
- Transform given features into different features
  - For the purposes of dimensionality reduction, a smaller set of features
    - Minimize the number of variables in the data

- Some of the commonly used algorithms
  - Principle Component Analysis (PCA)
  - Single Value Decomposition (SVD)
  - Independent Component Analysis (ICA)
  - Isomap
  - t-distributed Stochastic Neighbor Embedding (t-SNE)
  - Uniform Manifold Approximation and Projection (UMAP)
  - Self Organizing Map (SOM) or Self Organizing Feature Map (SOFM)
  - Linear Discriminant Analysis (LDA) or Quadratic Discriminant Analysis (QDA)
  - Various embedding methods
  - Autoencoders

Feature extraction example

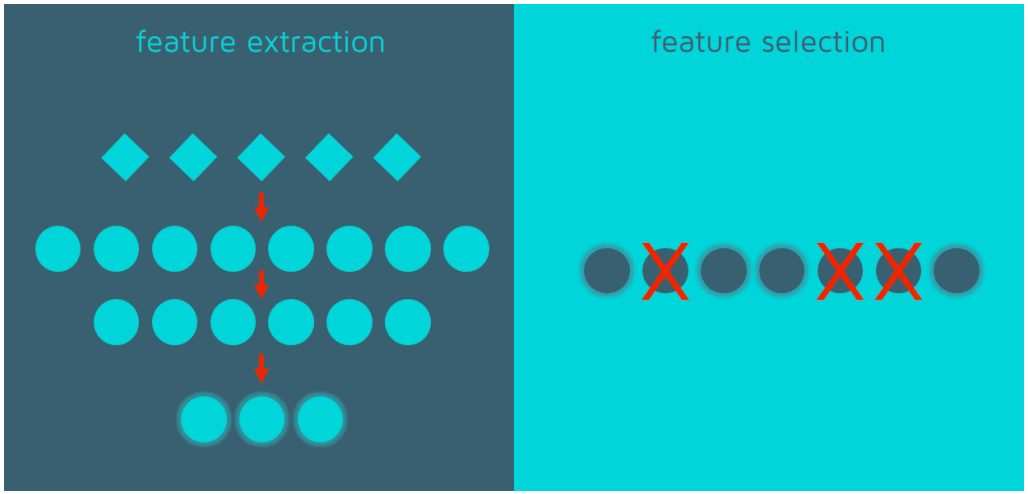
Started with 7 features, ended up with 3 different features originating from the starting features.

Feature extraction pipelines will differ for each dataset and each ML task. Here is an example of some feature extraction pipeline. Note that this is only a single example out of many-many ways to perform feature extraction.

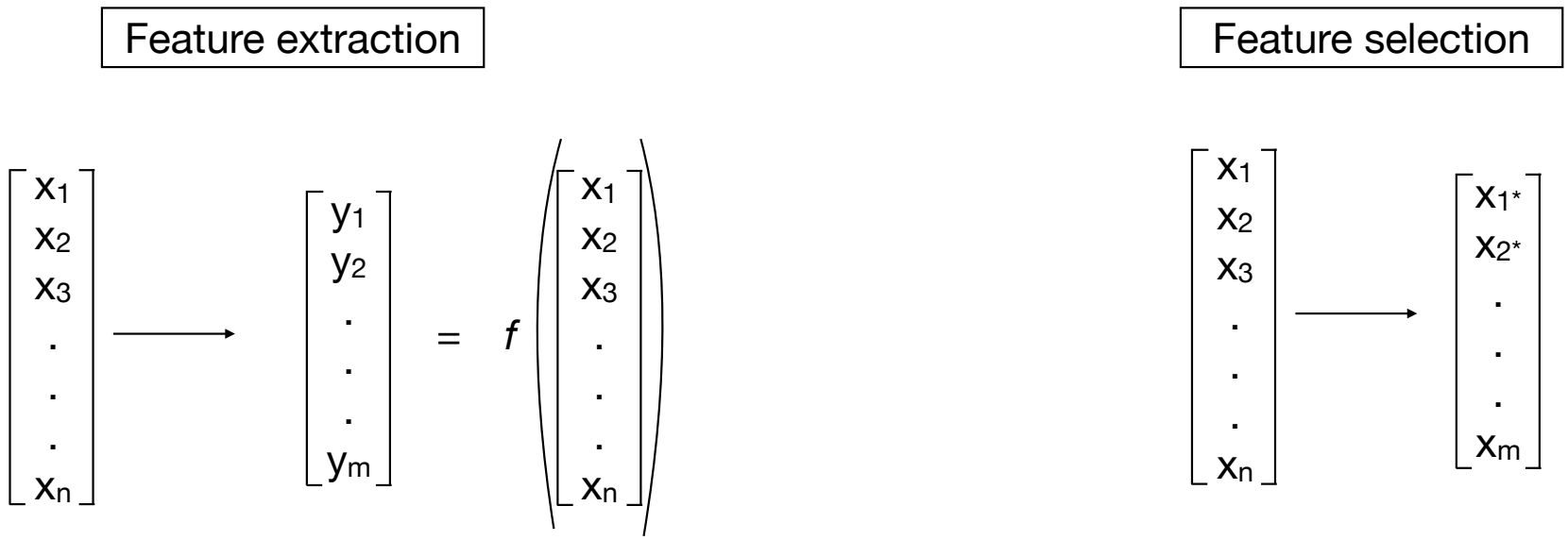


quantdare.com

Feature selection vs. feature extraction



Feature selection vs. feature extraction (cont'd)



Feature selection followed up by feature extraction

- Sometimes it’s appropriate to do both
  - Why not?

- Step1: feature selection
- Step2: feature extraction

Feature engineering

- Using domain knowledge to construct features from raw data using expert opinion, data mining, and machine learning techniques
- Considered to be a field of applied machine learning
- Helps you get the most out of your data to build a predictive model
- Better features lead to simpler model (remember Occam's razor?)
- Representation problem
  - What is the best representation of the data for this prediction task?

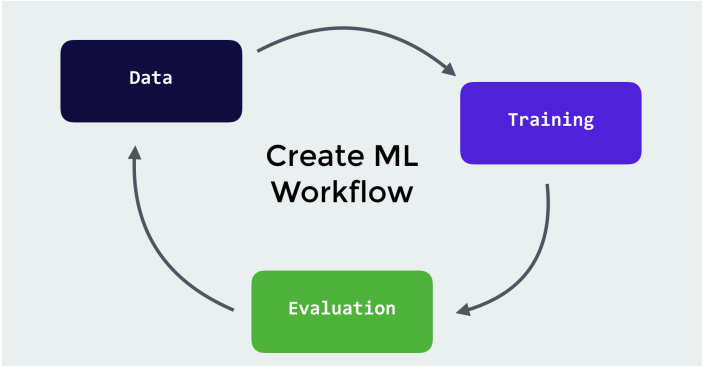
Feature engineering is the process of transforming raw data into features that better represent the underlying problem to the predictive models, resulting in improved model accuracy on unseen data.

feature engineering is manually designing what the input x's should be

https://machinelearningmastery.com
Tomasz Malisiewicz

Training a machine learning model

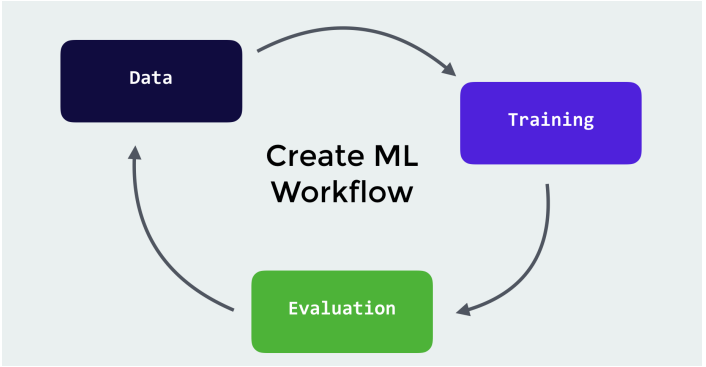
- In order to obtain a model that performs our desired predictive task, we have to train this model on previously seen data/observations
  - Remember the types of supervised ML problems (classification and regression)?
- Learn the function that takes the input data and outputs the response value/label with high accuracy
  - Parameter optimization



www.appcoda.com

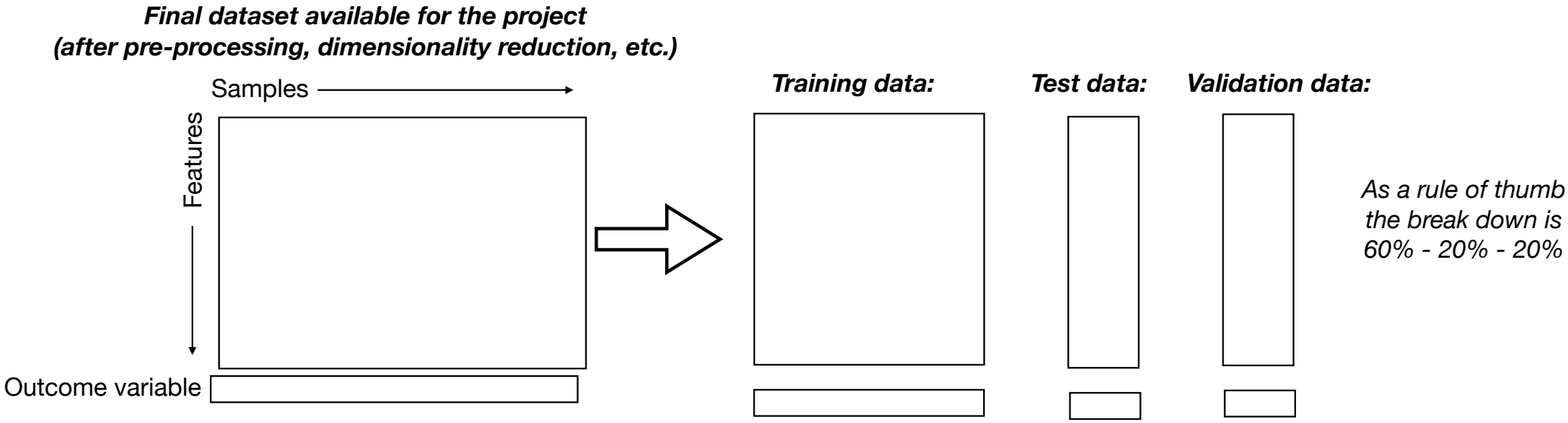
Training a machine learning model (cont'd)

- Evaluating the model on the same data as the model is being trained on is a sure way to over-fit your model
- You have to provide a previously unseen set of observations to evaluate how well your model predicts the output variable on data it has not had a chance to "memorize"



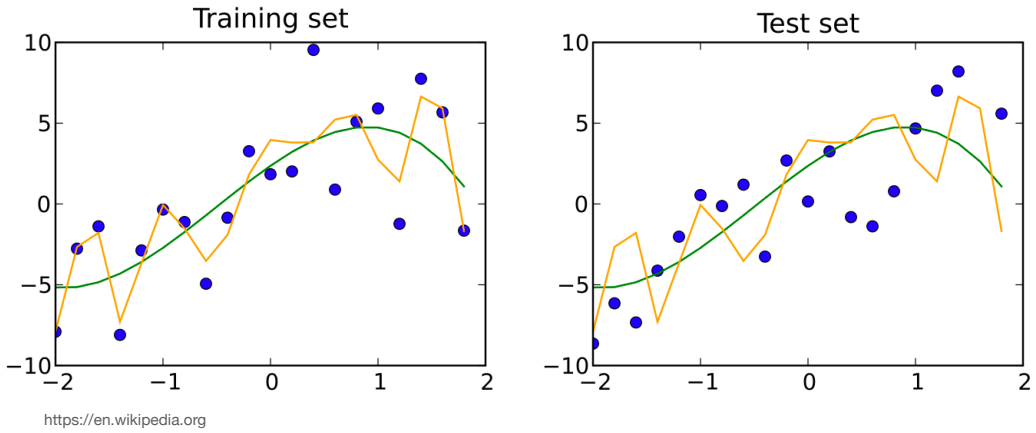
Training vs. validation vs. test data

- Training data is used to perform the initial fit of the model parameters
- Validation data is used for unbiased evaluation of that fit
- Test data is used for unbiased evaluation of the final model after all the parameters have been tuned and the model has been finalized
  - Often called holdout dataset



Training vs. validation vs. test data (cont'd)

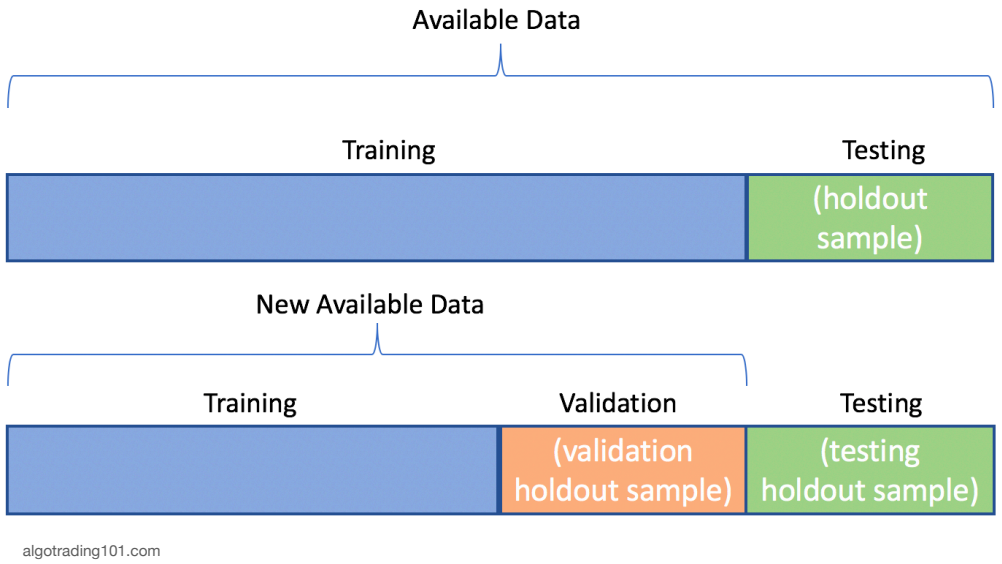
- Training data - examples used by the algorithm during the "learning" process
- Often "validation" and "test" terms get flipped and used interchangeably in both the industry and academia
  - Be careful to understand which type of dataset is being described in your resource
    - Ask yourself "is this dataset used to test intermediate model fit or the final model fit?"
- Both validation and test datasets should resemble the training dataset, otherwise the model will learn on different examples than it will be evaluated on
  - All three should resemble the real life unseen data



https://en.wikipedia.org

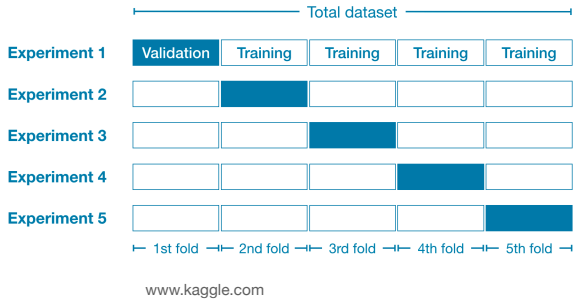
Training vs. validation vs. test data (cont'd)

- Sometimes your project is constrained by the lack of enough observations to set aside both validation and test data
  - Every ML project is unique and will require a bit of flexibility and consideration



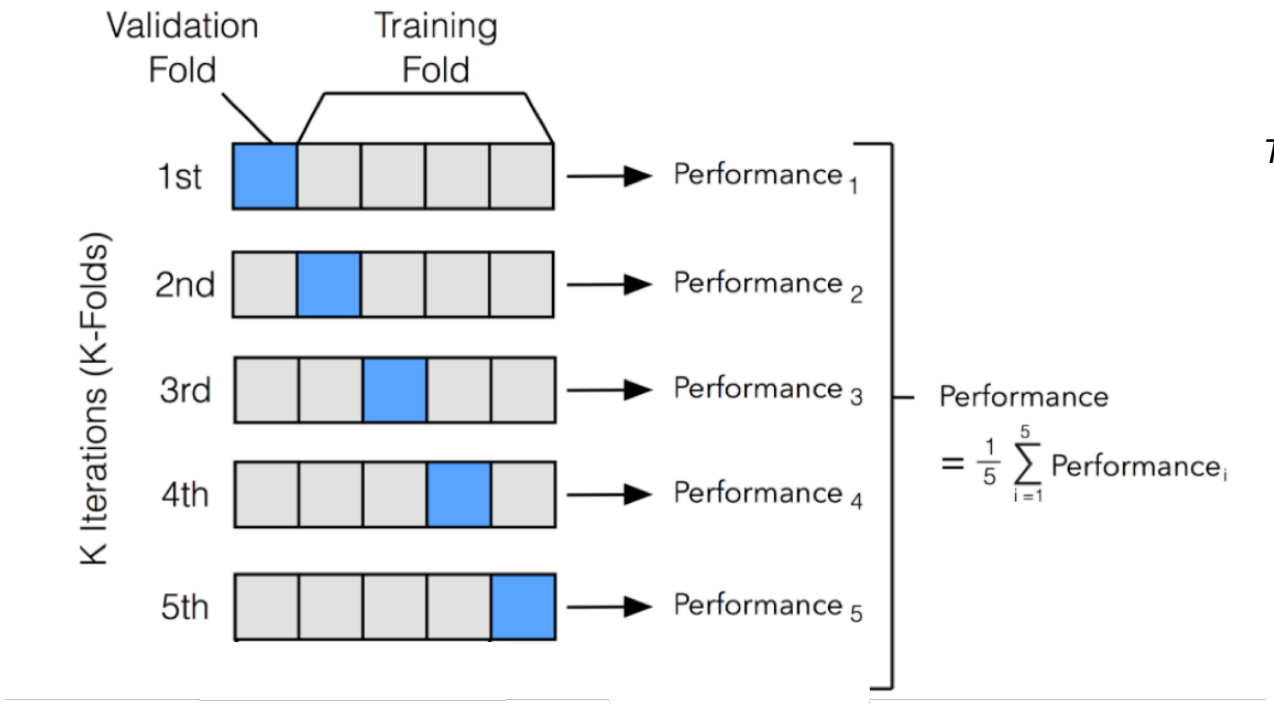
Cross-validation

- Sometimes the project does not have enough data for both training and validation datasets
- Cross-validation is an approach to resolve the lack of data
  - Procedure:
    - Break training data into X parts/slices/folds
    - For each fold, train the model on all the data except that fold and use that fold for validation dataset
    - Average performance across X models
    - We call this X-fold cross validation



This is 5-fold cross-validation

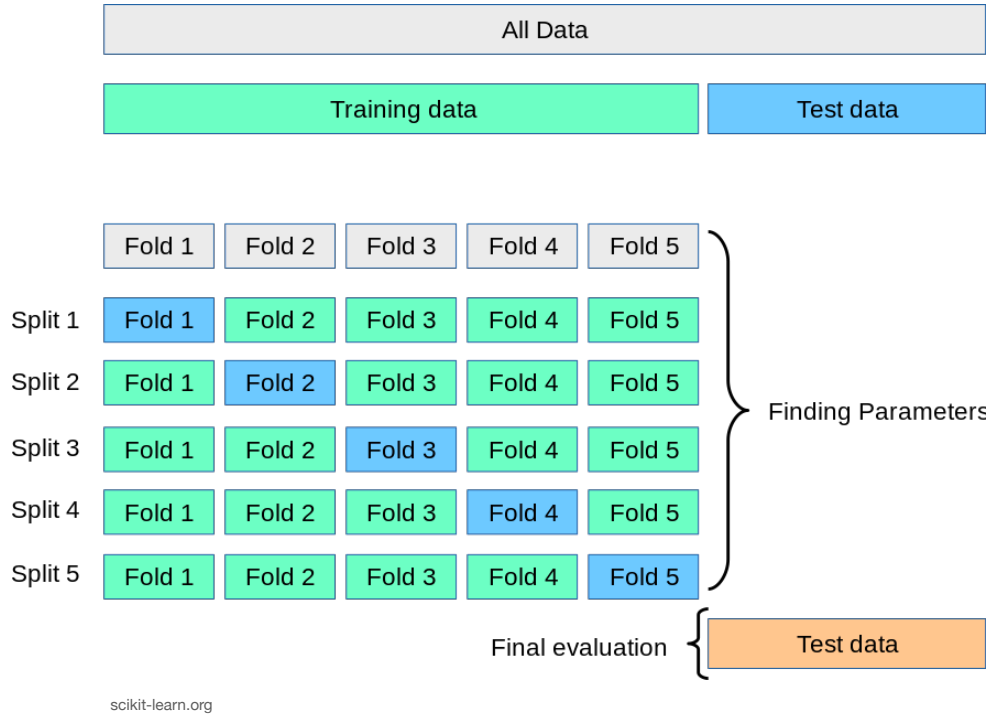
Cross-validation (cont'd)



http://ethen8181.github.io

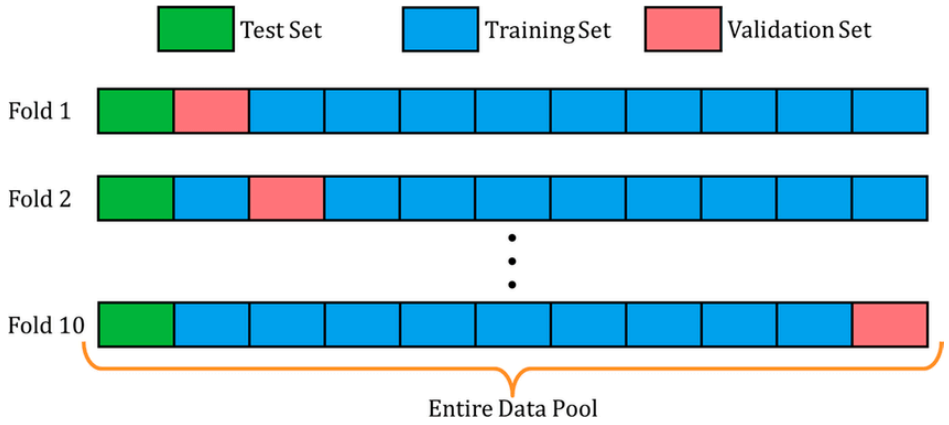
This is also 5-fold cross-validation

Cross-validation (cont'd)



This is also 5-fold cross-validation; notice the use of "test" term instead of validation in this figure

Cross-validation (cont'd)



This is 10-fold cross-validation

"A feasibility study for predicting optimal radiation therapy dose distributions of prostate cancer patients from patient anatomy using deep learning" 2019, Nature