UCCD2063

Artificial Intelligence Techniques

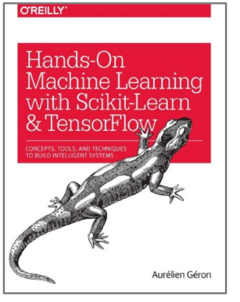
Unit 2:

The Fundamentals of Machine Learning

Outline

- **Machine Learning**
  - Introduction
  - Types of machine learning
  - Challenges of Machine Learning
  - The Machine Learning Framework

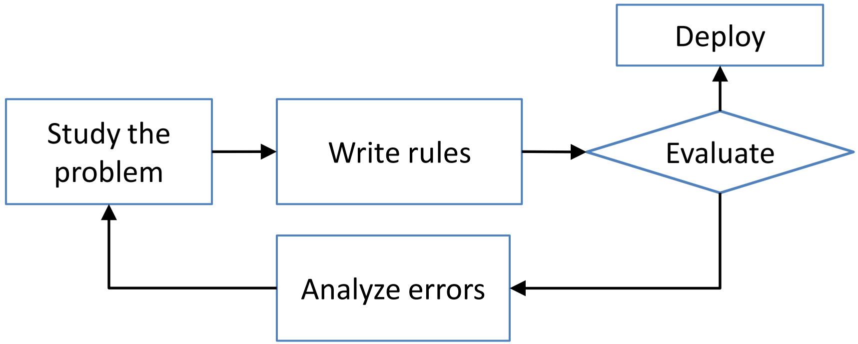
Reference:



Géron Chapter 1

Problems with Traditional Programming

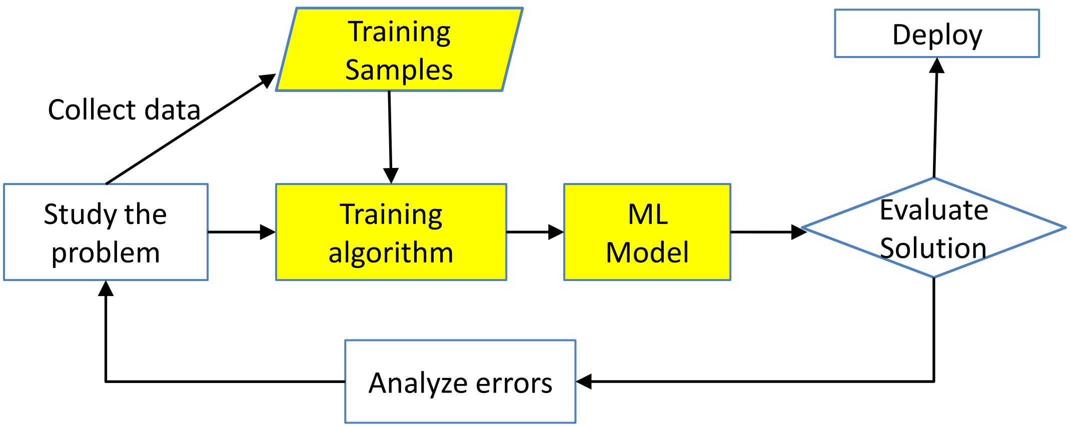
Traditional programming paradigm:



- Problems of traditional programming paradigm:
  - complexity – **too many rules**, very hard to cover all aspects of the problem, different problems require different rules
  - static – cannot adapt to new input, need to **keep writing new rules**, very hard to maintain

The Machine Learning Framework

- Instead of handcraft rules, ML learns a **model** from training samples (data)
- One learning algorithm for different problems



What is Machine Learning?

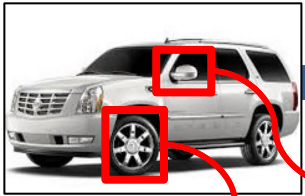
"Machine Learning: Field of study that gives computers the ability to learn (from data) without being explicitly programmed. "

[Arthur Samuel (1959)]

Why use Machine Learning?

- Some problems are too complex to solve by using rules
  - An example: image classification

Input



Complex, specific program

Output

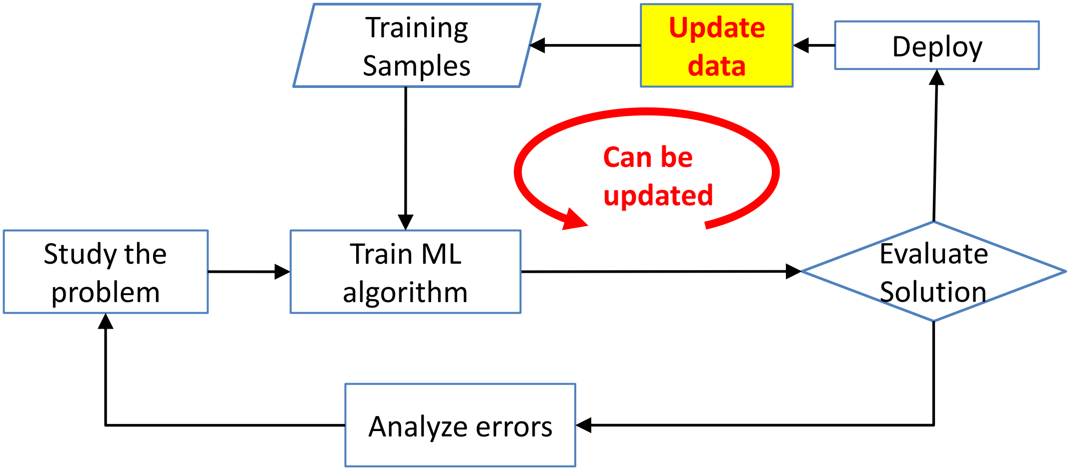
"car"

```
if this local region looks like a side mirror,
    and
if this local region area looks like a wheel
    then classify image as a car
```

Will work if given the same image again, but, given new images, the algorithm is bound to fail

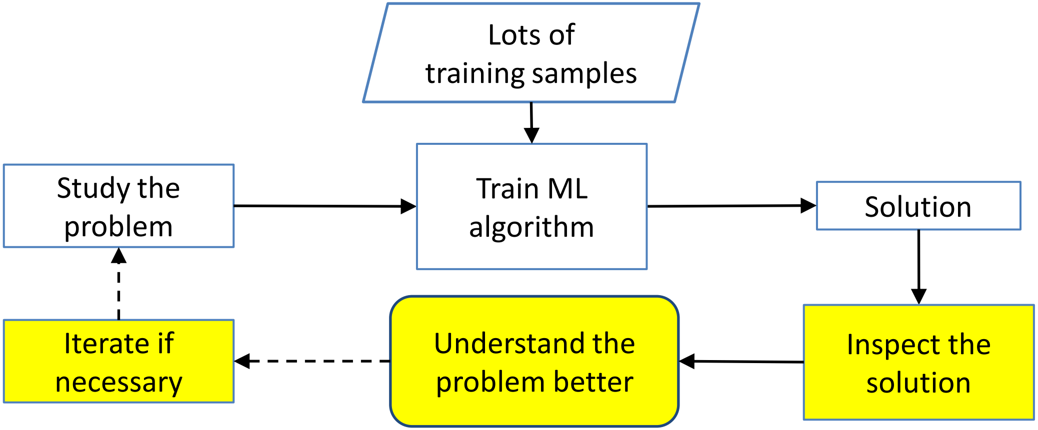
Why use machine learning?

- **Good for problems that evolves with time**
  - Machine learning can automatically rebuild the model when necessary
  - Example: spam classifier – learns new spam words when it become unusually frequent in spam flagged by users



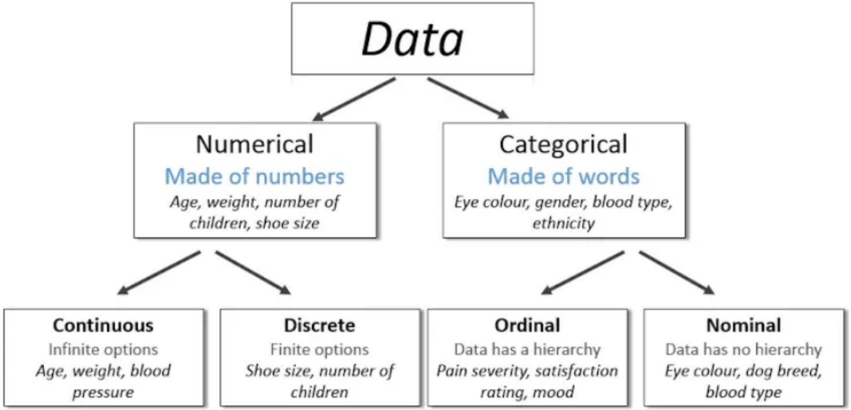
Why use machine learning?

- Help humans learn
  - Can inspect ML to see what they learn
    - may find unsuspected correlations or new trends
    - learn the problem better
    - Example: can examine the list of words or its combination that ML identifies as the best predictors of spam filter



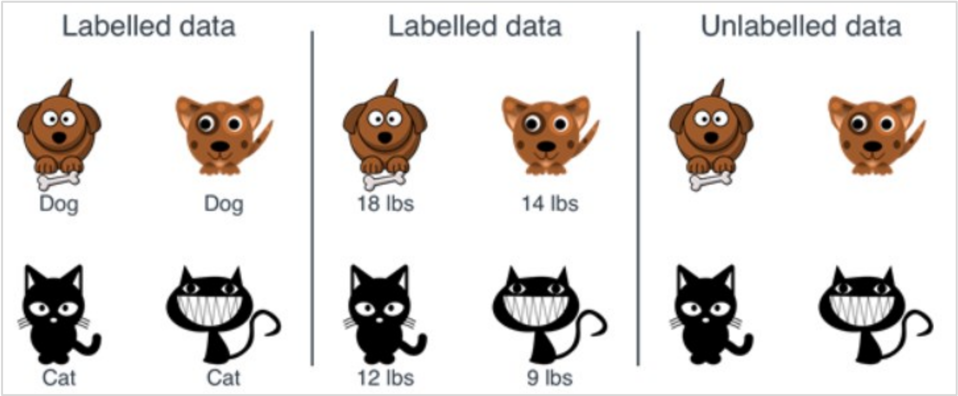
Data in Machine Learning

- **Numerical** (quantitative) data
  - Discrete (e.g. 1, 3, 8, -4, …)
  - Continue (e.g. 2.321, 0.2437, …)
- **Categorical** (qualitative) data
  - Ordinal (e.g. low, medium, high)
  - Nominal (e.g. red, blue, yellow)



Data in Machine Learning

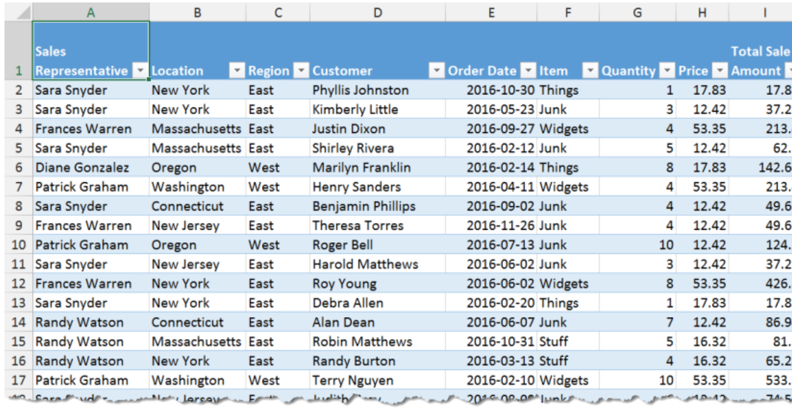
- **Labelled** data vs **unlabelled** data
  - Labelled data: data that comes with a tag (e.g. name, value)
  - Unlabelled data: data that comes with no tag



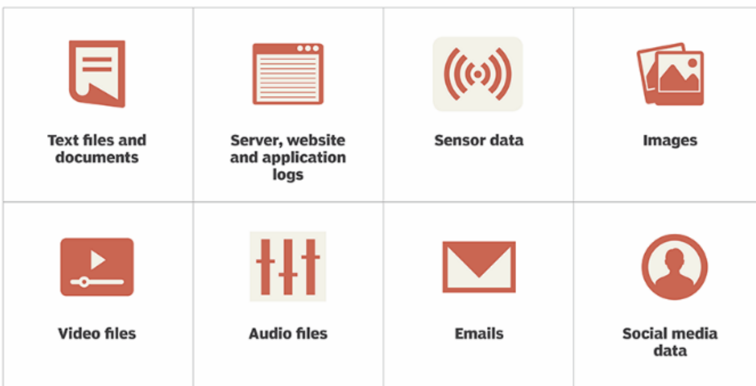
Structured Data vs Unstructured Data

Structured Data

- Specific and stored in a predefined format
- Suitable for traditional machine learning
- <u>Focused in this course</u>



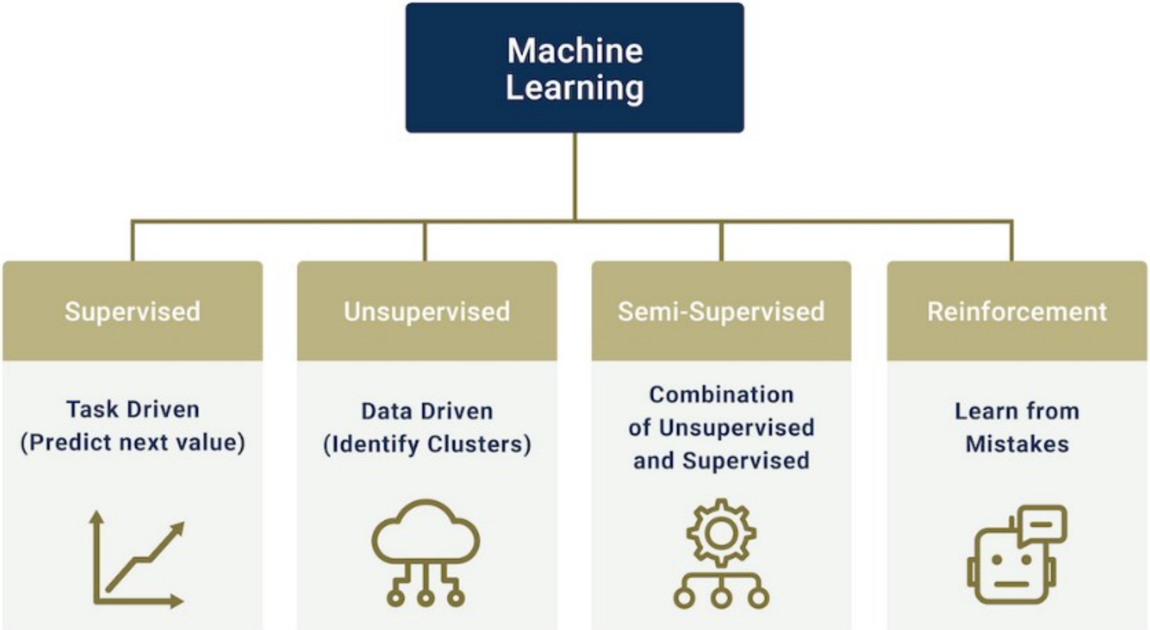
	A	B	C	D	E	F	G	H	I
1	Sales Representative	Location	Region	Customer	Order Date	Item	Quantity	Price	Total Sale Amount
2	Sara Snyder	New York	East	Phyllis Johnston	2016-10-30	Things	1	17.83	17.83
3	Sara Snyder	New York	East	Kimberly Little	2016-05-23	Junk	3	12.42	37.26
4	Frances Warren	Massachusetts	East	Justin Dixon	2016-09-27	Widgets	4	53.35	213.4
5	Sara Snyder	Massachusetts	East	Shirley Rivera	2016-02-12	Junk	5	12.42	62.1
6	Diane Gonzalez	Oregon	West	Marilyn Franklin	2016-02-14	Things	8	17.83	142.64
7	Patrick Graham	Washington	West	Henry Sanders	2016-04-11	Widgets	4	53.35	213.4
8	Sara Snyder	Connecticut	East	Benjamin Phillips	2016-09-02	Junk	4	12.42	49.68
9	Frances Warren	New Jersey	East	Theresa Torres	2016-11-26	Junk	4	12.42	49.68
10	Patrick Graham	Oregon	West	Roger Bell	2016-07-13	Junk	10	12.42	124.2
11	Sara Snyder	New Jersey	East	Harold Matthews	2016-06-02	Junk	3	12.42	37.26
12	Frances Warren	New York	East	Roy Young	2016-06-02	Widgets	8	53.35	426.8
13	Sara Snyder	New York	East	Debra Allen	2016-02-20	Things	1	17.83	17.83
14	Randy Watson	Connecticut	East	Alan Dean	2016-06-07	Junk	7	12.42	86.94
15	Randy Watson	Massachusetts	East	Robin Matthews	2016-10-31	Stuff	5	16.32	81.6
16	Randy Watson	New York	East	Randy Burton	2016-03-13	Stuff	4	16.32	65.28
17	Patrick Graham	Washington	West	Terry Nguyen	2016-02-10	Widgets	10	53.35	533.5



Unstructured Data

- Collection of varied types of data that are stored in their native formats (e.g. text, image, video, audio,…)
- Better result with deep learning techniques

Type of Machine Learning Algorithms



Labeled Data

Unlabeled Data

Mixed Data

No Training Data

Supervised Learning

- In supervised learning, the algorithm is given some example *input-output* pair and it *learns a function* that maps from *input* to *output*
- **input**: the set of **features** used to describe the samples
- **output**: the attribute we are interested to predict
- Example: Fruit classification

Input

Samples	Features		
	length	width	weight
fruit 1	165	38	172
fruit 2	218	39	230
fruit 3	76	80	145
fruit 4	145	35	150
…	…	…	…
fruit *m*	…	…	…

Function

Output

Label
Banana
Banana
Orange
Banana
…
…

Supervised Learning Tasks

Two main supervised learning tasks

Classification

Classification predicts **discrete** valued output (e.g., present/not present)

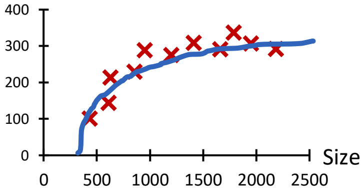
Yes

No

Object Detection (Images with Car)

Regression

Regression predicts **continuous** valued output (e.g., house price)



Housing Price Prediction

Supervised Learning – Classification Applications

Income classification:

- **Input**: numerical and categorical data
- **Output**: 1 (income<=50K), 0 (income>50K) – discrete
- **Features**: age, workclass, marital-status, race, education, area, …

age	workclass	marital-status	race	*class*
39	State-gov	Never-married	White	*<=50K*
49	Self-emp-inc	Married-civ-spouse	White	*>50K*
28	Private	Married-civ-spouse	Other	*<=50K*
35	Private	Divorced	White	*>50K*
38	Private	Divorced	White	*<=50K*
53	Local-gov	Never-married	White	*<=50K*
28	Private	Married-civ-spouse	Black	*<=50K*
37	Private	Married-civ-spouse	Black	*>50K*
37	Private	Married-civ-spouse	White	*<=50K*
49	Private	Married-spouse-absent	Black	*<=50K*
38	Federal-gov	Married-civ-spouse	White	*>50K*
42	Private	Married-civ-spouse	White	*>50K*

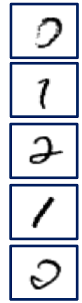
More Classification Applications

Digit Classification

- **Input**: images / pixel grids
- **Output**: a digit 0-9
- **Features**:
  - Signatures
  - Histogram of gradients
  - Shape Patterns: NumComponents, AspectRatio, NumLoops
  - …

Feature Extraction

Input (Image):



Output (discrete):

0

1

2

1

??

Spam mail classification

- **Input**: an email
- **Output**: *spam* or *non-spam*
- **Features**:
  - Words: FREE!, Earn, Call now,..
  - Text Patterns: $$$, ALL CAPS
  - Non-text: SenderInContacts
  - …

Input (Text):

Dear Andy,
How you are doing, buddy? Hopefully you are adapting well to your new school. All of us miss you dearly here. We miss your silly jokes.

Hello, I have a special offer for you…
WANT TO LOSE WEIGHT? The most powerful weight loss is now available without prescription.

Output (discrete):

Supervised Learning – Regression Applications

Predict the house price in the Boston area (regression):

- **Input:** numerical and categorical data
- **Output**: house price (in 1000usd) – continue
- **Features**:
  - CRIM: per capita crime rate by town
  - ZN: proportion of residential land
  - INDUS: proportion of non-retail business acres per town
  - CHAS: Charles River dummy variable (= 1 if near river; 0 otherwise)
  - NOX: nitric oxides concentration (parts per 10 million)
  - RM: average number of rooms per dwelling
  - AGE: proportion of units built prior to 1940
  - DIS: weighted distance to employment centres
  - RAD: index of accessibility to radial highways
  - TAX: full-value property-tax rate per $10,000
  - ...

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	Price
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1.0	296.0	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2.0	242.0	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2.0	242.0	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3.0	222.0	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3.0	222.0	18.7	396.90	5.33	36.2

A Simple Supervised Learning Example (1/6)

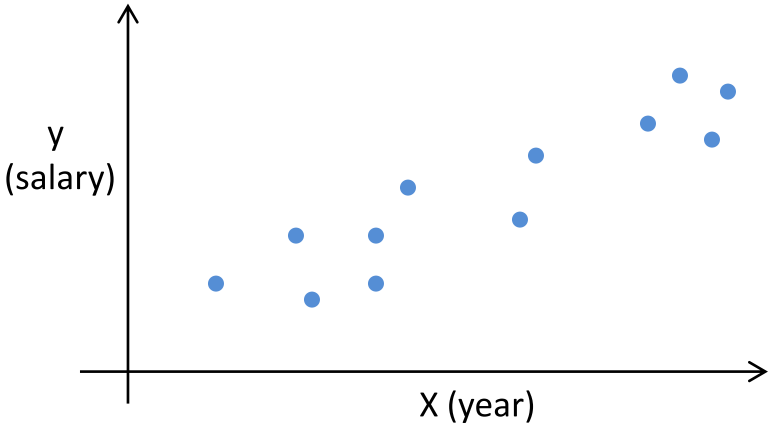
- Problem: want to predict IT salary
- Step 1: Collect Data
  - Consult domain expert/survey/research what key factors (features) affecting IT salary
    - Experience in years (used in this example)
    - Job title
    - Size of organization
    - Gender
    - Industry sector
    - Geographic region
    - ...
  - Collect the data
    - From survey/database
    - Data processing for missing/incomplete data, normalization (see L03)

Samples	
X (year)	**y (salary, k)**
5	38
10	64
7	36
8	44
0	na
…	…

A Simple Supervised Learning Example (2/6)

- Step 2: Model selection
  - Study the data and determine what model is suitable
    - For this plot, select linear regression

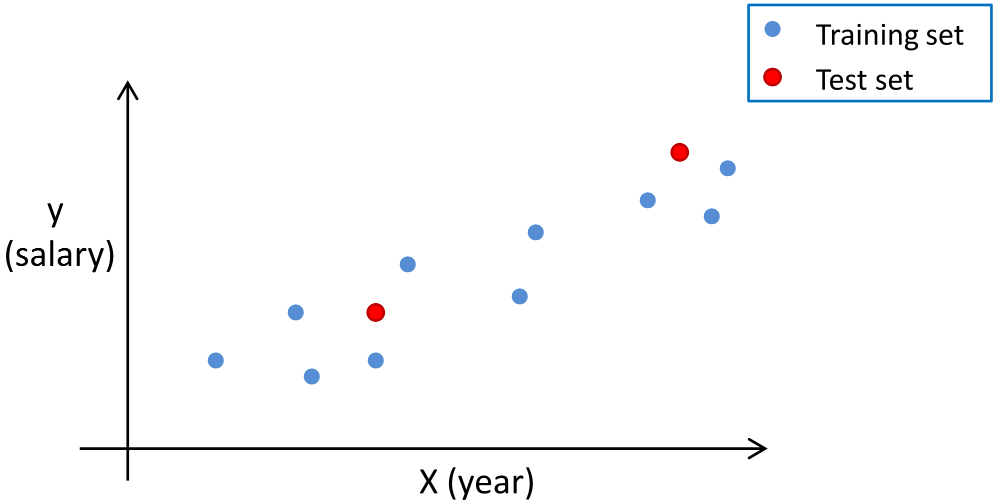
$$y = \theta_0 + \theta_1 \times x \qquad (\theta_0, \theta_1) = \text{model parameters}$$



ChatGPT has 175 billion parameters, GPT4 > 1 trillion parameters

A Simple Supervised Learning Example (3/6)

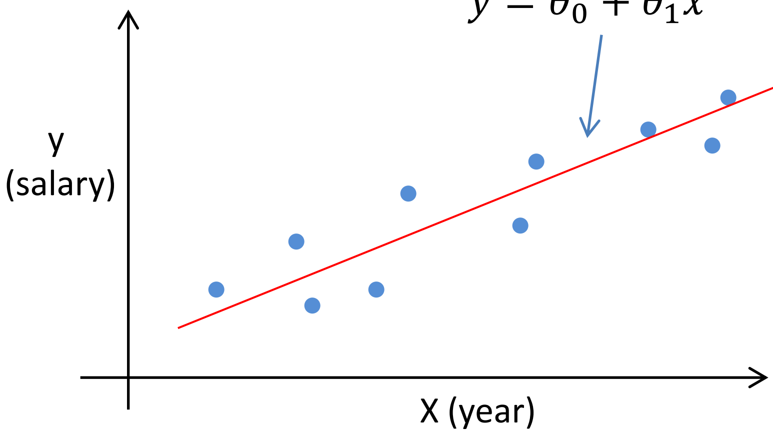
- Step 3: Train model
  - Separate data into training set (80%) and test set (20%)
    - Train model on training set, validate model using test set

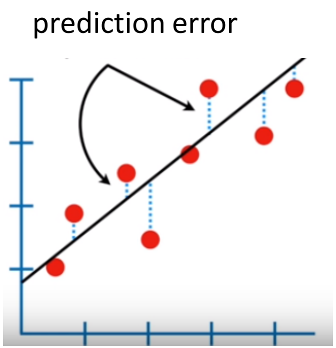


A Simple Supervised Learning Example (4/6)

- Step 3: Train model (cont.)
  - Train model with training set
    - Use normal equation or gradient descent (see L05)
    - Minimize sum-of-squared error (SSE) – training error

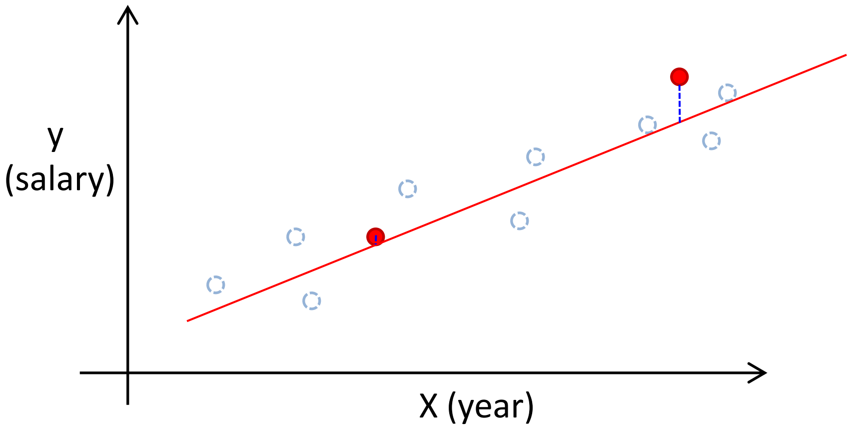
$$y = \hat{\theta}_0 + \hat{\theta}_1 x$$





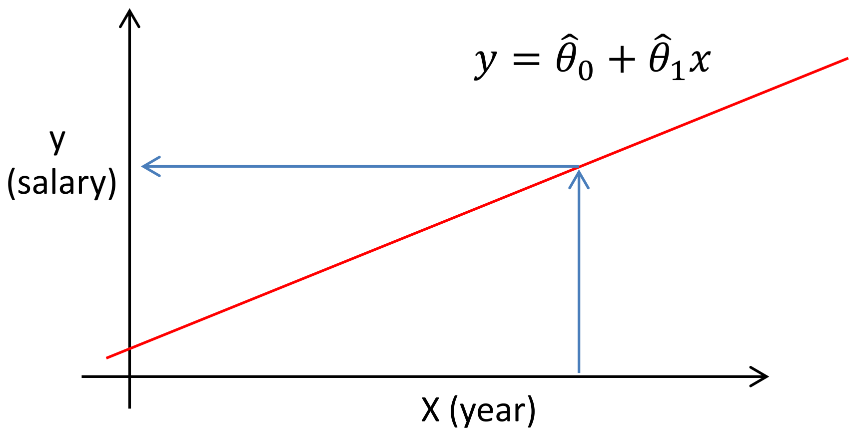
A Simple Supervised Learning Example (5/6)

- Step 4: Validate model
  - Validate the model using test set – test error



A Simple Supervised Learning Example (6/6)

- Step 5: Deploy model
  - Use the trained model for prediction



Supervised Learning Algorithms

- Algorithms for classification:
  - k-Nearest neighbour (k-NN)
  - Logistic Regression
  - Decision Tree and Random Forests
  - Support Vector Machine (SVM)
  - Neural Networks (NN)
  - ...

- Algorithms for regression:
  - K-NN Regressor
  - Linear Regression
  - Decision Tree and Random Forests
  - SVM Regressor
  - NN
  - Non-linear Regression ...

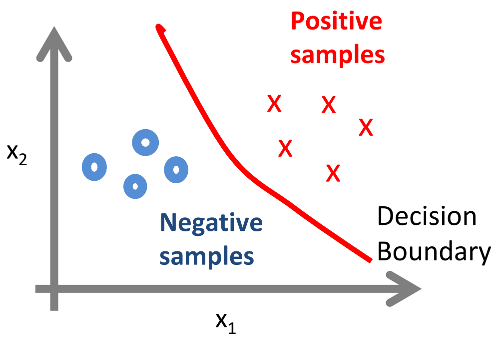
Unsupervised learning

Unsupervised Learning

- <u>No labels </u>are provided for all training samples
- Discovers the underlying structure, relationship or patterns based only on the features of the training sample

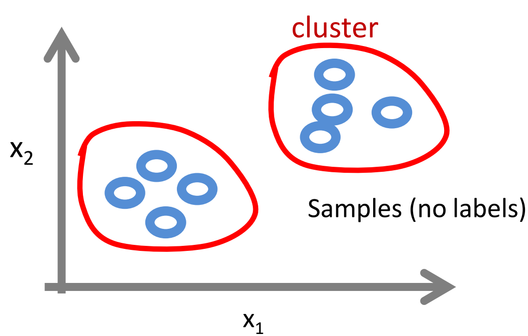
	Features		
	length	width	weight
fruit 1	165	38	172
fruit 2	218	39	230
fruit 3	76	80	145
fruit 4	145	35	150
…	…	…	…
fruit *m*	…	…	…

Supervised Learning



VS

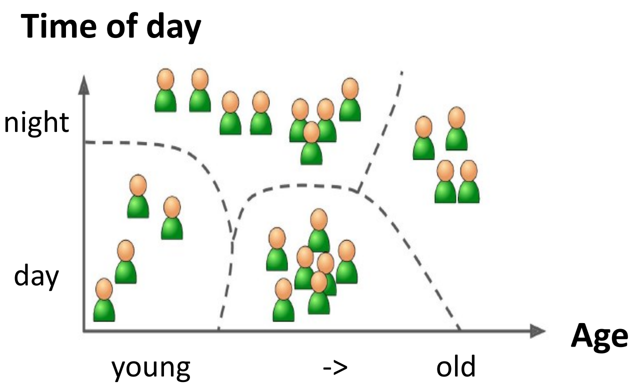
Unsupervised Learning



Unsupervised Learning Task: Clustering

- Detect groups of similar samples
- Example:

Detecting groups of visitors who visit your blog



Example analysis:

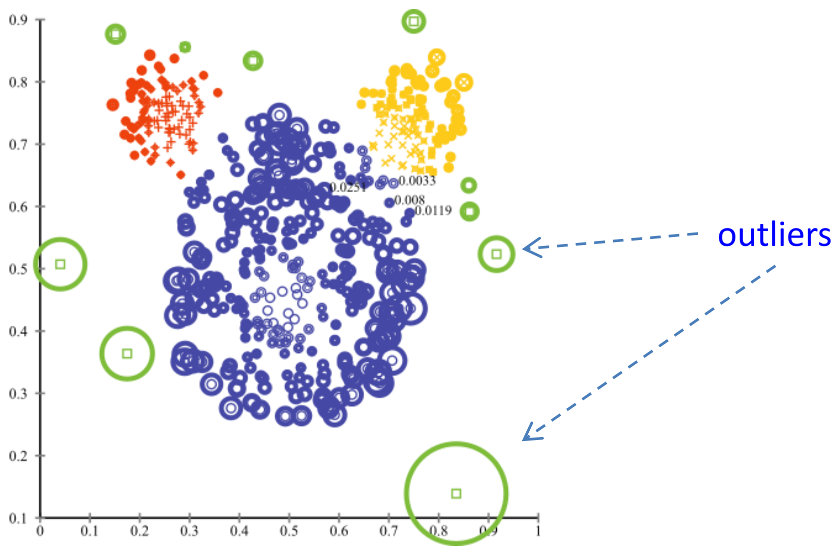
- 40% visitors who love comic books and read in the evening
- 20% are young sci-fi lovers who visit before school, etc.

How does it help?

- Can target your posts for each group

Unsupervised Learning Task: Anomaly detection

- Identify items, events or observations which do not conform to an expected pattern or other items in a dataset
- **Anomalies** are also referred to as **outliers**, **novelties** or **noise**



Applications: Bank fraud, medical problems or errors in a text.

Unsupervised Learning Task: Association Rule Mining

- Given a set of transactions, find **rules** that will predict the occurrence of an item based on the occurrences of other items in the transaction.

TID	Items
1	Bread, Peanuts, Milk, Fruit, Jam
2	Bread, Jam, Soda, Chips, Milk, Fruit
3	Steak, Jam, Soda, Chips, Bread
4	Jam, Soda, Peanuts, Milk, Fruit
5	Jam, Soda, Chips, Milk, Bread
6	Fruit, Soda, Chips, Milk
7	Fruit, Soda, Peanuts, Milk
8	Fruit, Peanuts, Cheese, Yogurt

Support(Bread) = #Bread / #total
Support(Jam) = #Jam / #total
Support(Bread, Jam) = #(Bread+Jam) / #total

Confident(Bread->Jam) = #(Bread+Jam) / #Bread
Lift(Bread->Jam) = Support(Bread, Jam) / (Support(Bread) x Support(Jam))

Unsupervised Learning Tasks and Algorithms

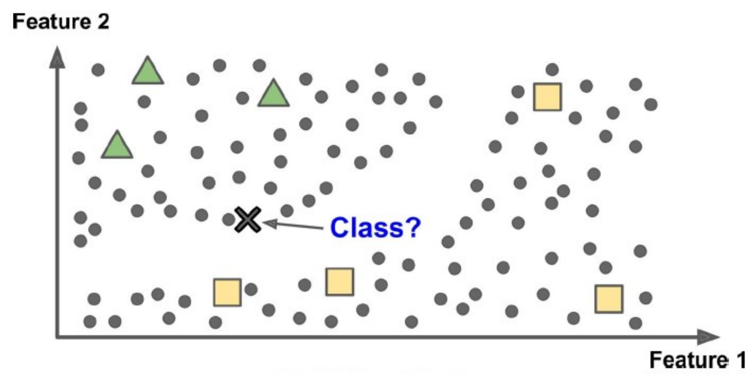
- Clustering
  - k-Means
  - Hierarchical Cluster Analysis (HCA)
  - Expectation Maximization
- Association rule learning
  - Apriori
  - Eclat

Semi-supervised learning

- Partially labeled training data. Typically more unlabeled data than labeled

- Most semi-supervised algorithms are combinations of *unsupervised* algorithms and *supervised* algorithms

	Features			Label
	length	**width**	**weight**	
fruit 1	165	38	172	Banana
fruit 2	218	39	230	?
fruit 3	76	80	145	Orange
fruit 4	145	35	150	?
…	…	…	…	…
fruit *m*	…	…	…	…

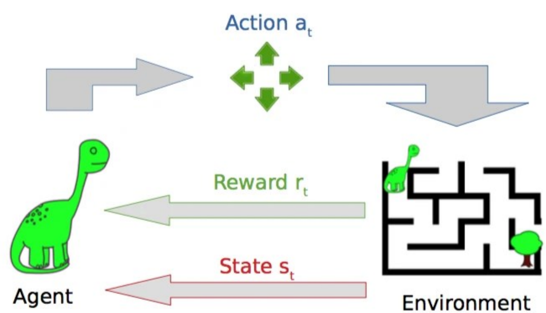


Reinforcement Learning

- No training set is provided
- Learns based on the feedback of the environment:
  - effect of the agent' action on the environment (**state**)
  - **rewards** of taking a particular action
- Learns by itself the best **policy** (state-action) to get the most reward over time

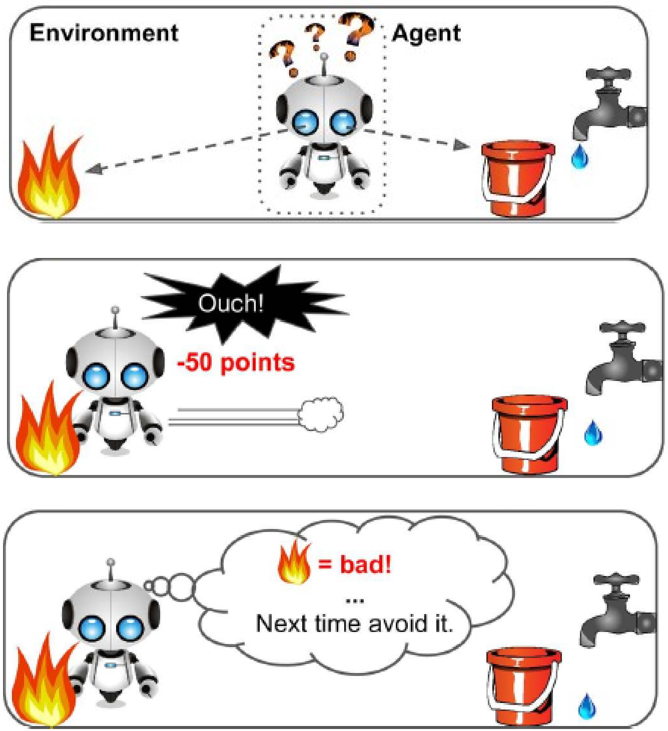
Policy

Q Table	
State-Action	Value
-	0
-	0
-	0
-	0
-	0
-	0
-	0
-	0
-	0



Applications: games (chess, go, video), robotics, traffic control, trading,...

Reinforcement Learning Example



The agent

1. observes the environment
2. Select action using policy
3. Perform action
4. Get reward or penalty
5. Update policy (learning step)
6. Iterate until an optimal policy is found

Differences between Machine Learning Types

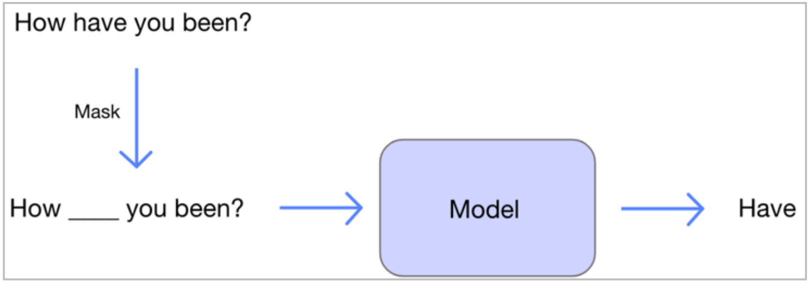
Supervised Learning	Unsupervised Learning	Reinforcement Learning
Labeled data with output specified	Unlabeled data, output not specified	Environment with rewards and penalty
Solves problems by mapping input to known output	Solves problems by discovering underlying patterns	Solves problems by trial and error
External supervision	No supervision	No supervision
Used for regression and classification tasks	Used for clustering and association tasks	Used for control and decision making tasks

Foundation Model & Self-Supervised Learning

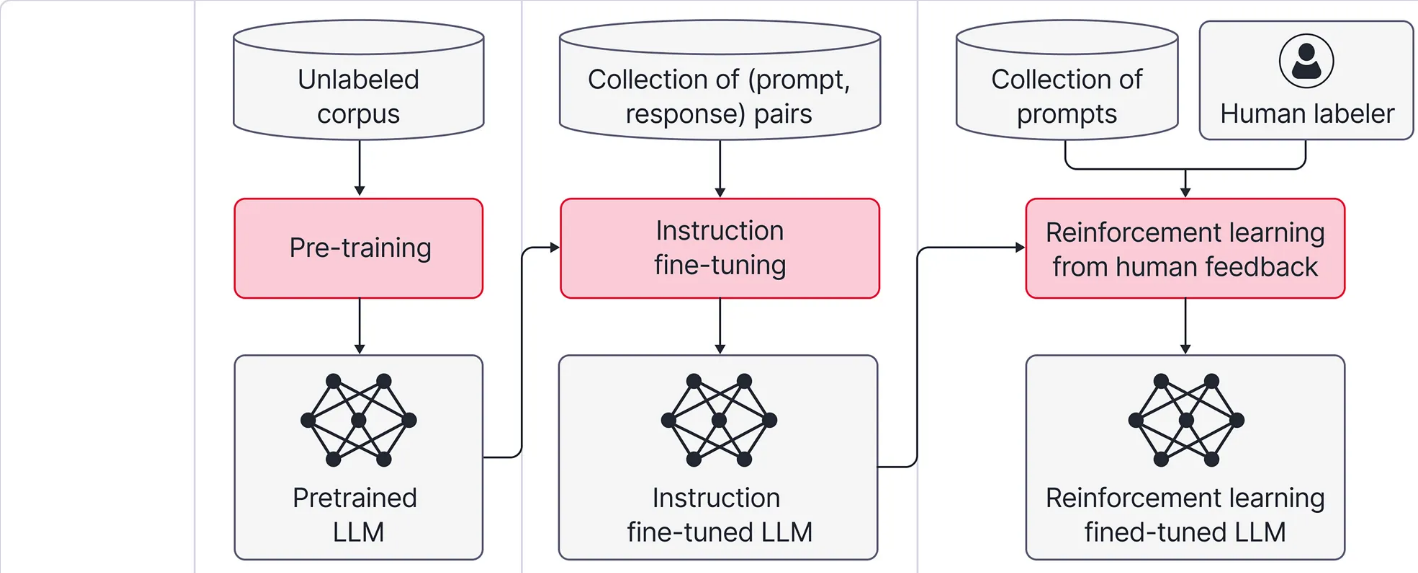
- Conventionally, a AI model is trained on task-specific data to perform very specific task.
- A new paradigm in AI has emerged called **foundation models**. Unlike traditional AI, foundation models learn from massive datasets across different domains.
- Through **self-supervised learning** techniques, a foundation model teach itself to acquire broad scope of knowledge and understanding of the world (general intelligence).
- Large language models (LLM) such as OpenAI's **GPT-4** and Google's **PaLM** are examples of foundation models .
- The foundation models can then be transferred to perform any other tasks through fine-tuning or prompting.
  - GPT -> ChatGPT, GPT -> Copilot, GPT -> Duolingo
- Foundation models require a lot more data and computing power to train.

Self-Supervised Learning

- **Self-supervised learning** is a new machine learning process where the model **trains itself** to **learn one part of the input from another part** of the input to obtain **useful representations** and **knowledge.**
- The trained model can help with **downstream learning tasks**.



How LLMs are trained



Dataset size	x00 billions to 1.x trillion tokens	~xk to x0k (prompt, response)	~x0k prompt
Example of models	GPT-3, LLaMA, Falcon, BLOOM	Dolly-v2, Falcon-instruct	Claude, GPT-4, ChatGPT
	(a)Pre-training Self-supervised learning	(b) Instruction fine-tuning	(c) Reinforcement learning from human feedback

Source: Borealis AI

Challenges of Machine Learning

- Insufficient quantity of training data
- Poor data quality
- Non-representative training data
- Irrelevant features
- Underfitting & overfitting the training data

Poor-quality data

- Training data may contain errors, for example:

#	Id	Name	Birthday	Gender	IsTeacher?	#Students	Country	City
1	111	John	31/12/1990	M	0	0	Ireland	Dublin
2	222	Mery	15/10/1978	F	1	15	Iceland	
3	333	Alice	19/04/2000	F	0	0	Spain	Madrid
4	444	Mark	01/11/1997	M	0	0	France	Paris
5	555	Alex	15/03/2000	A	1	23	Germany	Berlin
6	555	Peter	1983-12-01	M	1	10	Italy	Rome
7	777	Calvin	05/05/1995	M	0	0	Italy	Italy
8	888	Roxane	03/08/1948	F	0	0	Portugal	Lisbon
9	999	Anne	05/09/1992	F	0	5	Switzerland	Geneva
10	101010	Paul	14/11/1992	M	1	26	Ytali	Rome

Missing values

Invalid values

Misfielded values

Outlier

Uniqueness

Formats

Attribute dependencies

Misspellings

- **Data cleaning**: Most data scientists spend a significant time to clean the data. For example:
  - Fill up missing value
  - Drop a column (feature) with many missing values/errors
  - Remove rows (samples) with outliers
  - Fix error/format manually

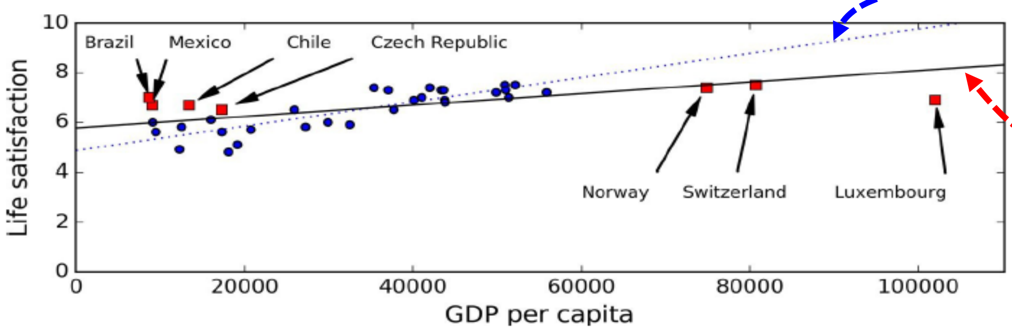
Non-representative data

- Training data should be representative of the new cases that you want to generalize to.
- Consider fitting a linear model to the GDP dataset with and without 7 missing countries :

(This model does not generalize well)

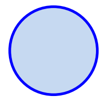
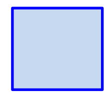
without missing countries

with missing countries

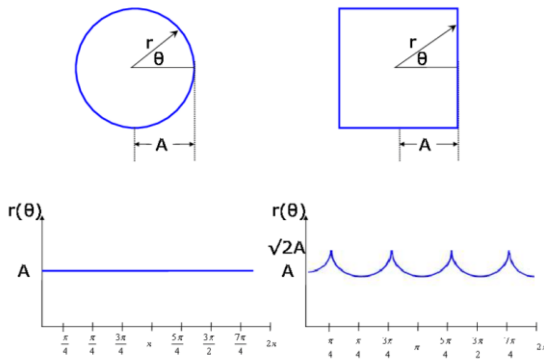


Irrelevant features

- Selected features must be relevant to the task at hand. Having irrelevant features in your data can decrease the accuracy of the models.
    - For example, *area* or *perimeter length* are irrelevant feature for classifying shapes like circle and rectangle.

f(area) =  ? 

    - Features such as *signature*, *number of corners* are more suitable for classifying shapes.



Feature Engineering

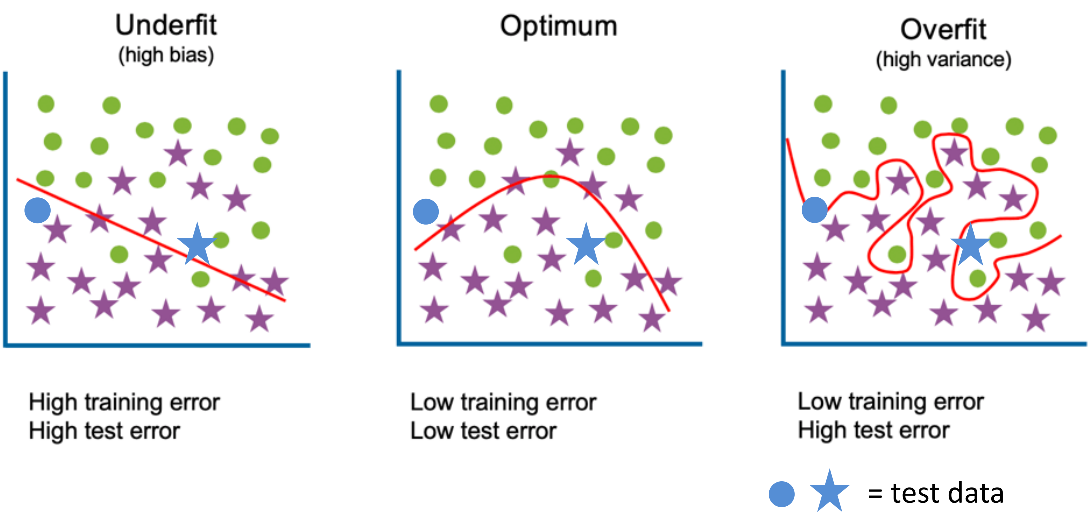
- **Feature engineering** is the process to come up with a good set of features to train on. It involves:
  - **feature extraction** – use some tools to extract features from samples (e.g., extract shape signature, colors from an object).
  - **feature selection** – choose the most useful features among all existing features that produce the best result for a machine learning model.

Deep learning learns features automatically but requires lots of training data.

Underfitting & Overfitting

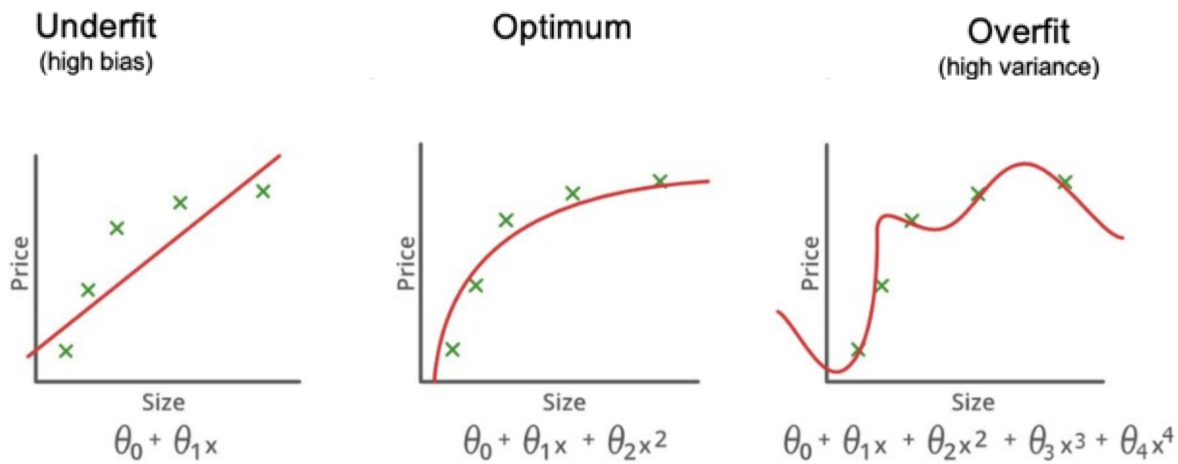
- **Underfitting (high bias)** may happen when our model is over-simplified or not **expressive** enough (high training error and high test error).

- **Overfitting (high variance)** may happen when our model is too complex and fits too specifically to the training set, but it does not generalize well to new data (low training error but high test error).



Underfitting & Overfitting

- Underfitting and overfitting in regression task

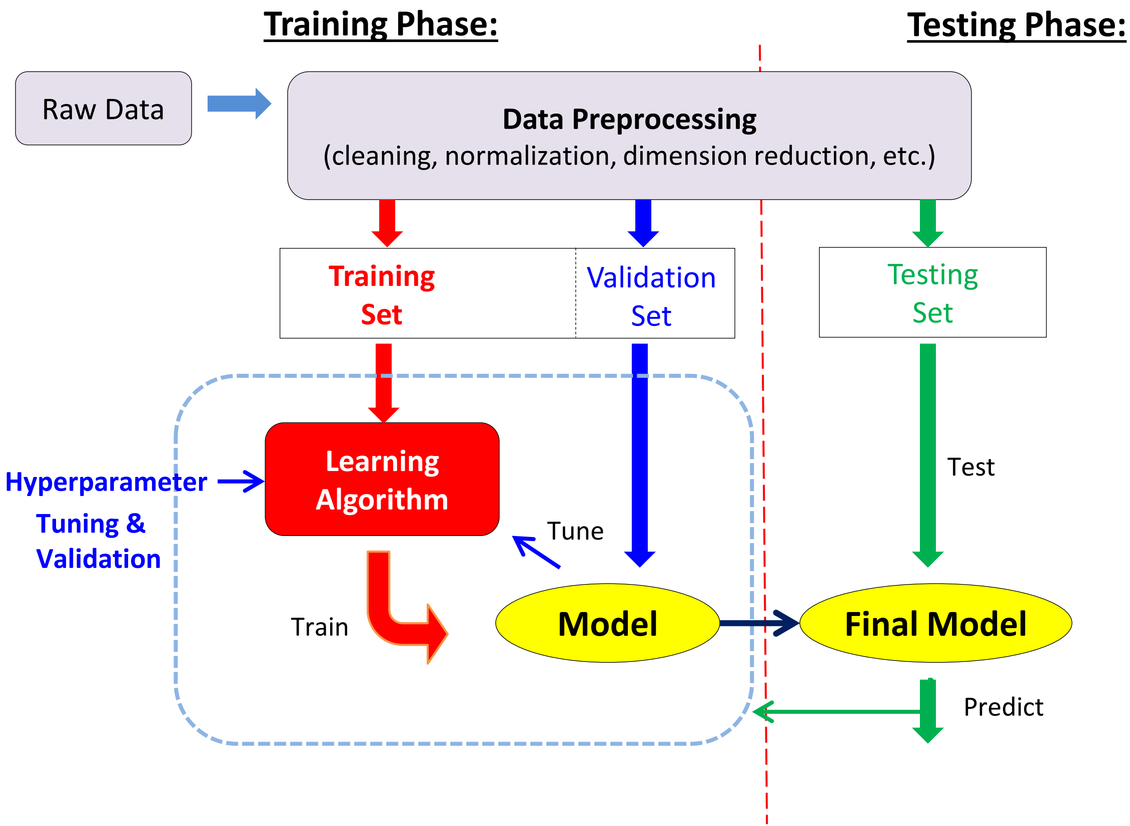


Hyperparameter Tuning

- **Hyperparameter** is a parameter whose value is set before the learning process begins and which is used to control the learning process.
  - For example, % of train-test split is a hyperparameter
- Different model has different set of hyperparameters. For example,
  - Polynomial model: *degree*
  - K-NN: n_*neighbors*, distance metric (Manhattan or Euclidean)
  - Neural Networks: α (learning rate), *max_iter* (maximum iterations)
  - SVM: *kernel* (linear, rbf), C (penalty parameter)
- In machine learning, **hyperparameter tuning** is the process of choosing a set of <u>optimal</u> hyperparameters for a learning algorithm.
- Need to balance between fitting the data perfectly and keeping the model simple to ensure it generalizes well (avoid underfit & overfit)
- Hyperparameter tuning is an important step of building a machine learning system.

The Machine Learning Framework

- Divided into two phases



Next:

The Regression Pipeline