Machine Learning

Lecture 1

Introduction to ML

Associate:wafaa Shalash

Fall 2026

Introduction to ML

AI and machine learning.

What is machine learning?

Machine learning process

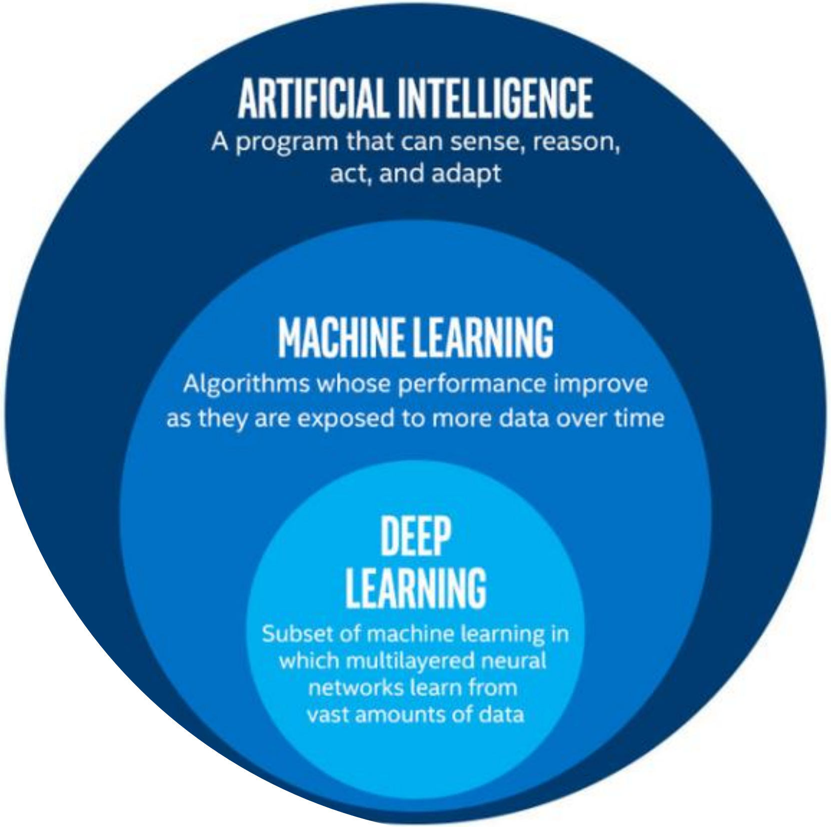
Data Model and Training

Resources

- *“Machine Learning: An Algorithmic Perspective”, Stephen Marsland, 2nd edition, 2015.*
- *“Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, And Techniques To Build Intelligent Systems”, 2nd edition, Aurélien Géron, 2019.*
- *“Data Science & Big Data Analytics, Discovering, Analyzing, Visualizing and Presenting Data”, EMC Education Services, Wiley, 2015.*
- *Research papers*
- *Related web Sites*
- *Course Web site*

What Is Artificial Intelligence?

- The term artificial intelligence first appeared in the 1950s to describe systems comprising a set of human-defined, if/then decision rules—which have always been easily broken and hard to maintain.
- Machine learning (ML) is AI's brain —a type of algorithm that enables computers to analyze data, learn from past experiences, and make decisions, all in a way that resembles human behavior.



ARTIFICIAL INTELLIGENCE

A program that can sense, reason, act, and adapt

MACHINE LEARNING

Algorithms whose performance improve as they are exposed to more data over time

DEEP LEARNING

Subset of machine learning in which multilayered neural networks learn from vast amounts of data

AI History

late 50’s

The main concept of AI was introduced only in the late 50’s. In ‘56, a young assistant professor of mathematics at Dartmouth College reunited a group of scientists in order to discuss ideas about the "thinking machines".

1997

In 1997 Kasparov v Deep Blue

2018

In 2018, Google's CEO, Sundar Pichai, played a recorded phone call where Google's Assistant, Duplex, schedules a hair salon appointment and books a restaurant.

Artificial Intelligence

- Engineering of making intelligent machine and program
- Starts 1950s

Machine Learning

- Ability of learn without being explicitly programmed
- Needs hand crafted features.
- Moderate data size.
- Moderate computing power.
- Shows success 1980s

Deep Learning

- Learn based on neural networks
- Self extracting features.
- Needs huge data.
- High computing power (GPU,HPC)
- Arises 2000s

What is machine learning?

- **Machine Learning (ML) is a scientific field that gives computers the ability to learn without being explicitly programmed for a specific task.** It's the process of using algorithms to parse data, learn from that data, and then make a determination or prediction about something in the world. Instead of hand-coding a software routine with specific instructions, a model is trained on a large amount of data to recognize patterns and make data-driven decisions.

What is machine learning?

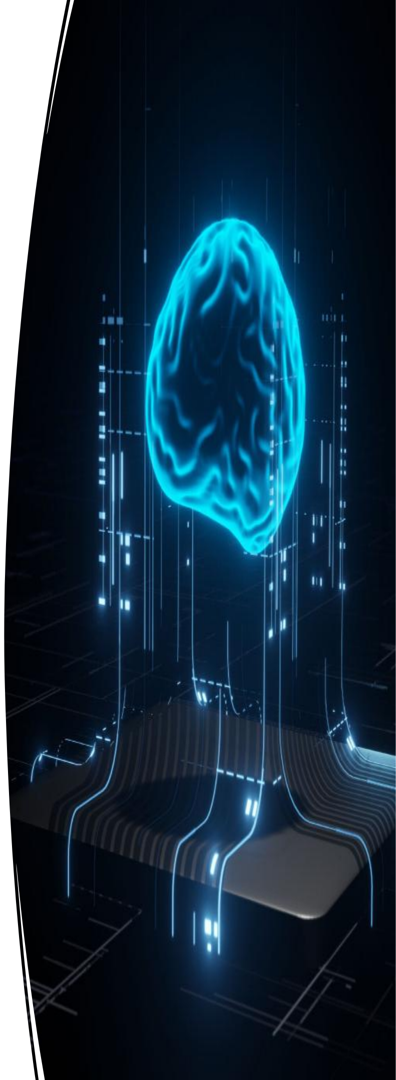
- **Machine Learning** is a branch of Artificial Intelligence (AI) that enables systems to **learn from data** and **improve performance** on a task **without being explicitly programmed**.

- Detecting patterns and regularities with a good and generalizable approximation (“model” or “hypothesis”)

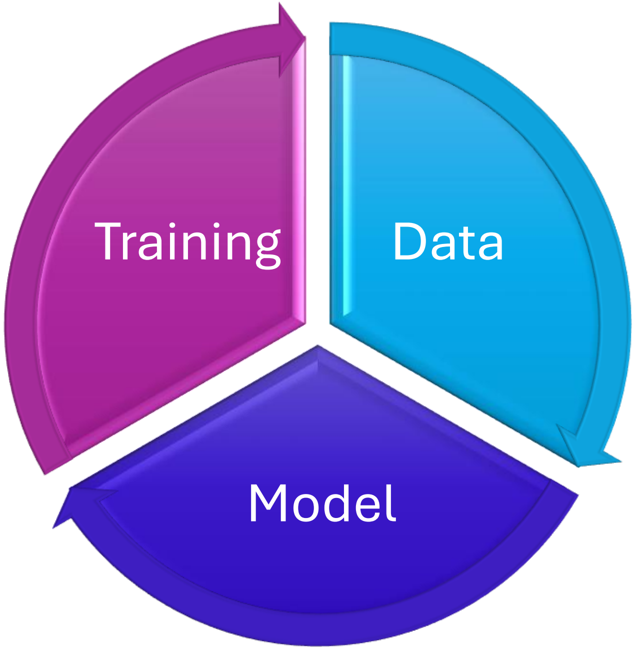
- Execution of a computer program to optimize the parameters of the model using training data or past experience.

1. What is machine learning?

- machine learning is a family of statistical and mathematical modelling techniques that uses a variety of approaches to automatically learn. And improve the prediction of a target objective without explicit programming.
- systems that improve their performance in a given task through exposure to experience or data.



2. Elements of Machine Learning



2.1 Data

- **Data is the Fuel:** ML algorithms require data—lots of it. This data can be numbers, text, images, audio, etc.
- All learning methods are data driven. Sets of data are used to train the system. These sets may be collected and edited by humans or gathered autonomously by other software tools. Control systems may collect data from sensors as the systems operate and use that data to identify parameters or train the system.

2.2 Model

- **The ML Algorithm:** The algorithm's job is to find underlying patterns, relationships, or trends within that data.
- **The Model is the Recipe:** The output of the learning process is a model. This model is a mathematical file that has learned from the data and can now be used to make predictions on new, unseen data.
- **Models** are often used in learning systems. A model provides a mathematical framework for learning. A model is human-derived and based on human observations and experiences. For example, a model of a car, seen from above, might be that it is rectangular with dimensions that fit within a standard parking spot. Models are usually thought of as human-derived and provide a framework for machine learning. However, some forms of machine learning develop their models without a human-derived structure.

2.3 Training

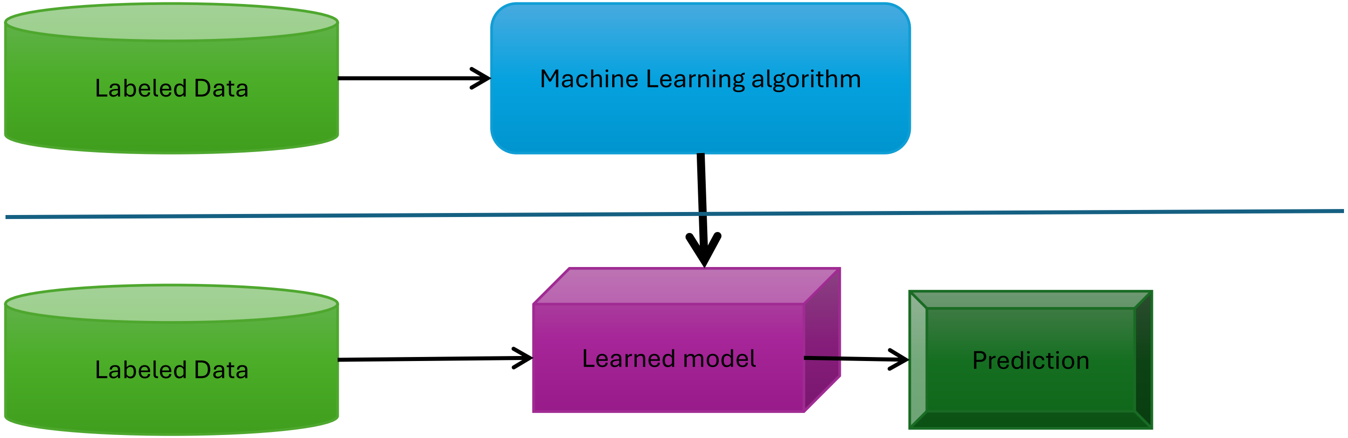
- Training or Learning, Not Programming: The system improves with experience (more data) rather than through direct human coding of the solution.

2.3 Training

- **A system which maps an input to an output needs training** to do this in a useful way. Just as people need to be trained to perform tasks, machine learning systems need to be trained. Training is accomplished by giving the system an input and the corresponding output and modifying the structure (models or data) in the learning machine so that mapping is learned. In some ways, this is like curve fitting or regression.
- If we have enough training pairs, then the system should be able to produce correct outputs when new inputs are introduced.
- For example, if we give a face recognition system thousands of cat images and tell it that those are cats, we hope that when it is given new cat images it will also recognize them as cats.
- Problems can arise when you don't give it enough training sets, or the training data is not sufficiently diverse, for instance, identifying a long-haired cat or hairless cat when the training data is only of short-haired cats. A diversity of training data is required for a functioning algorithm.

Machine Learning Basics

Machine learning is a field of computer science that gives computers the ability to learn without being explicitly programmed



Methods that can learn from and make predictions on data

3. Key Steps in Machine Learning Workflow (from the diagram):

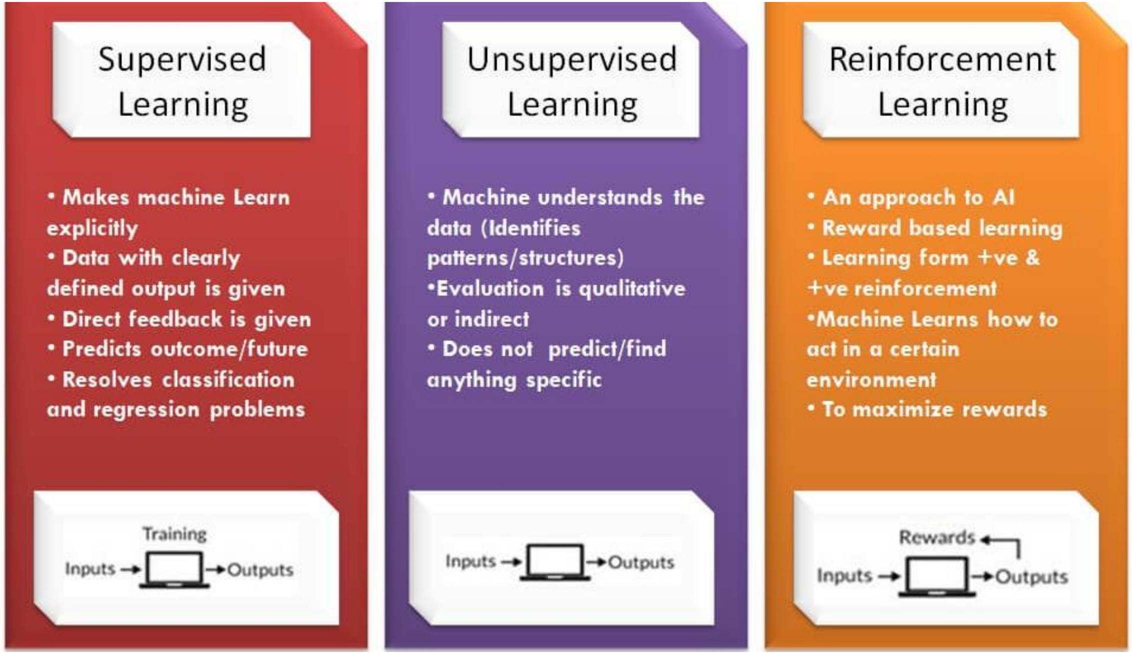
- **Input Data**
- Raw data is collected from various sources.
- **Preprocessing**
- Data is cleaned, normalized, and transformed to prepare it for learning.
- May involve handling missing values, converting categories to numbers, etc.
- **Feature Extraction**
- Important characteristics (features) are extracted from the data to represent it effectively.
- **Model Selection**
- Choose a suitable algorithm (e.g., Decision Tree, SVM, Neural Network) depending on the task (classification, regression, etc.).
- **Training**
- The model learns patterns from the training dataset by adjusting internal parameters.
- **Testing**
- The model is evaluated on new/unseen data to assess its performance.
- **Prediction**
- Once trained and validated, the model can predict outcomes on new inputs.
- **Feedback Loop (optional)**
- Performance feedback can be used to fine-tune the model over time.

Main Types of Machine Learning

- There are three primary categories, defined by the kind of "signal" or "feedback" available to the learning system.

Type	How It Learns	Simple Example
1. Supervised Learning	Learns from **labeled data** (data that already contains the correct answer). The goal is to learn a mapping from input to output.	**Spam Filtering:** You show the algorithm many emails pre-labeled as "spam" or "not spam." It learns the patterns and can label new emails.
2. Unsupervised Learning	Finds hidden patterns in **unlabeled data** (data without any predefined answers). The goal is to discover the underlying structure.	**Customer Segmentation:** You give the algorithm customer purchase data, and it groups customers into clusters based on shopping habits, without you telling it what the groups are.
3. Reinforcement Learning	An **agent** learns by interacting with an **environment** and receiving **rewards** or **penalties** for its actions. It learns the best strategy (policy) to maximize long-term reward	**Learning to Play a Game:** A computer program plays a video game. It gets a reward for gaining points and a penalty for losing a life. Through trial and error, it learns the best moves to win.

Types of Machine learning



Supervised Learning

- Makes machine Learn explicitly
- Data with clearly defined output is given
- Direct feedback is given
- Predicts outcome/future
- Resolves classification and regression problems

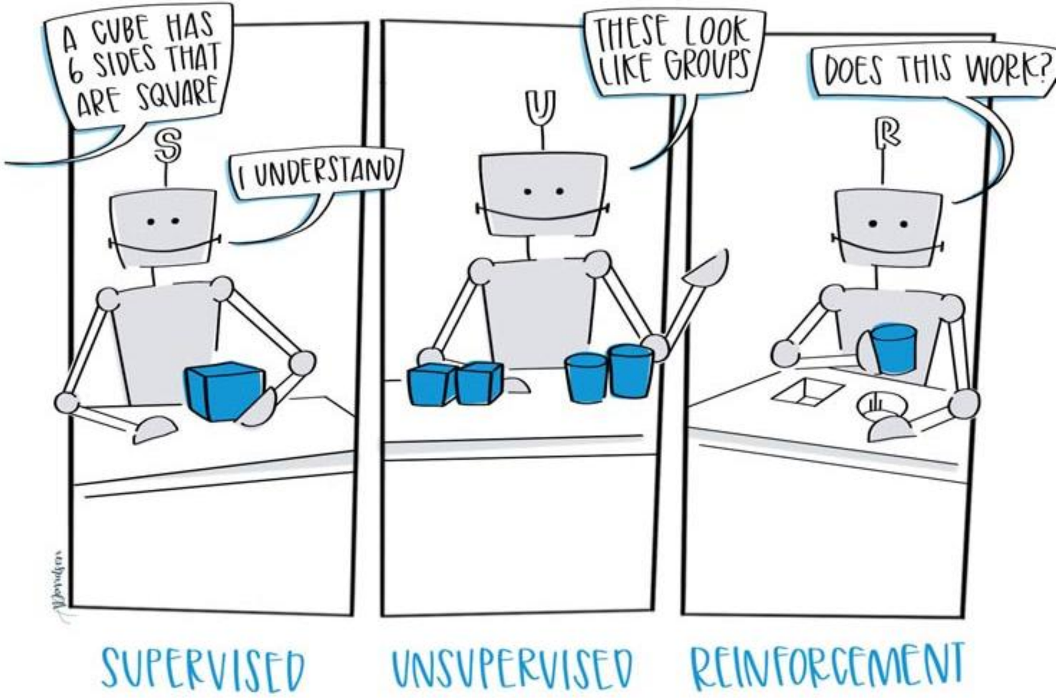
Unsupervised Learning

- Machine understands the data (Identifies patterns/structures)
- Evaluation is qualitative or indirect
- Does not predict/find anything specific

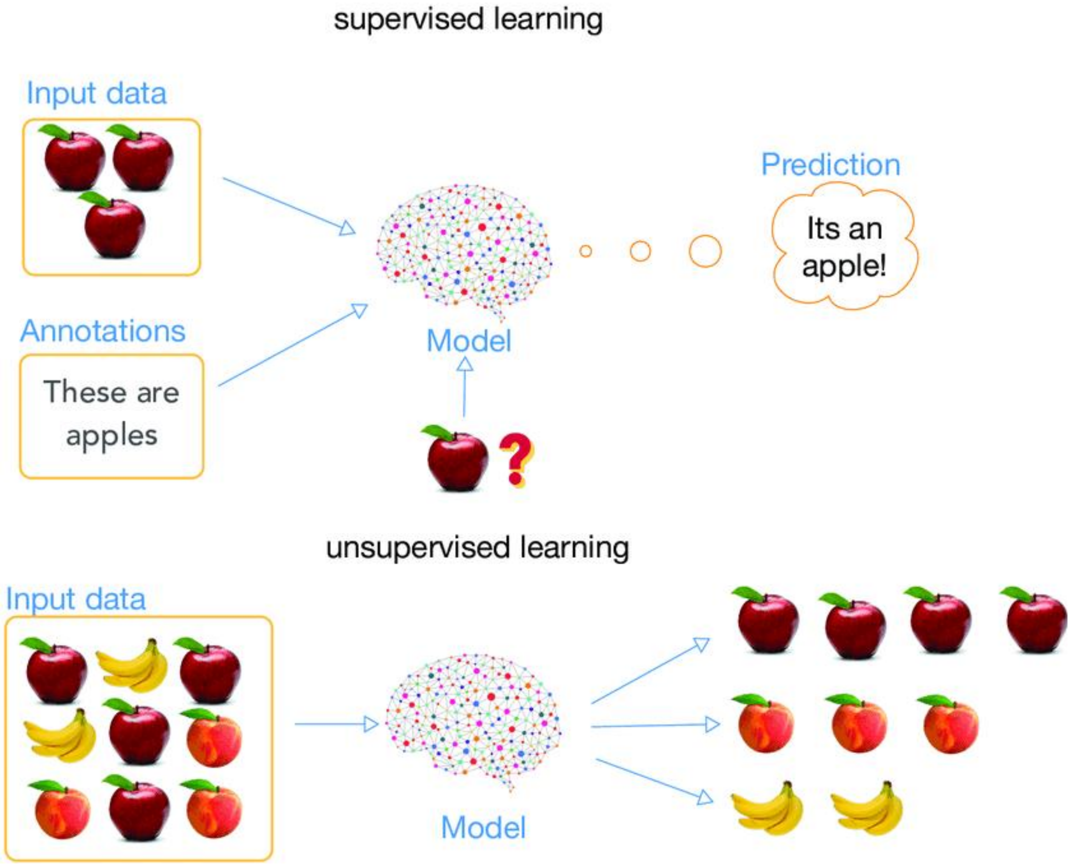
Reinforcement Learning

- An approach to AI
- Reward based learning
- Learning form +ve & +ve reinforcement
- Machine Learns how to act in a certain environment
- To maximize rewards

Types of Machine learning



Supervised learning Vs Unsupervised learning



Machine learning Tasks

- Loan application classification
- Signature recognition
- Voice recognition over phone
- Credit card fraud detection
- Spam filter
- Suggesting other products at Amazon.com
- Stock market prediction
- biometric identification (fingerprints, DNA, iris scan, face)

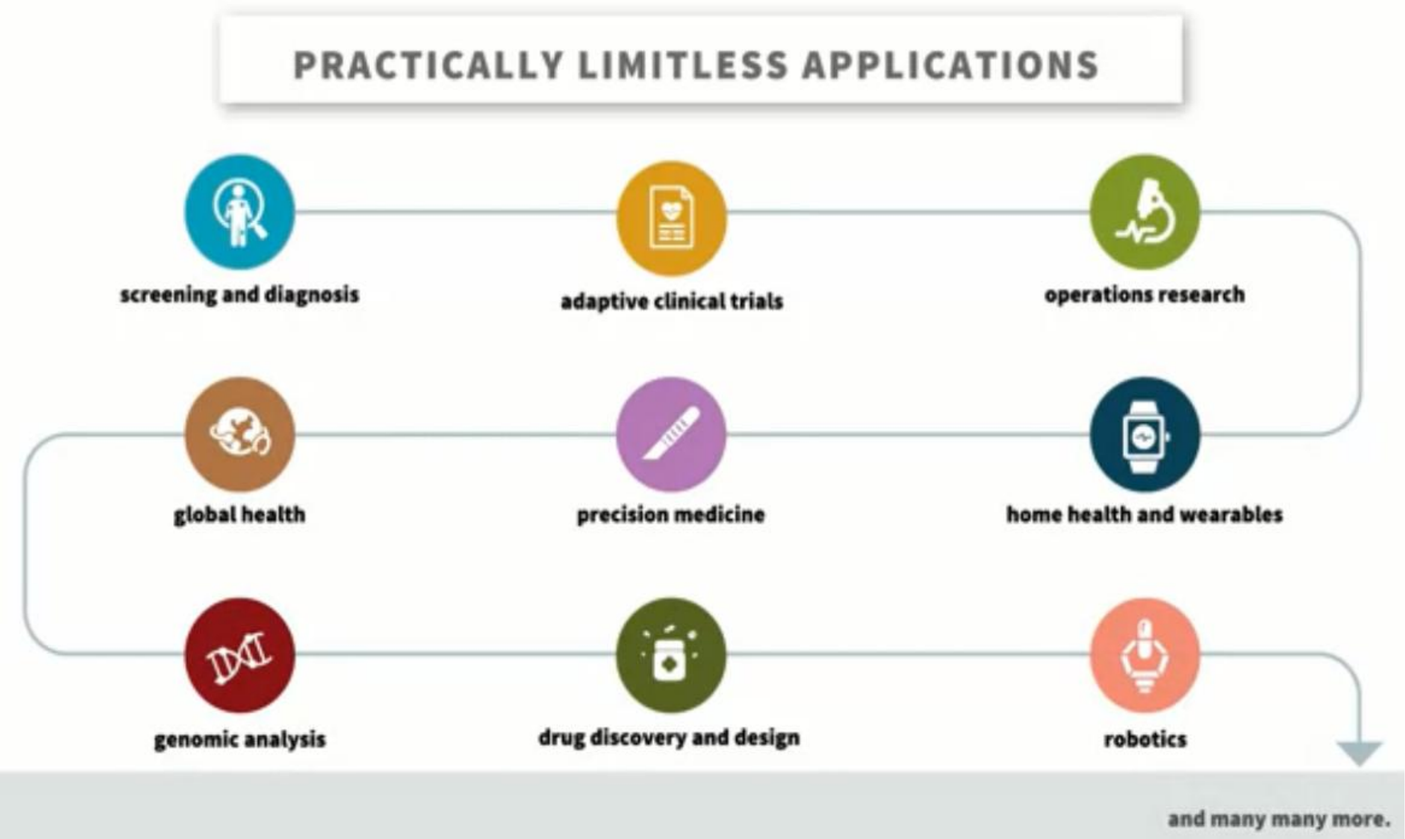
- Expert level chess and checkers systems
- machine translation
- web-search
- document & information retrieval
- camera surveillance
- robo_soccer
- ....

Machine learning Tasks in Medicine

- Identifying Diseases and Diagnosis. ...
- Drug Discovery and Manufacturing. ...
- Medical Imaging Diagnosis. ...
- Personalized Medicine. ...
- Machine Learning-based Behavioral Modification. ...

- Smart Health Records. ...
- Clinical Trial and Research. ...
- Crowdsourced Data Collection
- Better Radiotherapy
- Outbreak Prediction
- ....

- https://www.flatworldsolutions.com/healthcare/articles/top-10-applications-of-machine-learning-in-healthcare.php

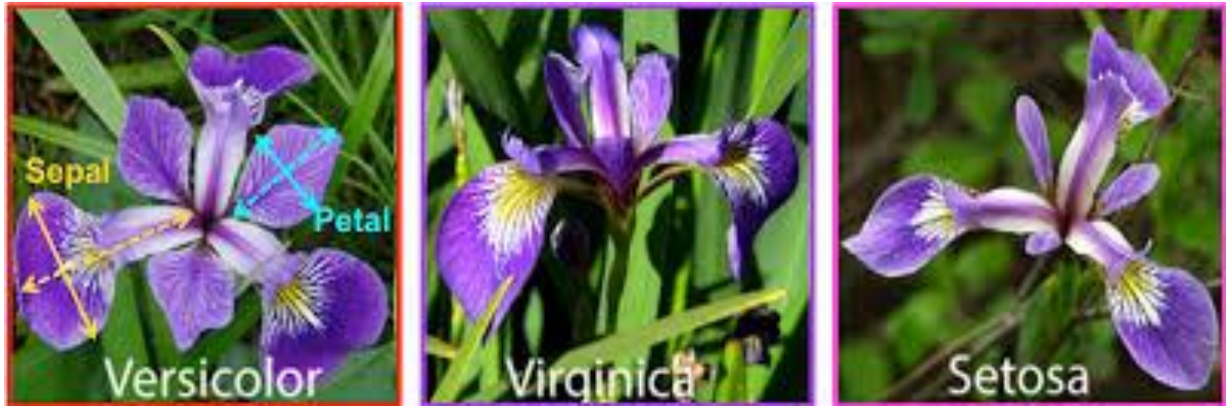


PRACTICALLY LIMITLESS APPLICATIONS

- screening and diagnosis
- adaptive clinical trials
- operations research
- home health and wearables
- precision medicine
- global health
- genomic analysis
- drug discovery and design
- robotics

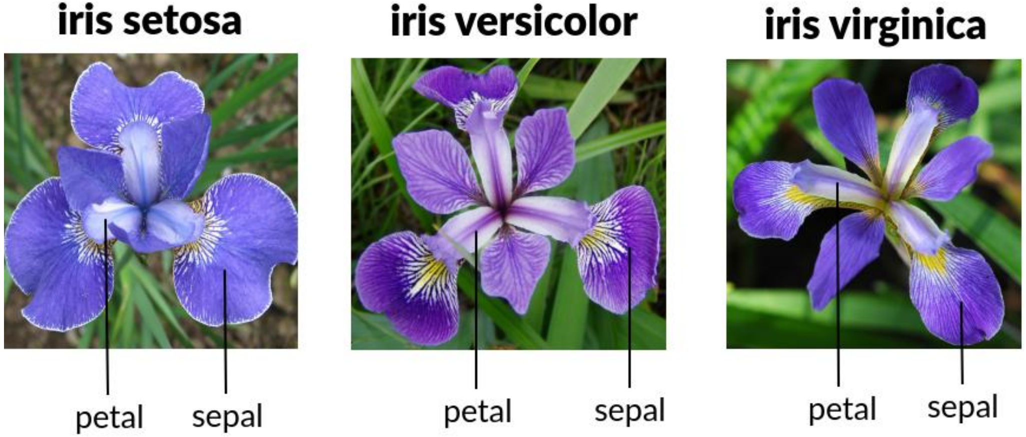
and many many more.

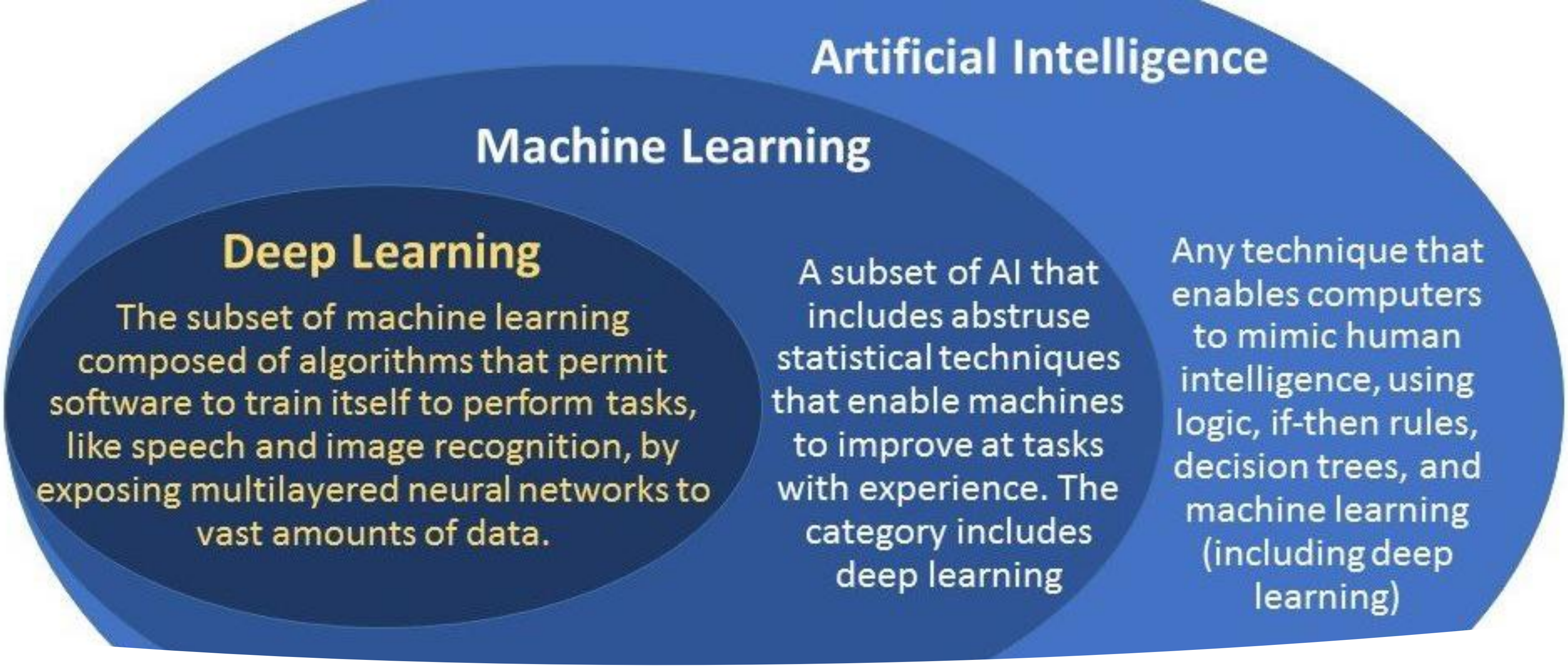
What is meant by features in classical Machine Learning?



- **Iris dataset Attribute (Features) Information:**

- 1. sepal length in cm
2. sepal width in cm
3. petal length in cm
4. petal width in cm
5. class:
-- Iris Setosa
-- Iris Versicolour
-- Iris Virginica





Artificial Intelligence

Machine Learning

Deep Learning

The subset of machine learning composed of algorithms that permit software to train itself to perform tasks, like speech and image recognition, by exposing multilayered neural networks to vast amounts of data.

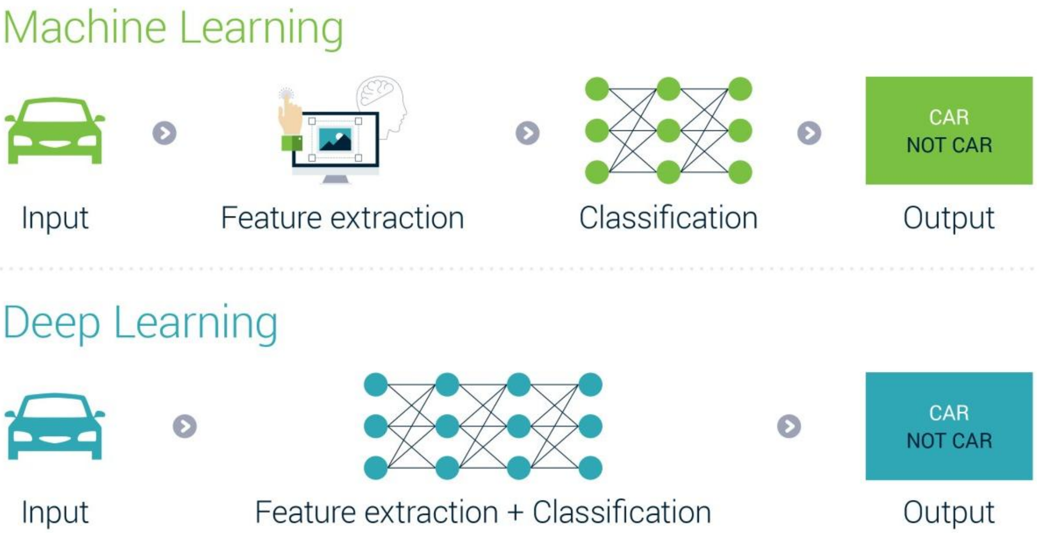
A subset of AI that includes abstruse statistical techniques that enable machines to improve at tasks with experience. The category includes deep learning

Any technique that enables computers to mimic human intelligence, using logic, if-then rules, decision trees, and machine learning (including deep learning)

The Evolution from Expert Systems to Data mining and AI

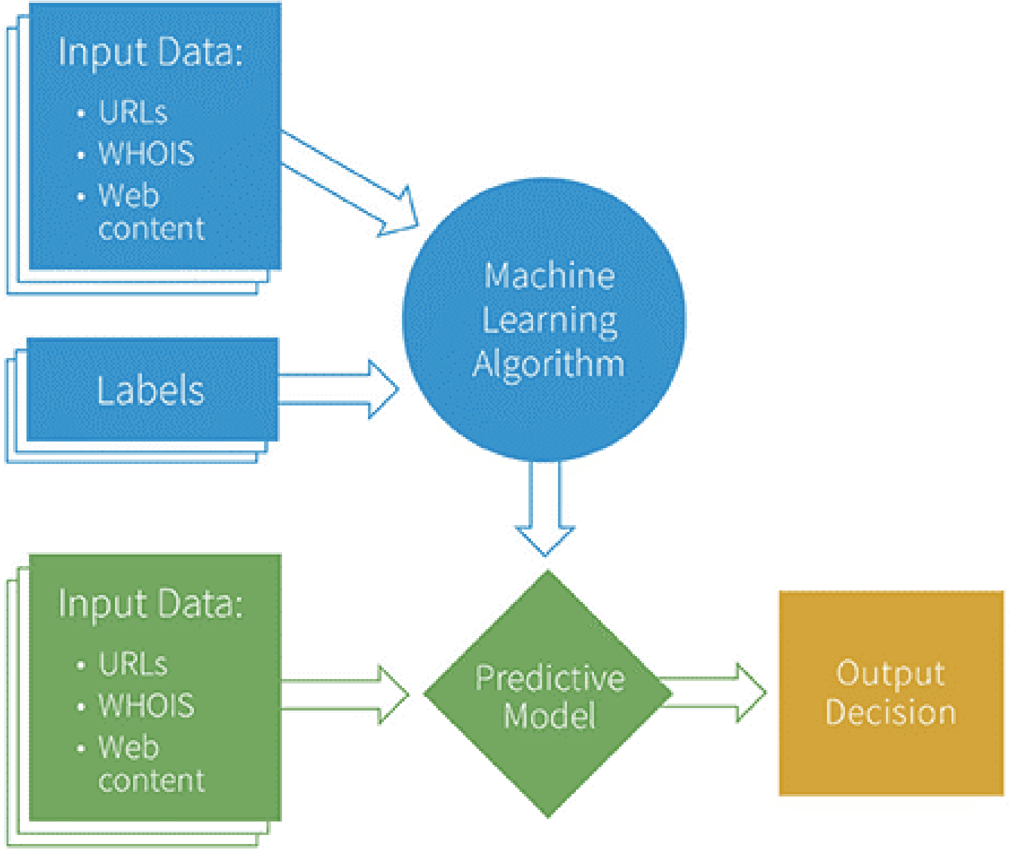
Deep Learning

- What is the difference between Machine Learning and Deep learning?
- Deep learning applications are best suited in the image processing and natural language processing fields. **In cybersecurity, it has found a home in packet stream and malware binary analysis.** These benefit most from supervised learning, when labeled (i.e., legitimate vs. malicious) data is available.

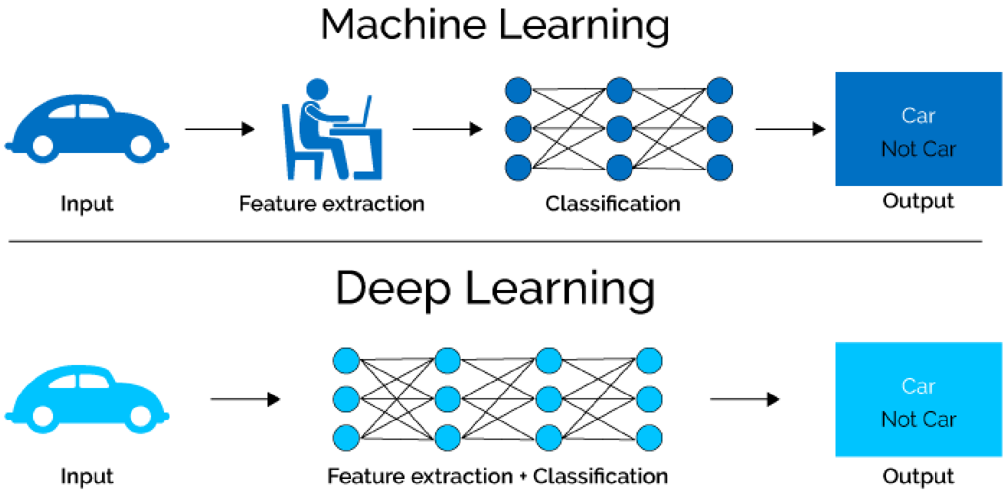


Example of Using AI in other Fiels (Cybersecurity)

- On the other hand, many cybersecurity problems cannot be solved without machine learning. Consider the phishing scam domain detection shown in Figure . Here, the URLs, WHOIS data, other properties, as wells as the known (legitimate or malicious) labels of URLs are examined in a supervised learning setting to predict whether a domain is malicious. It does so without resorting to conventional, but less effective, blacklist-based matching.



What is Deep Learning?



- A machine learning subfield of learning **representations** of data. Exceptional effective at **learning patterns**.
- Deep learning algorithms attempt to learn (multiple levels of) representation by using a **hierarchy of multiple layers**
- If you provide the system **tons of information**, it begins to understand it and respond in useful ways.

AI applications in medicine; why now?

Why did it take so long to see meaningful progress until much later around these ideas?

The concepts and ideas have always worked well together.

Historically progress plateaued because even as the ideas were developing, we didn't have all the ingredients needed for AI algorithms to perform at high levels for most of the 20th century.

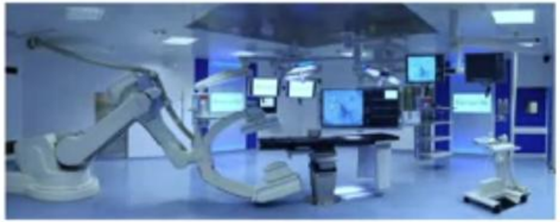
So these great ideas really started to get moving once AI algorithms capable of representing highly complex models were available.

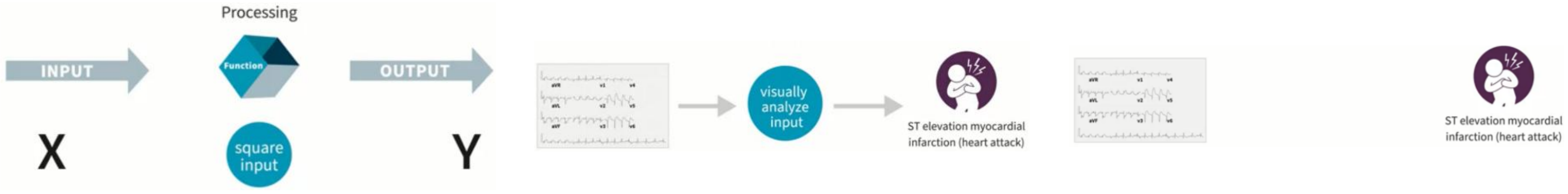
But it took two more important components for things to really come together:

access to large volumes of digital healthcare data for training

powerful graphical

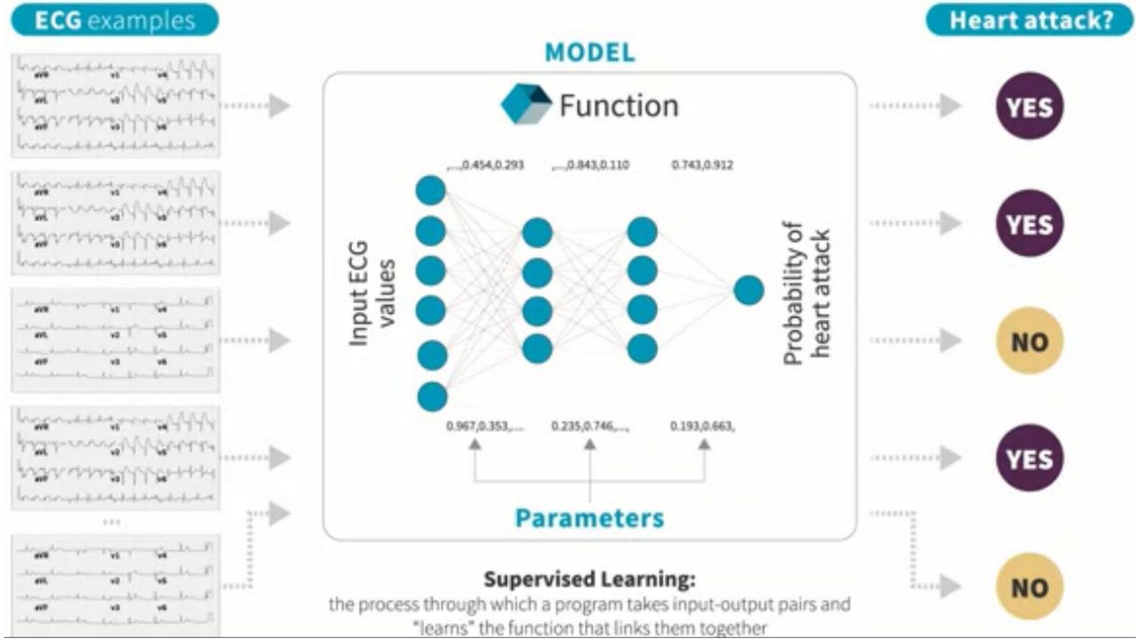
AI applications in medicine ; why now?




Processing
INPUT
Function
OUTPUT
X
square input
Y
visually analyze input
ST elevation myocardial infarction (heart attack)
ST elevation myocardial infarction (heart attack)

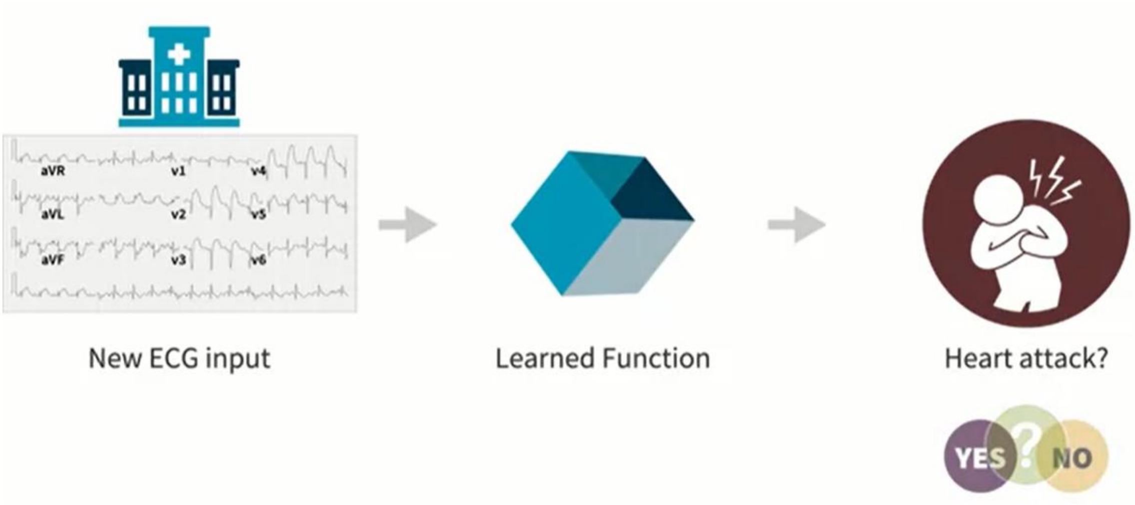
Machine learning vs traditional techniques

Example

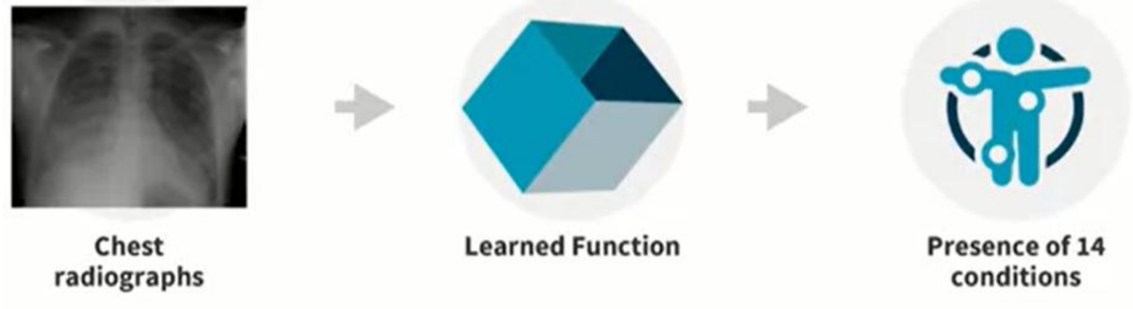


Supervised Learning:
the process through which a program takes input-output pairs and "learns" the function that links them together

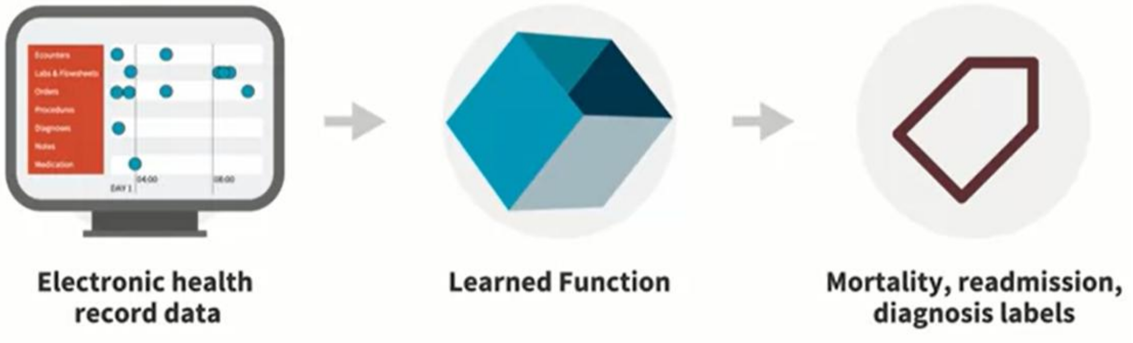
Example


aVR
v1
v4
aVL
v2
v5
aVF
v3
v6
New ECG input
Learned Function
Heart attack?
YES
?
NO

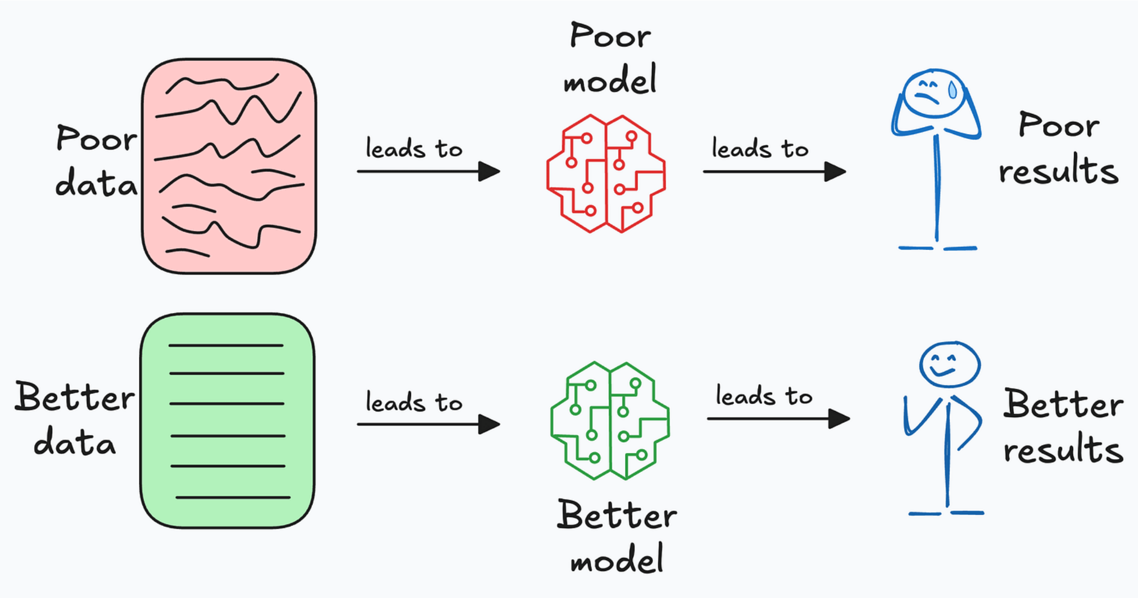
Example


Chest radiographs
Learned Function
Presence of 14 conditions

Example


Encounters
Labs & Flowsheets
Orders
Procedures
Diagnoses
Notes
Medication
DAY 1
04:00
08:00
Electronic health record data
Learned Function
Mortality, readmission, diagnosis labels

More on Data


Poor data
leads to
Poor model
leads to
Poor results
Better data
leads to
Better model
leads to
Better results

How to get Data?

Step 1: Define Your Problem Clearly

- Before searching, be specific about what you need:
- Is it a **classification**, **regression**, or **clustering** problem?
- What's your **domain** (healthcare, finance, IoT, images, text, etc.)?
- What kind of **features (inputs)** and **labels (outputs)** do you need?
- *Example:*
- "I want a dataset of images of fruits labeled by type to train a classification model."

Step 2: Choose Reliable Sources

- Here are **trusted repositories** for datasets — depending on your domain:
- 🧠 **General Machine Learning**
- Kaggle Datasets — Best overall, community-curated, ready-to-use.
- UCI Machine Learning Repository — Classic datasets used in research and education.
- Google Dataset Search — Search engine for datasets.
- OpenML — Share and discover datasets with metadata and ML tasks.

Resources Examples:

- **Image & Computer Vision**
  - ImageNet — Large-scale labeled image dataset.
  - COCO Dataset — Common Objects in Context (for object detection).
  - Kaggle's "Dogs vs Cats" or "MNIST" — Great for beginners.
- **Text & NLP**
  - IMDb Reviews — Sentiment analysis.
  - 20 Newsgroups — Text classification.
  - Hugging Face Datasets — Massive NLP dataset hub.
- **Tabular / Structured Data**
  - UCI Repository — Many small structured datasets.
  - Kaggle — Finance, healthcare, marketing, IoT data.
  - Data.gov — Public US government datasets.
  - World Bank Data — Economic and social indicators.
- **IoT / Sensor / Time-Series**
  - UCI Energy or Gas Sensor Datasets
  - Awesome Time Series Datasets (GitHub)
  - PhysioNet — Biomedical sensor signals.

Before using any dataset, check

Step 3: Evaluate Dataset Quality

Criterion	Why It Matters
Relevance	Fits your ML goal.
Size	Enough samples for training/testing.
Balance	Classes not too skewed.
Completeness	Few missing or corrupted values.
Label Accuracy	Correct annotations.
Licensing	Free and legal to use.

Step 4: Clean and Prepare Data

Even the best dataset needs:

Handling missing values.

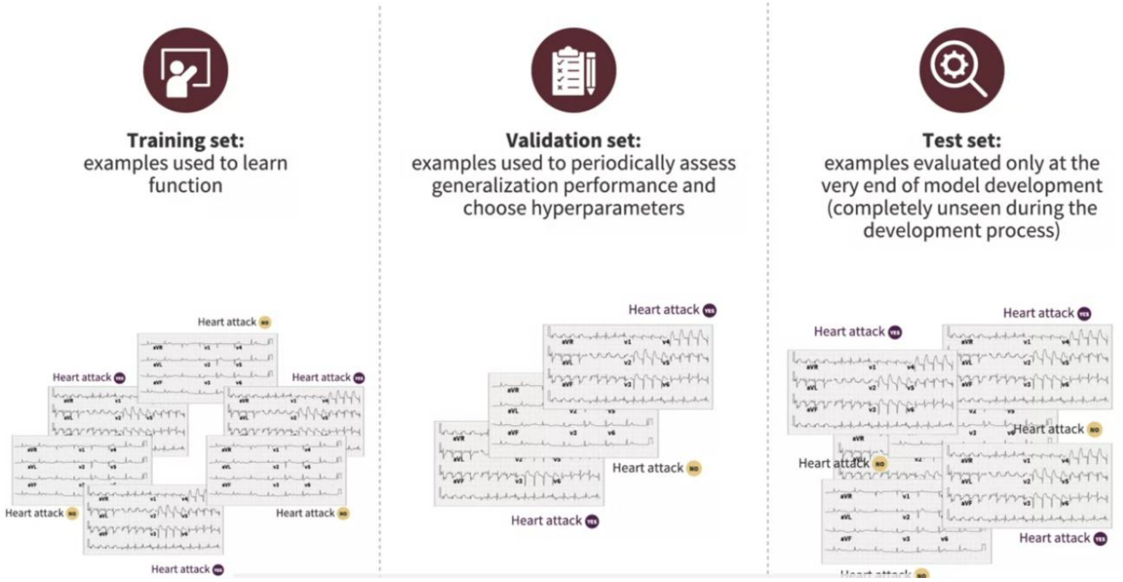
Normalization or standardization.

Removing duplicates and outliers.

Splitting into **train/test/validation sets.**

- **Create your own** (collect data manually or via APIs).
- Use **data augmentation** to expand small datasets.
- **Combine** multiple open datasets.

Dataset categories



Training set:
examples used to learn function

Validation set:
examples used to periodically assess generalization performance and choose hyperparameters

Test set:
examples evaluated only at the very end of model development (completely unseen during the development process)

Training, Testing and Validation Sets

- **Training Set**:
  - this data set is used to train the algorithm and adjust parameters.
- **Validation Set**:
  - this data set is used to minimize overfitting. You're not adjusting the parameters with this data set, you're just verifying that the network is training well.
  - If the accuracy over the training data set increases, but the accuracy over the validation data set stays the same or decreases, then you're overfitting and you should stop training or change the algorithm.
- **Testing Set**:
  - this data set is used only for testing the final solution.
- Typically; Training:testing:validation is 50:25:25 if you have plenty of data or 60:20:20, 70,20,10, 80, 10,10 if you do not.

Generalization

- We want to find a model that can process new inputs and produce new and accurate outputs.
- This means that we can't evaluate the model on inputs it's already seen. We want to ensure that our model can generalize what it's learned, to new inputs from real world applications, which means evaluating it on examples that it was not already exposed to, during function learning, which would essentially be cheating.
- We want to evaluate our model on unseen examples. If a model can produce accurate outputs for these unseen examples, then we can say that the model generalizes well to new inputs.
- The validation set is a set of examples, or input-output pairs, that we hold out and do not expose the model to during training. And instead we use it to periodically assess, or validate the generalization performance of our model, as we develop the model

Training, validate and test Cycle

START TRAINING

APPLY TRAINING DATA

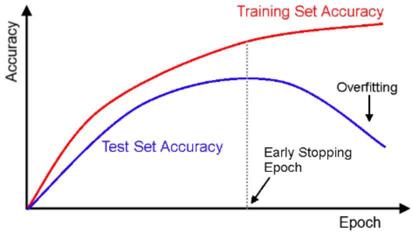
EVALUATE THE SYSTEM PERFORMANCE THROUGH VALIDATION DATA

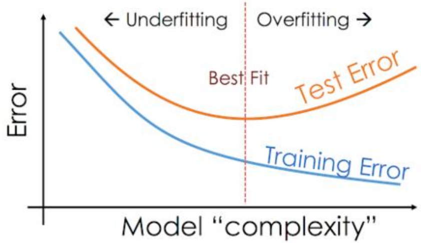
REPEAT STEPS 2 AND 3 UNTIL THE PERFORMANCE REACHES THE TARGET OR STOP IMPROVING OR EXCEED MAX. TRAINING TIMES.

STOP TRAINING AND APPLY THE TEST DATA TO EVALUATE THE SYSTEM

Overfitting

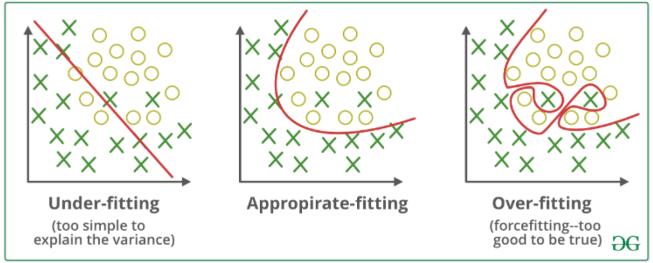
- An overfit model is one that is too complicated for your data set.
- Overfitting happens when a model learns too much from the training data, including details that don’t matter (like noise or outliers).
- For example, imagine fitting a very complicated curve to a set of points. The curve will go through every point, but it won’t represent the actual pattern.
- As a result, the model works great on training data but fails when tested on new data.
- Overfitting models are like students who memorize answers instead of understanding the topic. They do well in practice tests (training) but struggle in real exams (testing).
- **Reasons for Overfitting:**
- High variance and low bias.
- The model is too complex.
- The size of the training data.





Underfitting

- Underfitting is the opposite of overfitting. It happens when a model is too simple to capture what's going on in the data.
- For example, imagine drawing a straight line to fit points that actually follow a curve. The line misses most of the pattern.
- In this case, the model doesn't work well on either the training or testing data.
- Underfitting models are like students who don't study enough. They don't do well in practice tests or real exams. **Note: The underfitting model has High bias and low variance.**
- **Reasons for Underfitting:**
- The model is too simple, So it may be not capable to represent the complexities in the data.
- The input features which is used to train the model is not the adequate representations of underlying factors influencing the target variable.
- The size of the training dataset used is not enough.
- Excessive regularization are used to prevent the overfitting, which constraint the model to capture the data well.
- Features are not scaled.


Under-fitting
(too simple to explain the variance)
Appropirate-fitting
Over-fitting
(forcefitting--too good to be true)

QUESTIONS?