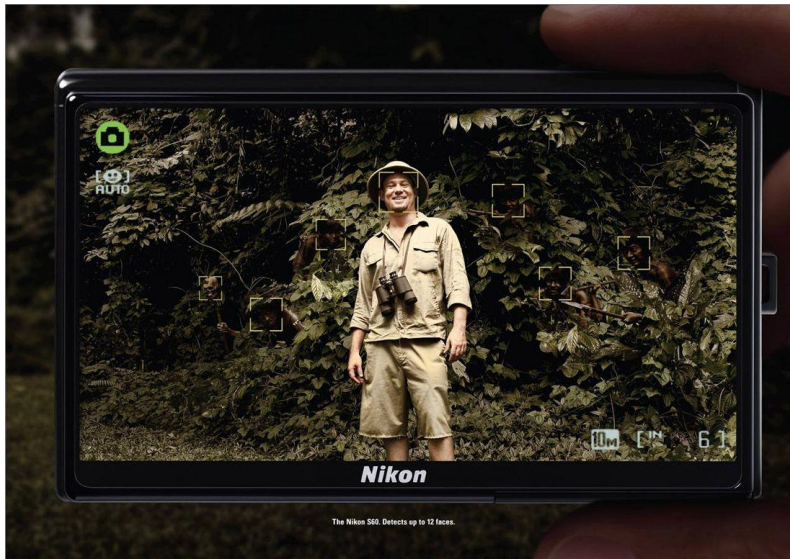


## SESSION 3: MACHINE LEARNING FUNDAMENTALS

Overview of machine learning concepts and introduce popular machine learning algorithms.

## Introduction: Machine learning is everywhere!



Face Recognition



Object recognition (e.g., for self-driving cars)

## Introduction: Machine learning is everywhere!

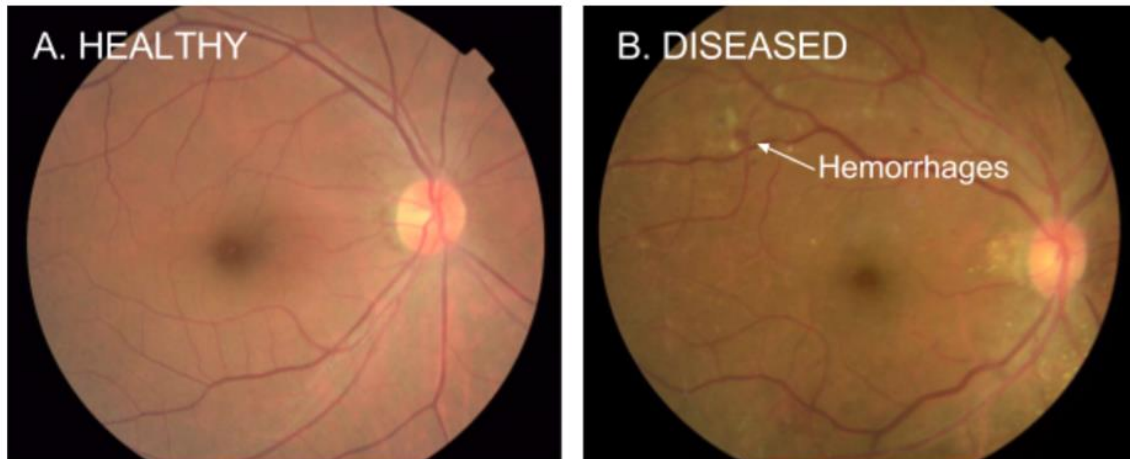


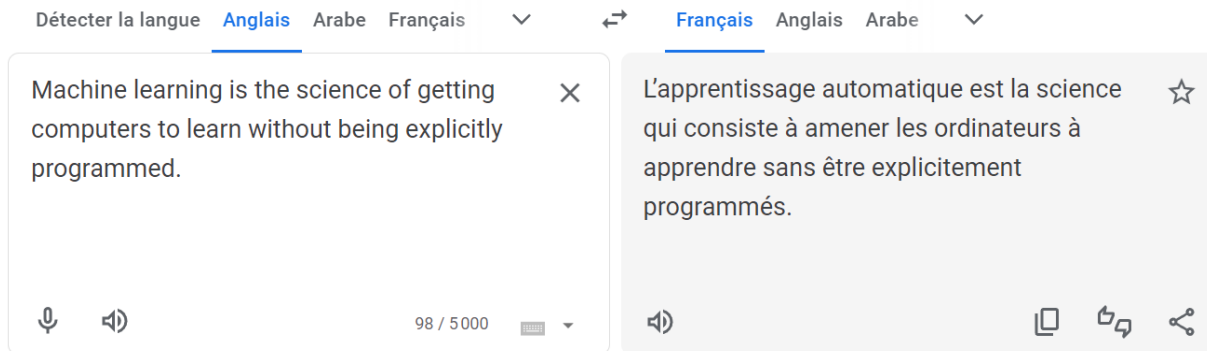
Figure 1. Examples of retinal fundus photographs that are taken to screen for DR. The image on the left is of a healthy retina (A), whereas the image on the right is a retina with referable diabetic retinopathy (B) due a number of hemorrhages (red spots) present.

**Analysis of medical images**



**Conversational agents (i.e.,  
speech recognition... )**

# Introduction: Machine learning is everywhere!



## Machine Translation



**You**

What neural network architecture are you built on?



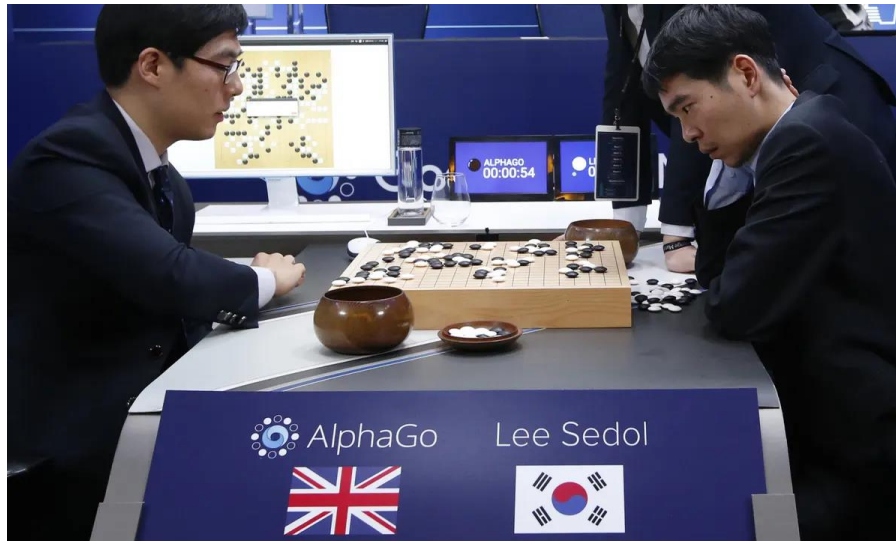
**ChatGPT**

I'm built on the GPT-3.5 architecture, which stands for "Generative Pre-trained Transformer 3.5." The underlying architecture is a transformer neural network developed by OpenAI. The transformer architecture is known for its effectiveness in natural language processing tasks and has been widely used in various language models.

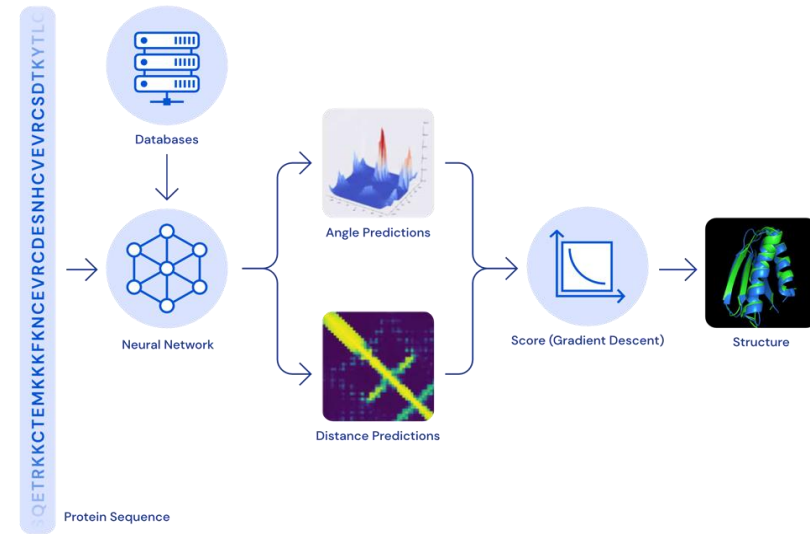
GPT-3.5 is one of the latest versions of the GPT series, featuring 175 billion parameters, making it one of the largest language models created by OpenAI as of my last knowledge update in January 2022. If there have been any updates or new models since then, I may not be aware of them.

## Advanced Chatbots

## Introduction: Machine learning is everywhere!

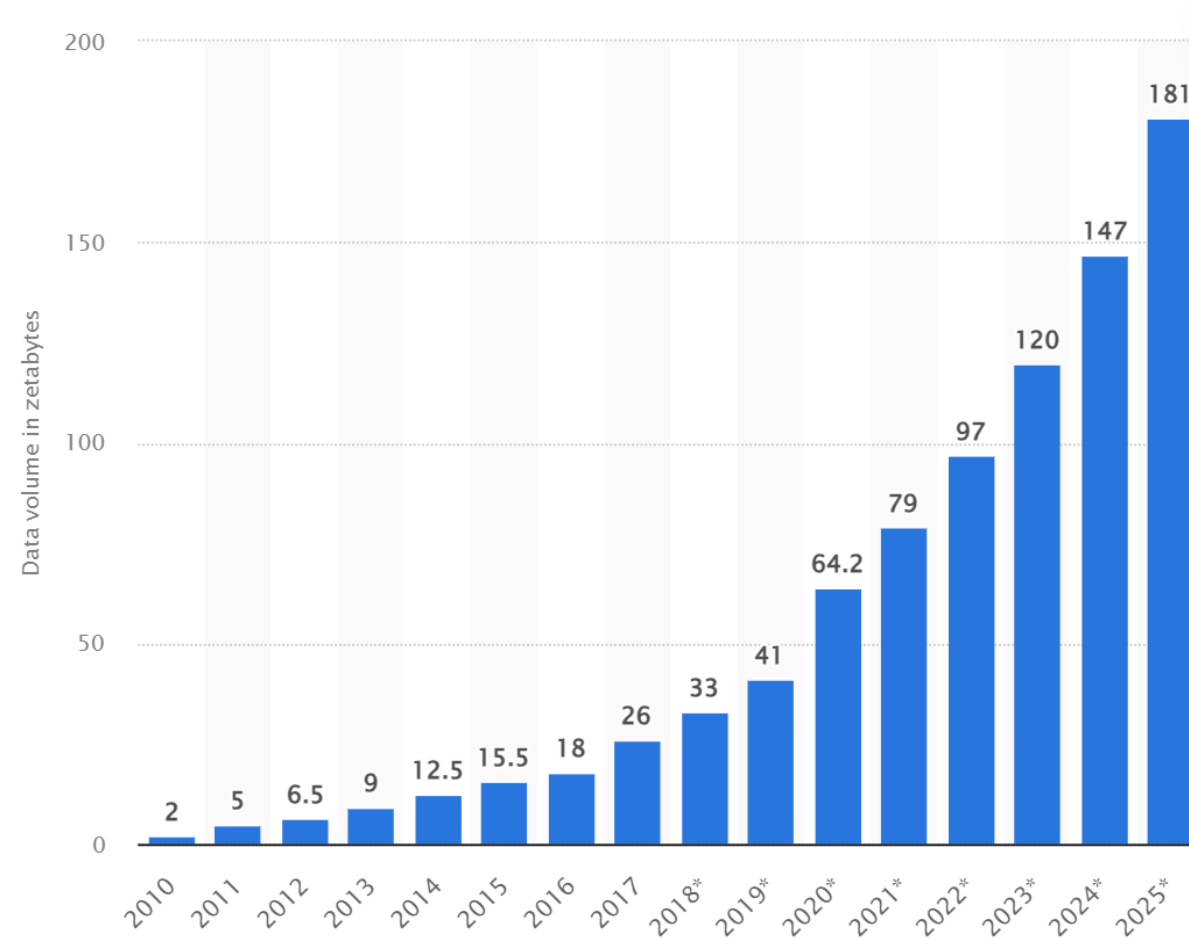


AlphaGo beats Lee Sedol



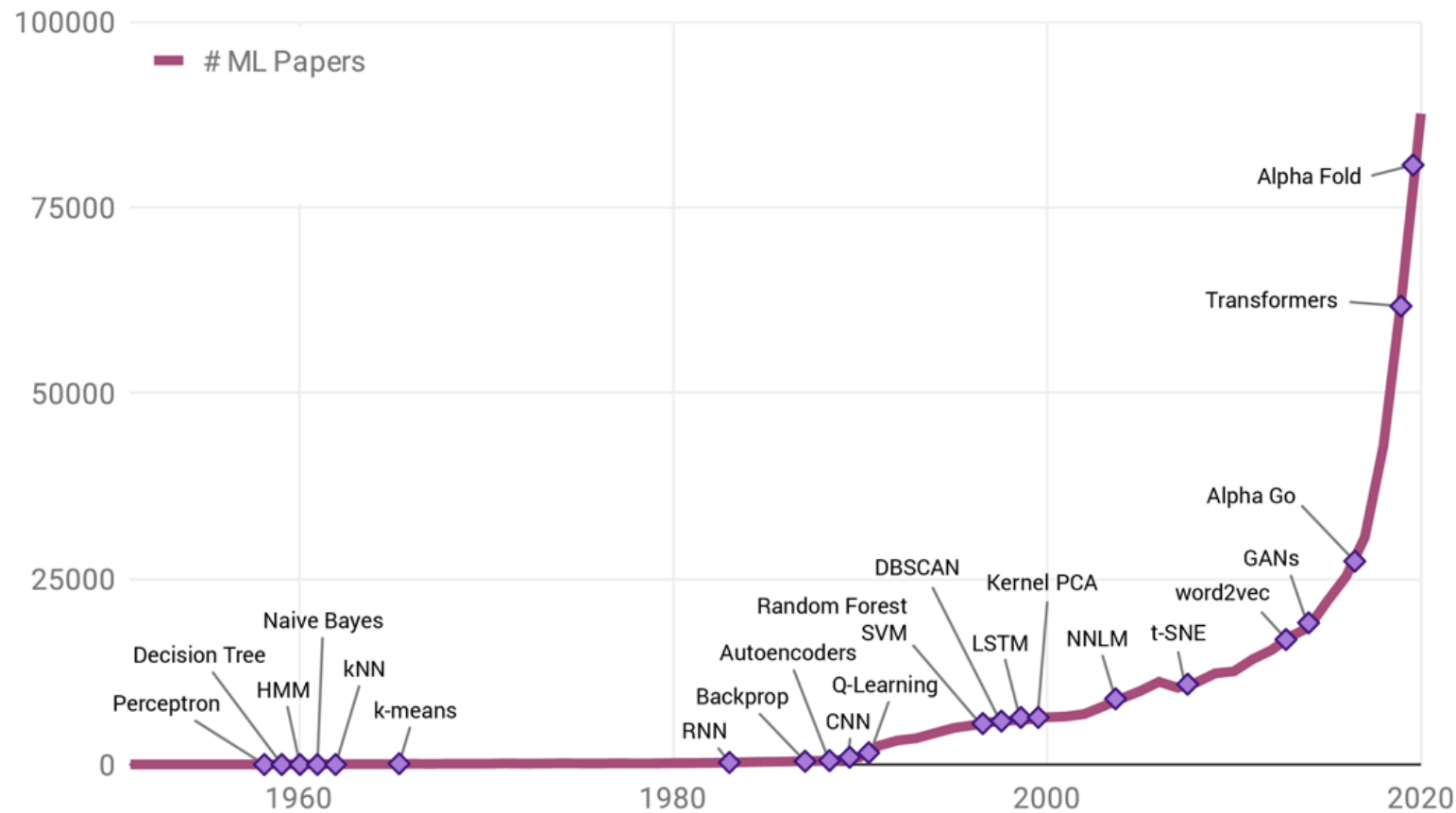
Solving protein folding challenge  
(AlphaFold)

## Introduction: Why is ML so popular? <Increased volumes of data>



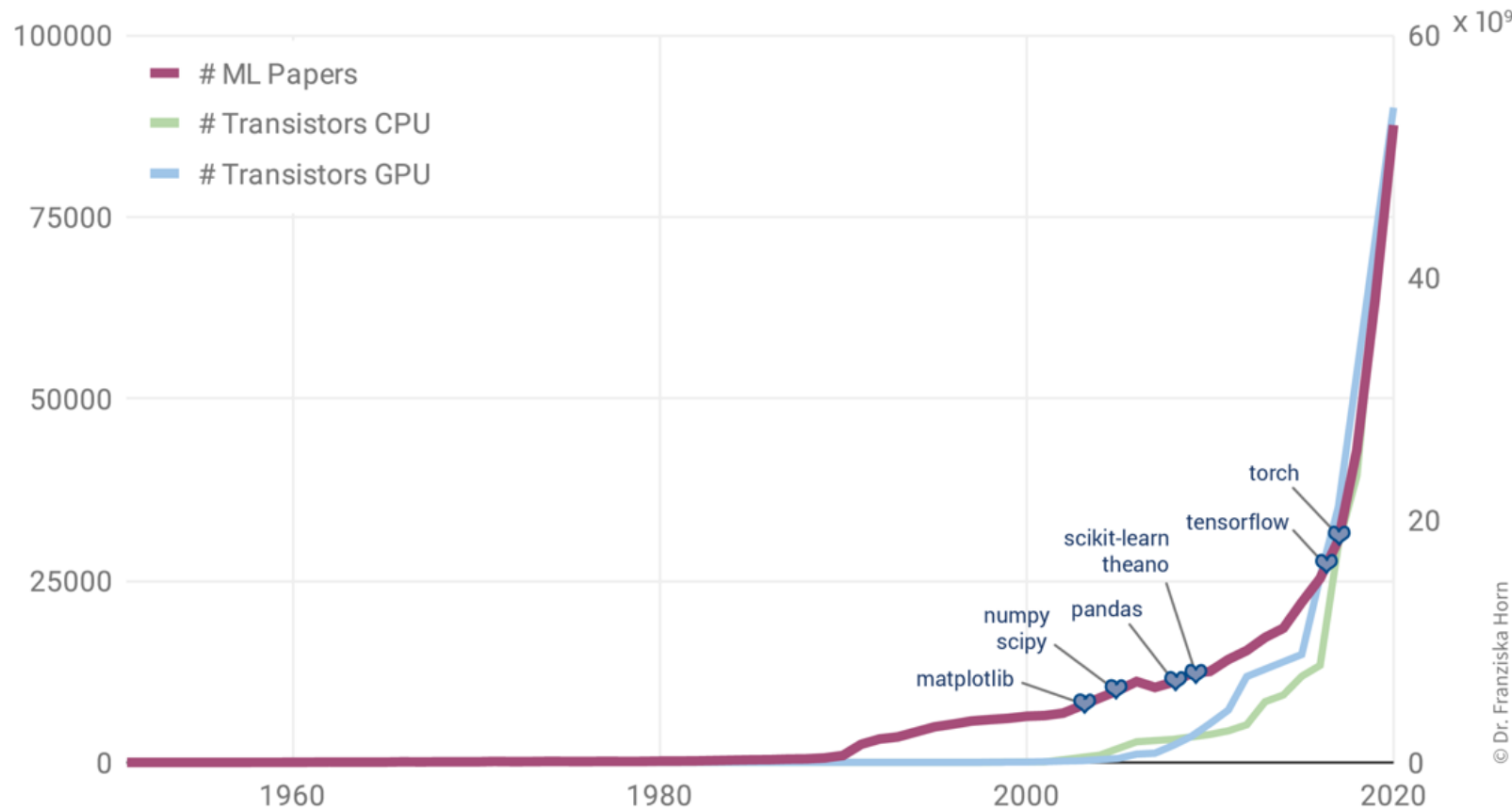
Source: [Statista.com](https://www.statista.com)

# Introduction: Why is ML so popular? <Advanced Algorithms>



© Dr. Franziska Horn

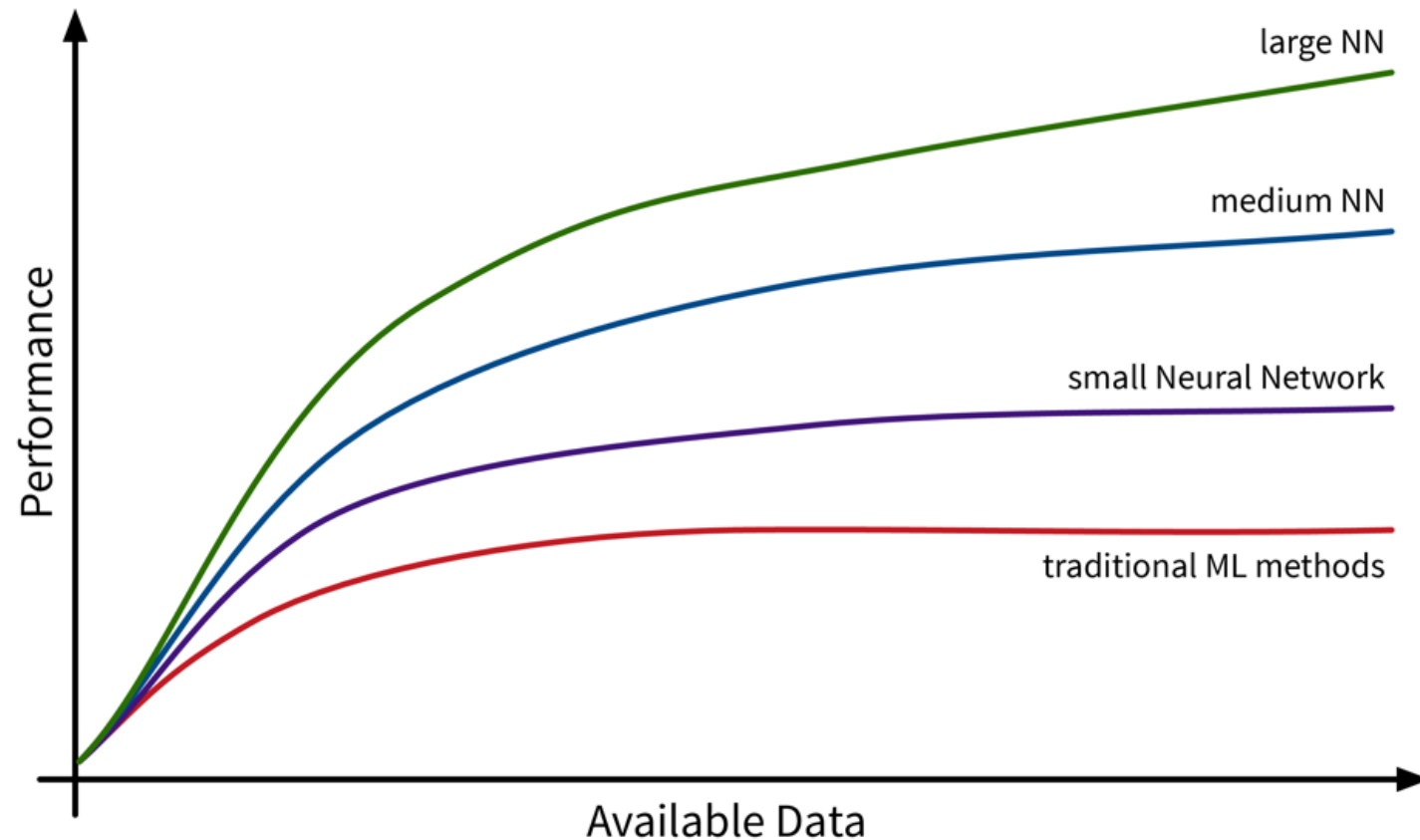
## Introduction: Why is ML so popular? <Advanced Computational Resources>



Source: [A Practitioner's Guide to Machine Learning](#)



## Introduction: Why is ML so popular? <Available Data vs. Performance>

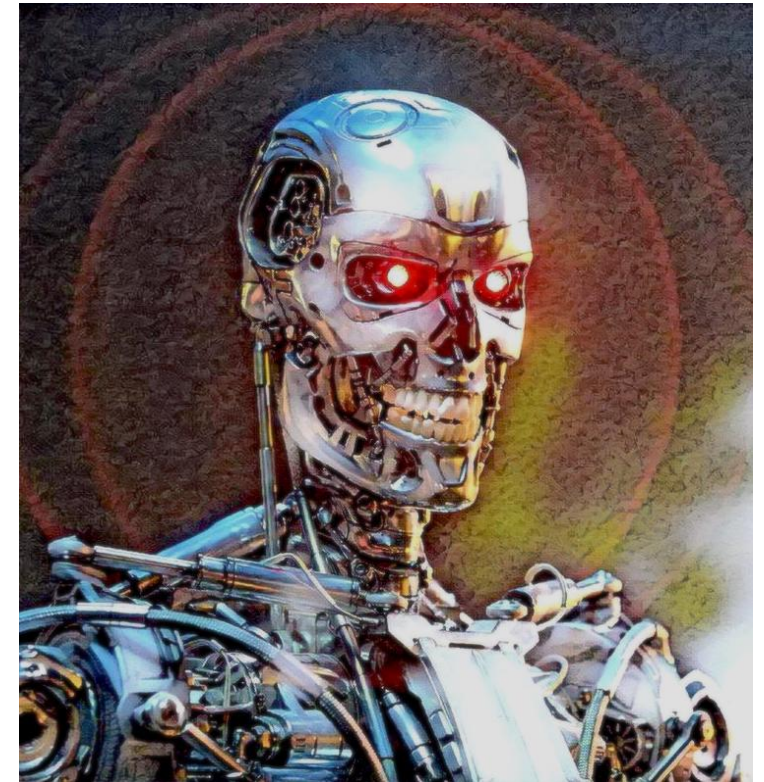


Source: [A Practitioner's Guide to Machine Learning](#)

## Introduction: Is AGI Coming?

**Artificial General Intelligence (AGI)** refers to a type of artificial intelligence that possesses the ability to understand, learn, and apply knowledge across a wide range of tasks, similar to the broad cognitive abilities of humans. AGI is often contrasted with Narrow AI, which is designed for specific tasks and lacks the versatility of human-like intelligence.

**Wikipedia says:** The timeline for AGI development remains a subject of ongoing debate among researchers and experts. Some argue that it may be possible in years or decades; others maintain it might take a century or longer; and a minority believe it may never be achieved



Source: [A Practitioner's Guide to Machine Learning](#)

## Basics: Data is the New Oil!

It all started with the data, "**data is the new oil**" is a metaphor that has been used to highlight the increasing importance of data in the modern economy.

The reason why oil is considered valuable is because we have important use cases for it: powering our cars, heating our homes, and producing plastics or fertilizers. Similarly, our data is only as valuable as what we make of it. **So, what can we use data for?**

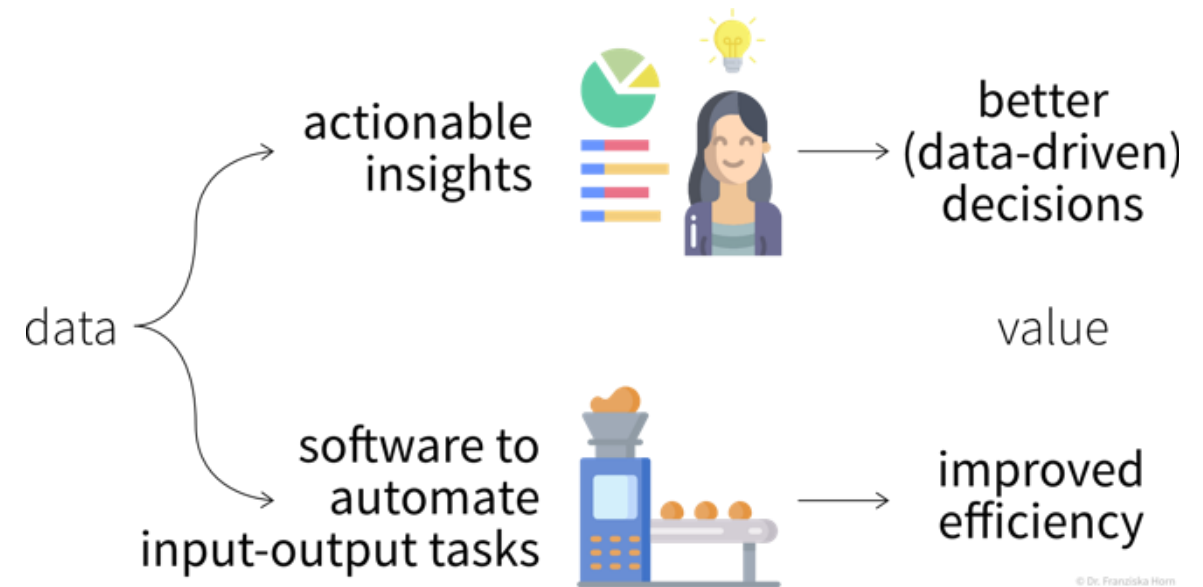


Source: [economist.com](https://www.economist.com)

## Basics: Data is the New Oil!

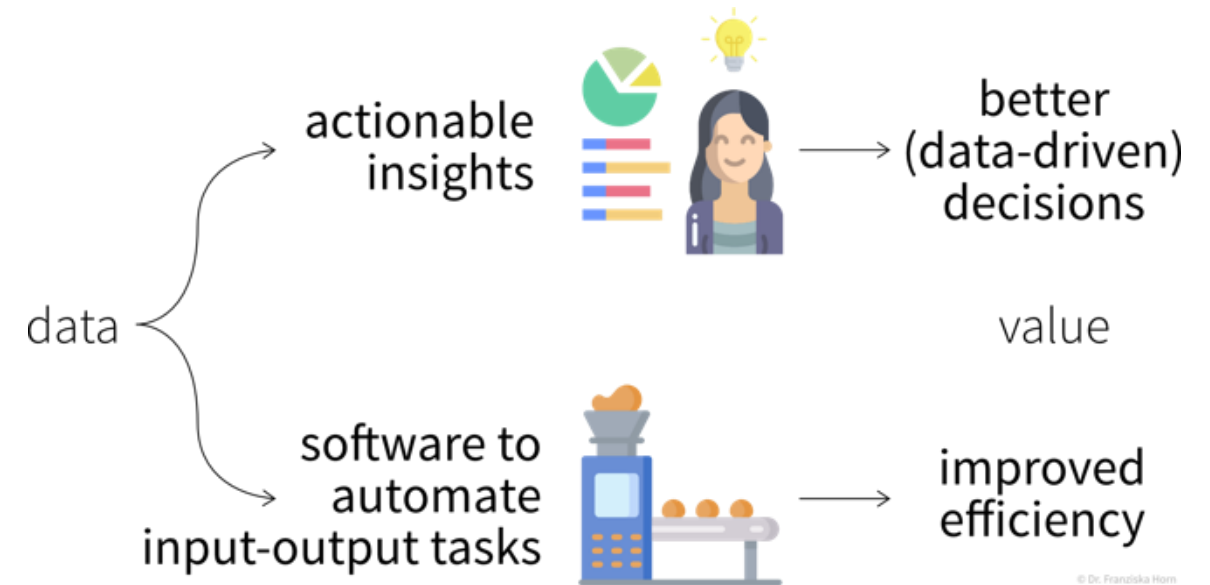
The main use cases belong to one of two categories:

- **Insights:** We can generate insights either through continuous monitoring (“Are we on track?”) or a deeper analysis (“What’s wrong?”). By visualizing important variables or Key Performance Indicators (KPIs) in reports or dashboards, we increase the transparency of the current status and quantify our progress toward some goal.
- **Automation:** machine learning models can be used to automate ‘input → output’ tasks otherwise requiring a human (expert). These tasks are usually easy for an (appropriately trained) human, for example: Translating texts from one language into another, Sorting out products with scratches when they pass a checkpoint on the assembly line, Recommending movies to a friend. For this to work, the ML models need to be trained on a lot of historical data (e.g., texts in both languages, images of products with and without scratches, information about different users and which movies they watched).



## Basics: Data is the New Oil!

**Note:** These categories are not mutually exclusive, and many applications involve a combination of insights and automation. For example, data insights derived from analytics might inform the development of automation systems, and automated processes generate more data that can be analyzed for continuous improvement.



**Source:** [A Practitioner's Guide to Machine Learning](#)

## Basics: What is Machine Learning?

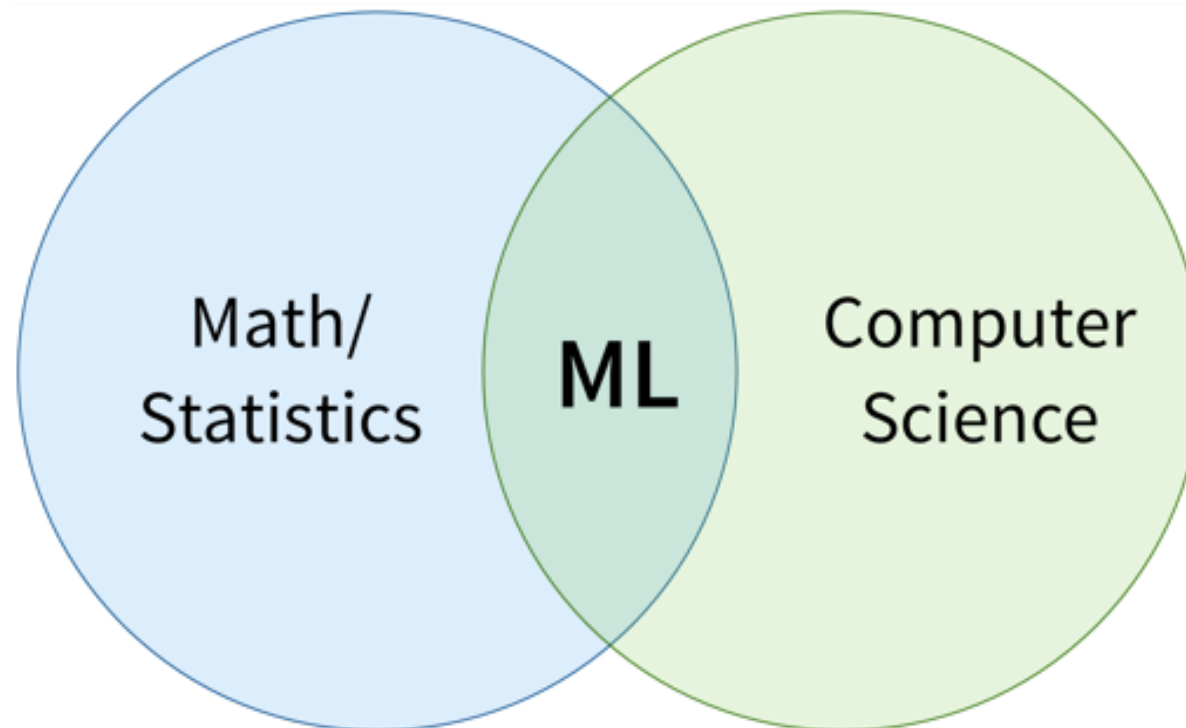
- **Definition 1:** Machine learning is the science of getting computers to act without being explicitly programmed
- **Definition 2:** A subset of artificial intelligence (AI) and computer science, machine learning (ML) deals with the study and use of data and algorithms that mimic how humans learn. This helps machines gradually improve their accuracy. ML allows software applications to improve their prediction accuracy without being specifically programmed to do so. It estimates new output values by using historical data as input.





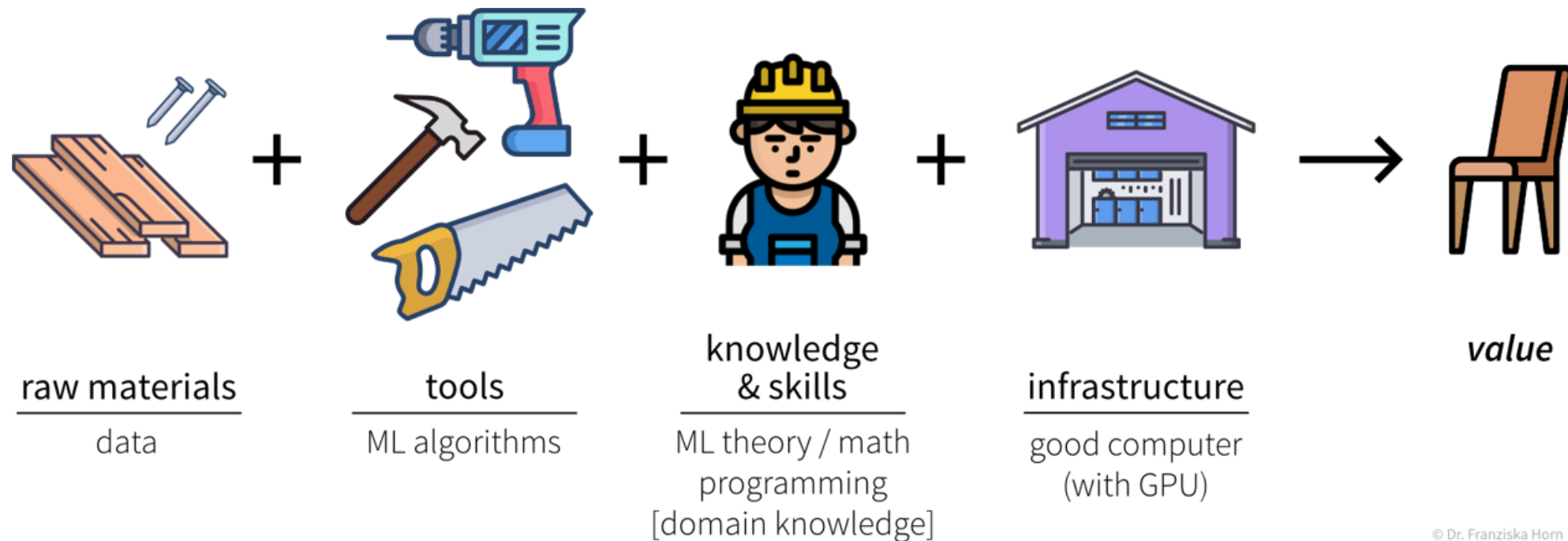
## Basics: What is Machine Learning?

ML is an area of research in the field of theoretical computer science, i.e., at the intersection of mathematics and computer science



## Basics: What is Machine Learning?

Machine learning algorithms are considered as tools to generate the value from the data



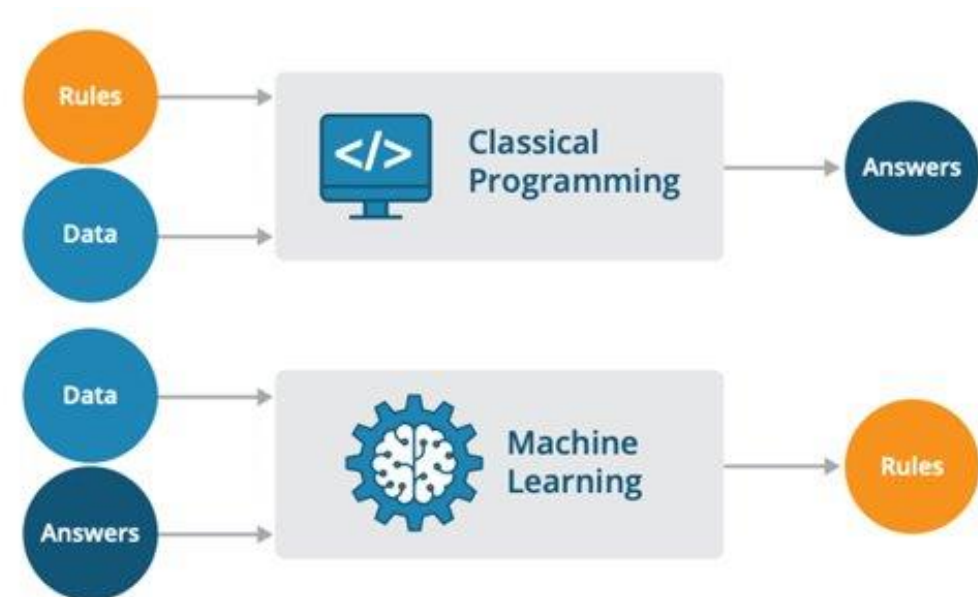
© Dr. Franziska Horn

Source: [A Practitioner's Guide to Machine Learning](#)



## Basics: Machine Learning vs. Traditional Software

While traditional software solutions are used to automate tasks that can be formulated as a fixed, predefined sequence of actions, executed according to some hard-coded rules (e.g., “a gate should open if an object passes through a photoelectric barrier and 20 seconds later the gate should close again”), machine learning can be used to automate “input → output” tasks for which it would otherwise be difficult to come up with such rules.



## Basics: Avoid "Solution In Search of a Problem" Syndrome

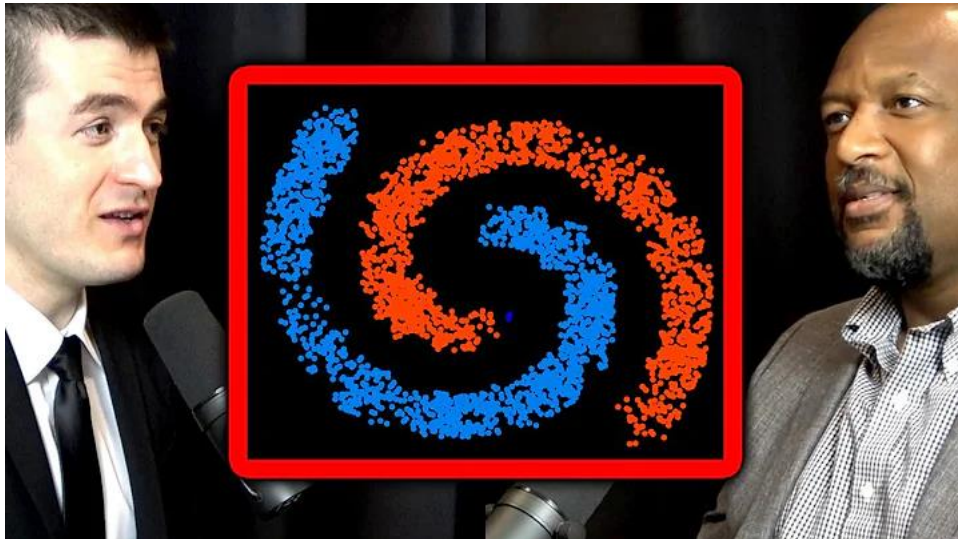
"IT WOULD BE COOL TO USE <PLACE LATEST ML TECHNOLOGY/ALGORITHM HERE"

- **Answer the question:** What problem are you solving?
  - If you can't answer this question, it is probably not worth building an ML
- **ML is overkill if:**
  - a manually defined set of rules or mechanistic (white box) model can solve the problem



Source: [reddit.com](https://www.reddit.com)

## Basics: Is Machine Learning Just Statistics?

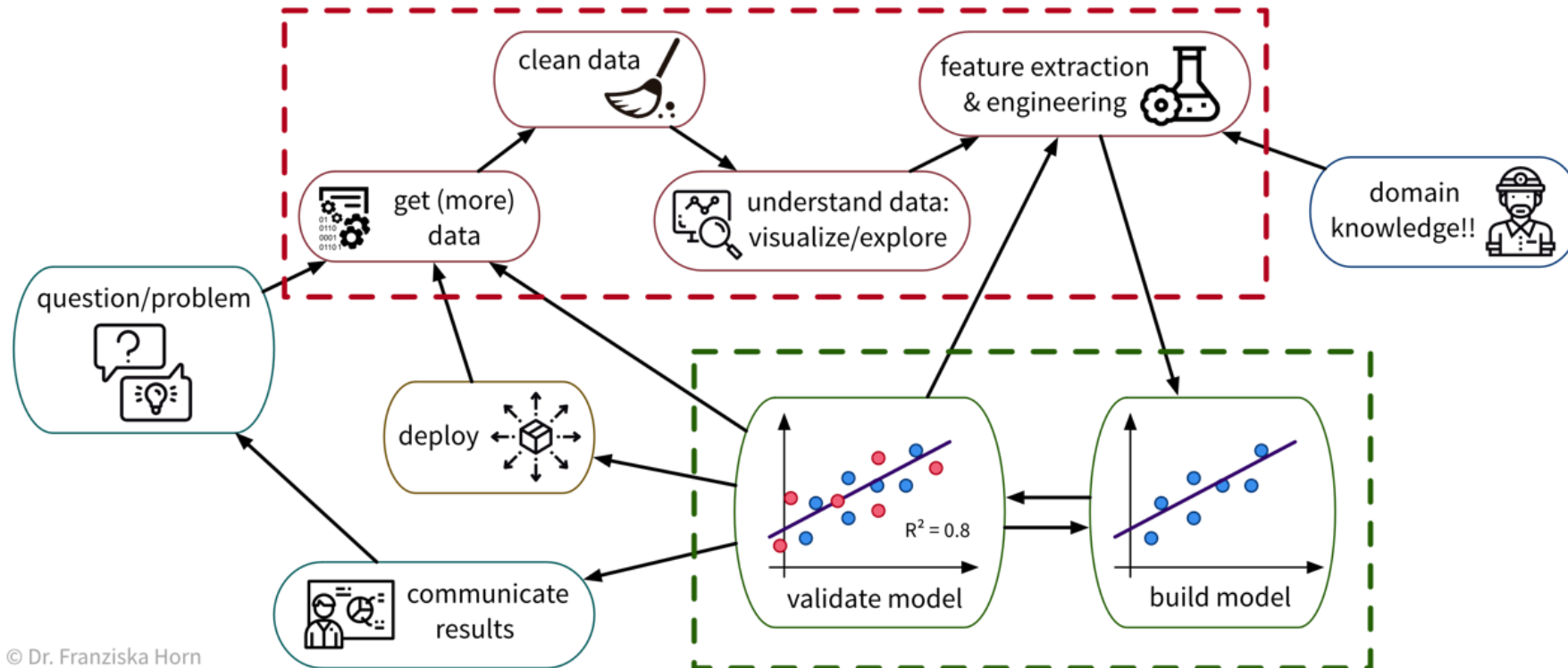


[Is machine learning just statistics? | Charles Isbell and Michael Littman and Lex Fridman](#)



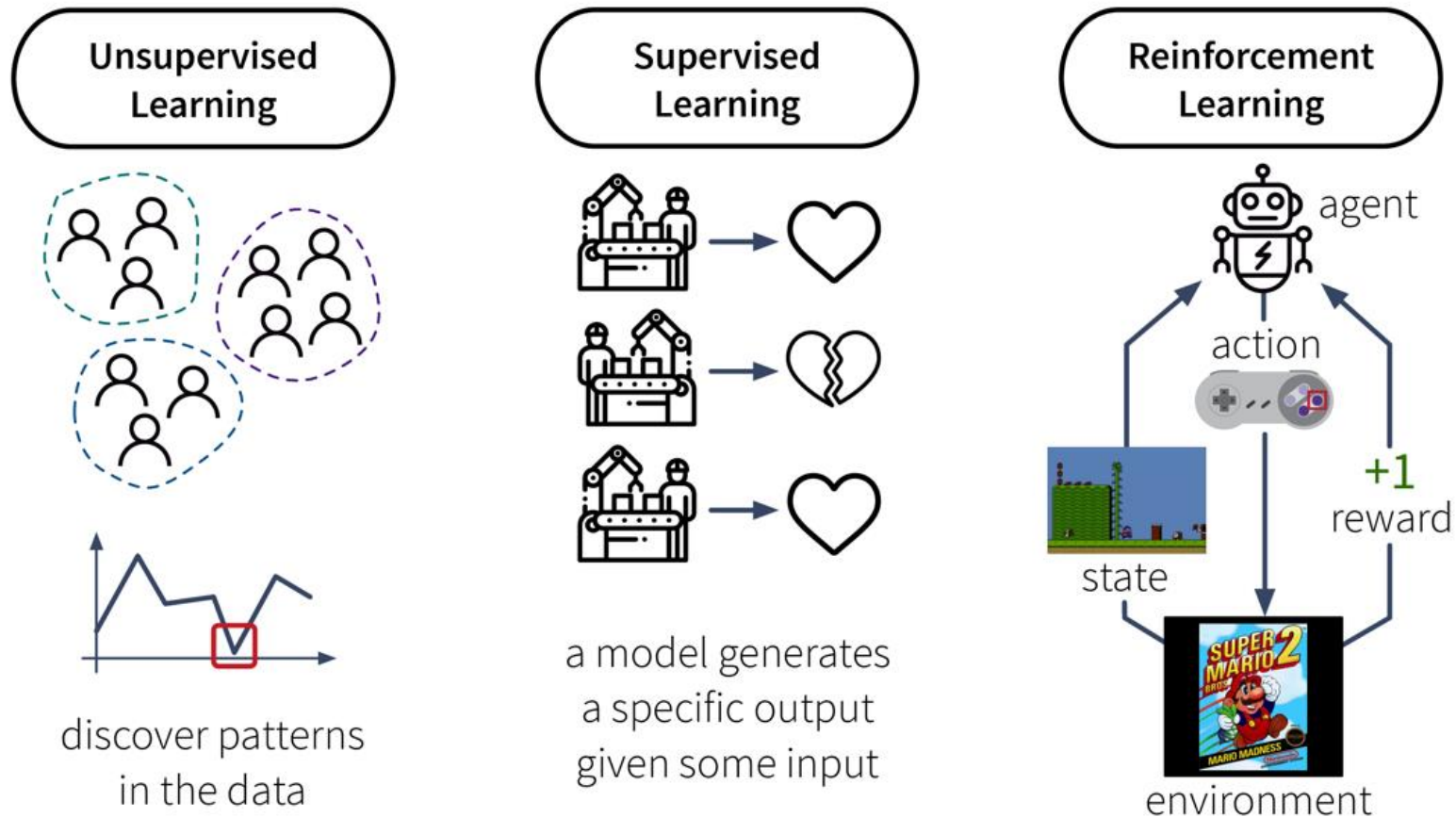
[Is AI just statistics? | Yann LeCun and Lex Fridman](#)

## Basics: Solving Problems with ML <Workflow>



© Dr. Franziska Horn

# Types of Machine Learning



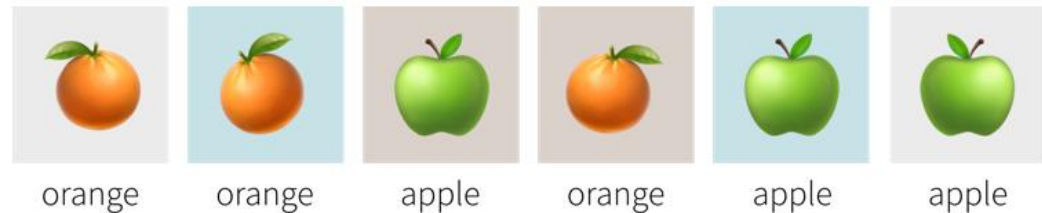


## Types of Machine Learning

**Unsupervised Learning:** a dataset with examples



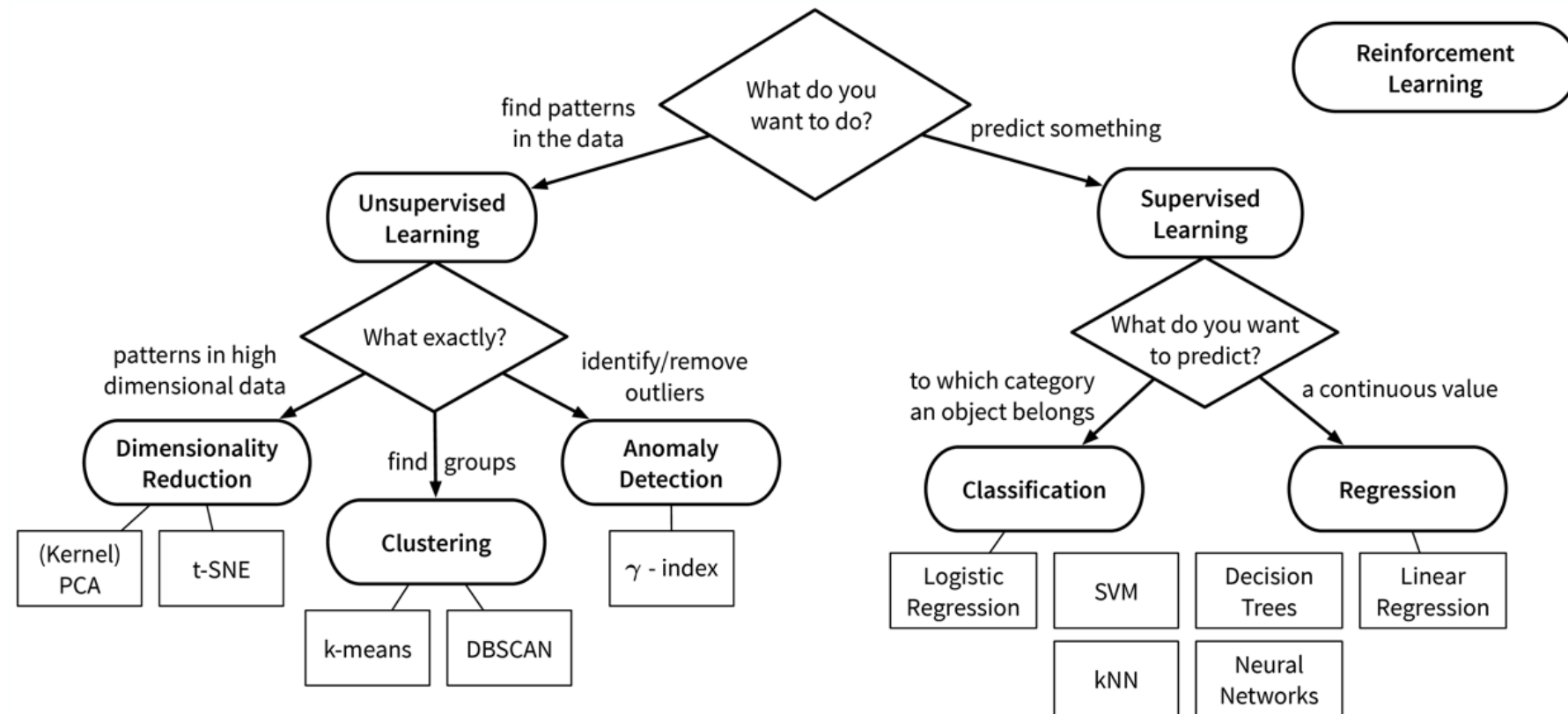
**Supervised Learning:** a dataset with labeled examples



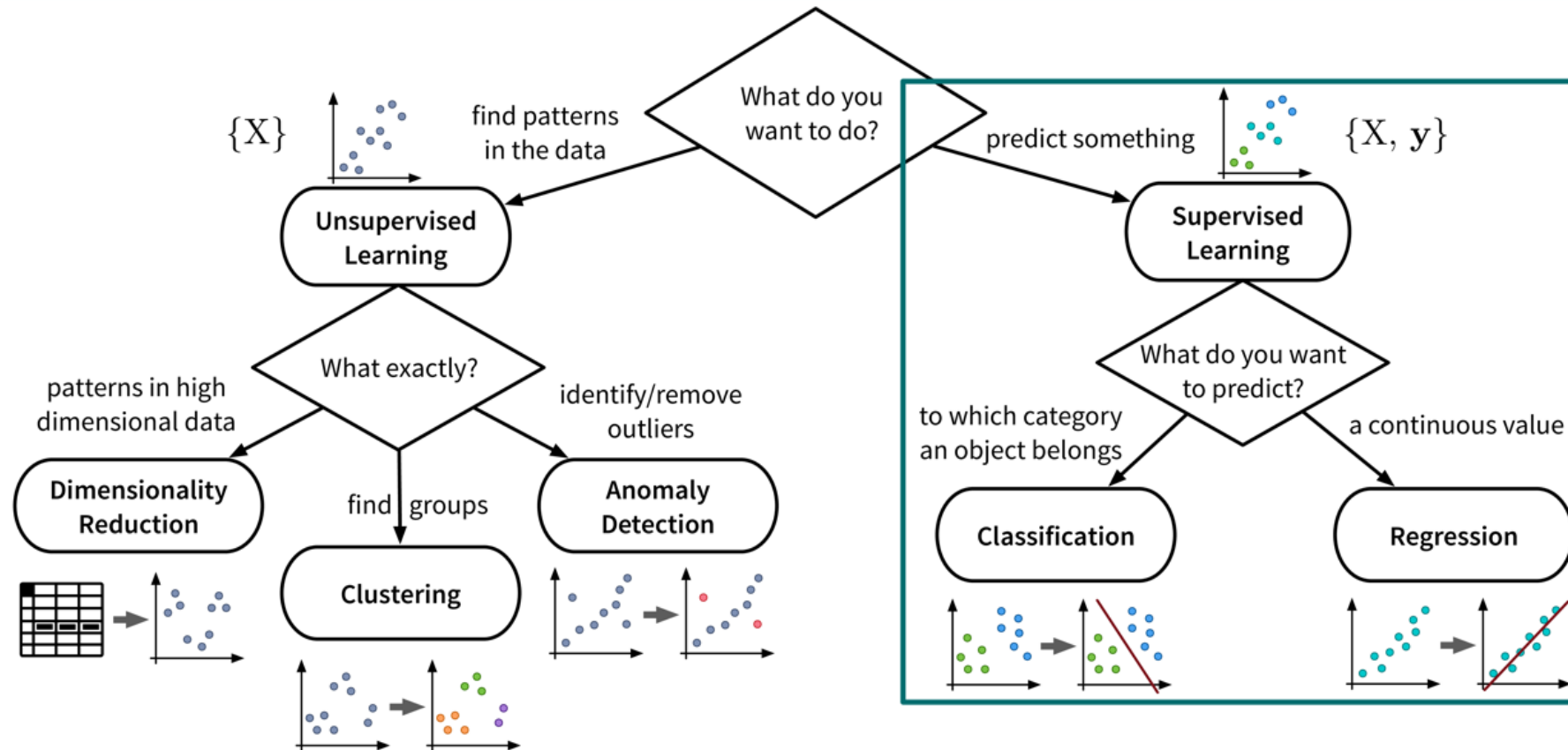
**Reinforcement Learning:** a (simulation) environment that generates data (i.e., reward + new state) in response to the agent's actions



# Types of Machine Learning



# Supervised Learning Basics



© Dr. Franziska Horn



# Supervised Learning Basics



## Regression



What will be the temperature tomorrow?

84°



Fahrenheit

## Classification



Will it be hot or cold tomorrow?

COLD

HOT



Fahrenheit

Source: [enjoyalgorithms.com](http://enjoyalgorithms.com)

# Supervised Learning Basics: Different Types of Models

The most important task of a data scientist is to select an appropriate model for solving a problem. Three considerations when choosing a supervised learning model:

## 1. Problem type

What kind of problem are you trying to solve: regression or classification?

→ Depends on the type of target variable, i.e., if it has continuous or discrete values.

## 2. Problem complexity

How complicated is the relationship between the input features and target variable: linear or nonlinear?

→ Depends on the available data, i.e., how easily the target can be predicted from the inputs.

## 3. Algorithmic approach

Which type of model works best for this dataset size & complexity: features-based or similarity-based?

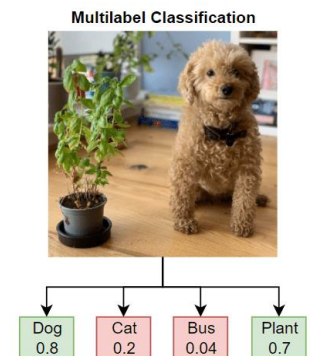
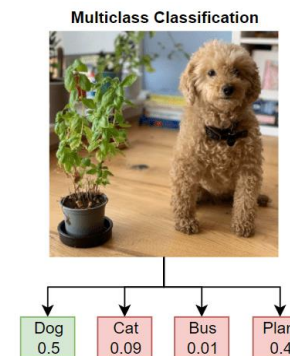
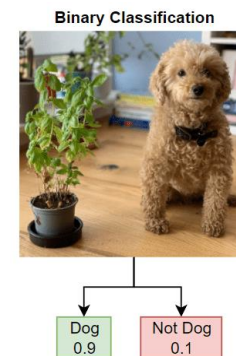
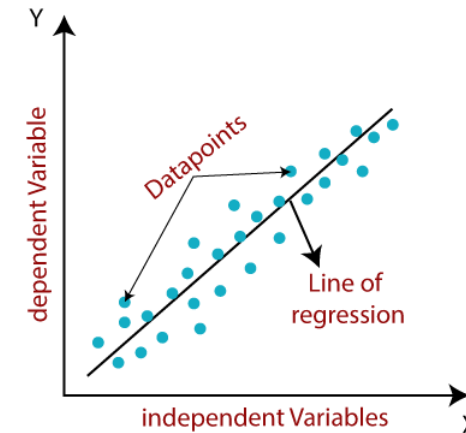
→ Depends on the model you choose, i.e., it either learns according to the first or second strategy.

# Different Types of Models: Problem Type

## <Regression vs. Classification>

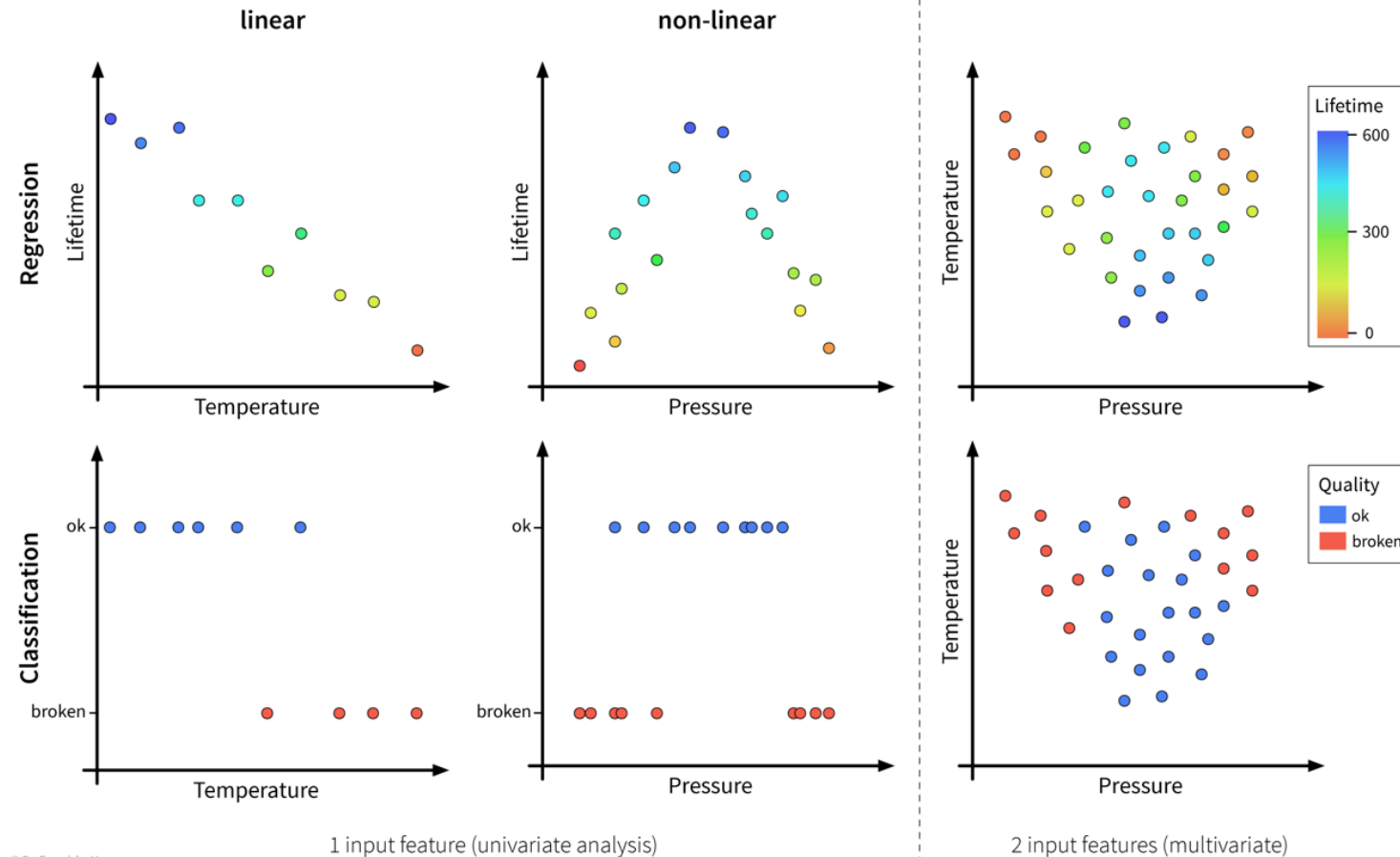
The type of the target variable that we want to predict determines whether we are dealing with a regression or classification problem.

- **Regression:** Prediction of continuous value(s) (e.g., price, number of users, etc.).
- **Classification:**
  - Prediction of discrete values:
    - binary (e.g., product will be faulty: yes/no)
    - multi-class (e.g., picture displays cat/dog/house/car/... )
    - multi-label (e.g., picture may display multiple objects)



Source: [mathworks.com](https://www.mathworks.com)

# Different Types of Models: Problem Complexity <Linear or nonlinear>



© Dr. Franziska Horn

Source: [A Practitioner's Guide to Machine Learning](#)

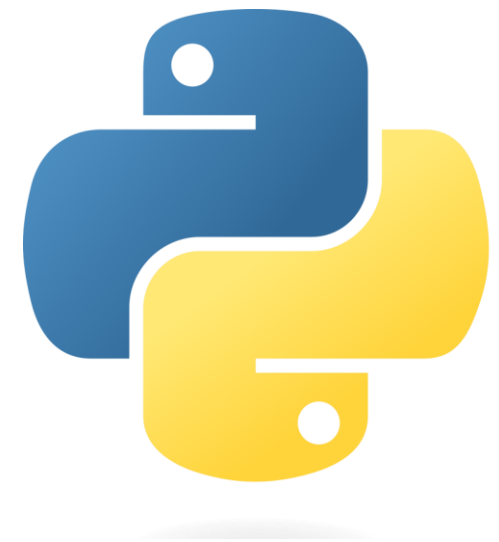
# Machine Learning with Python

## Why Python?

- free & open source (unlike, e.g., MatLab)
- easy; fast prototyping
- general purpose language (unlike, e.g., R): easy to incorporate ML into regular applications or web apps
- fast: many numerical operations are backed with C libraries
- a lot of open-source ML libraries with a very active community

## How?

- regular scripts (i.e., normal text files ending in .py), especially useful for function definitions that can be reused in different projects
- iPython shell: interactive console to execute code
- Jupyter Notebooks (i.e., special files ending in .ipynb): great for experimenting & sharing work with others (also works with other programming languages: Jupyter stands for Julia, Python, and R; you can even mix languages in the same notebook)

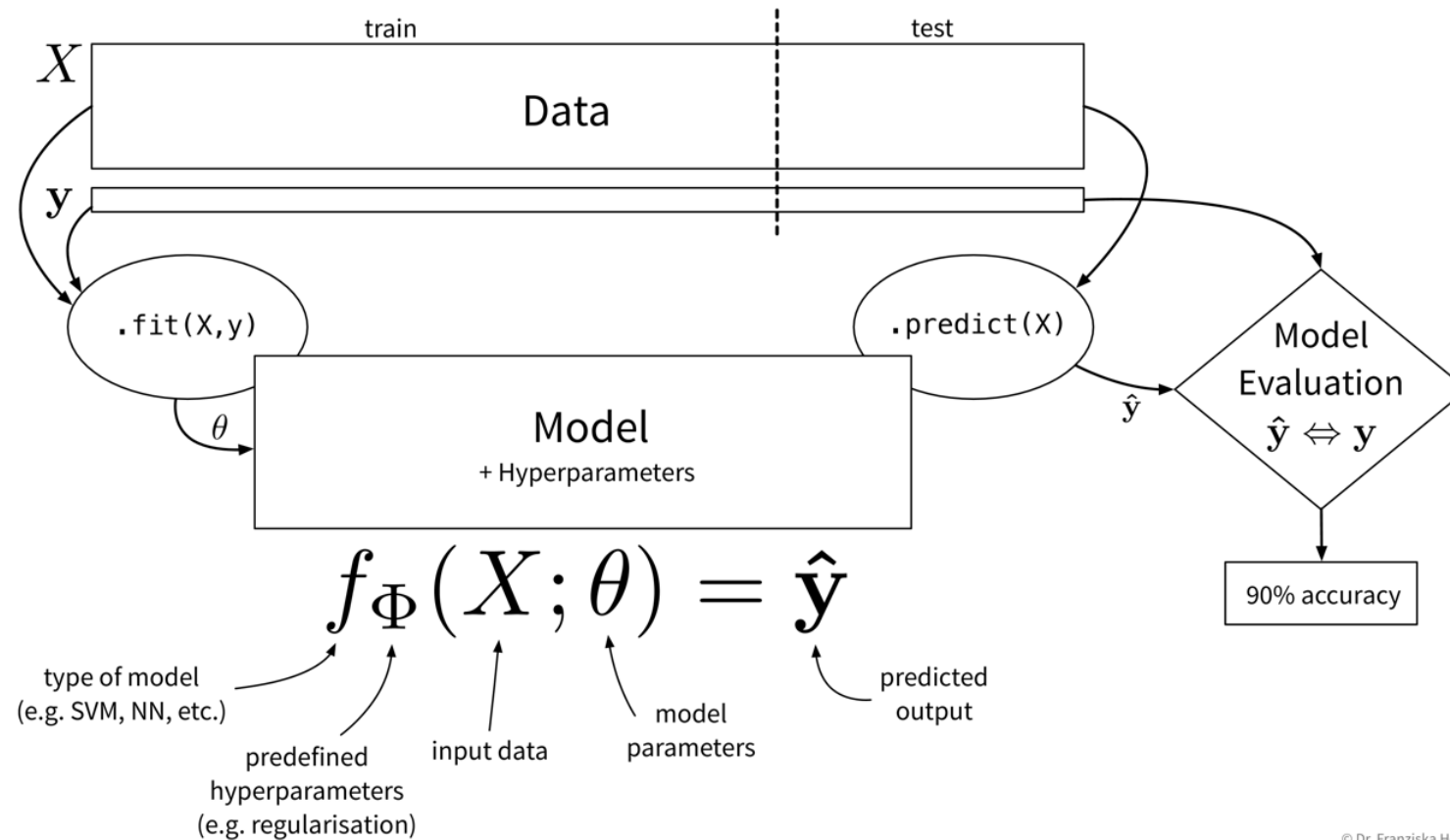


## Machine Learning with Python: Scikit-learn Library

- **Scikit-learn** (also known as sklearn) is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support-vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.
- It includes a lot of **(non-deep learning)** machine learning algorithms, preprocessing tools, and evaluation functions with an unified interface, i.e., all models (depending on their type) have these `.fit()`, `.transform()`, and/or `.predict()` methods, which makes it very easy to switch out models in the code by just changing the line where the model was initialized
- scikit-learn integrates well with many other Python libraries, such as Matplotlib and plotly for plotting, NumPy for array vectorization, Pandas dataframes, SciPy, and many more.



# Machine Learning with Python: Scikit-learn Library



© Dr. Franziska Horn

## Machine Learning with Python: Scikit-learn Library

```
# import the model class from the specific submodule
from sklearn.xxx import Model
from sklearn.metrics import accuracy_score

# initialize the model (usually we also set some parameters here)
model = Model()

# preprocessing/unsupervised learning methods:
model.fit(X) # only pass feature matrix X
X_transformed = model.transform(X) # e.g., the StandardScaler would return a scaled feature matrix

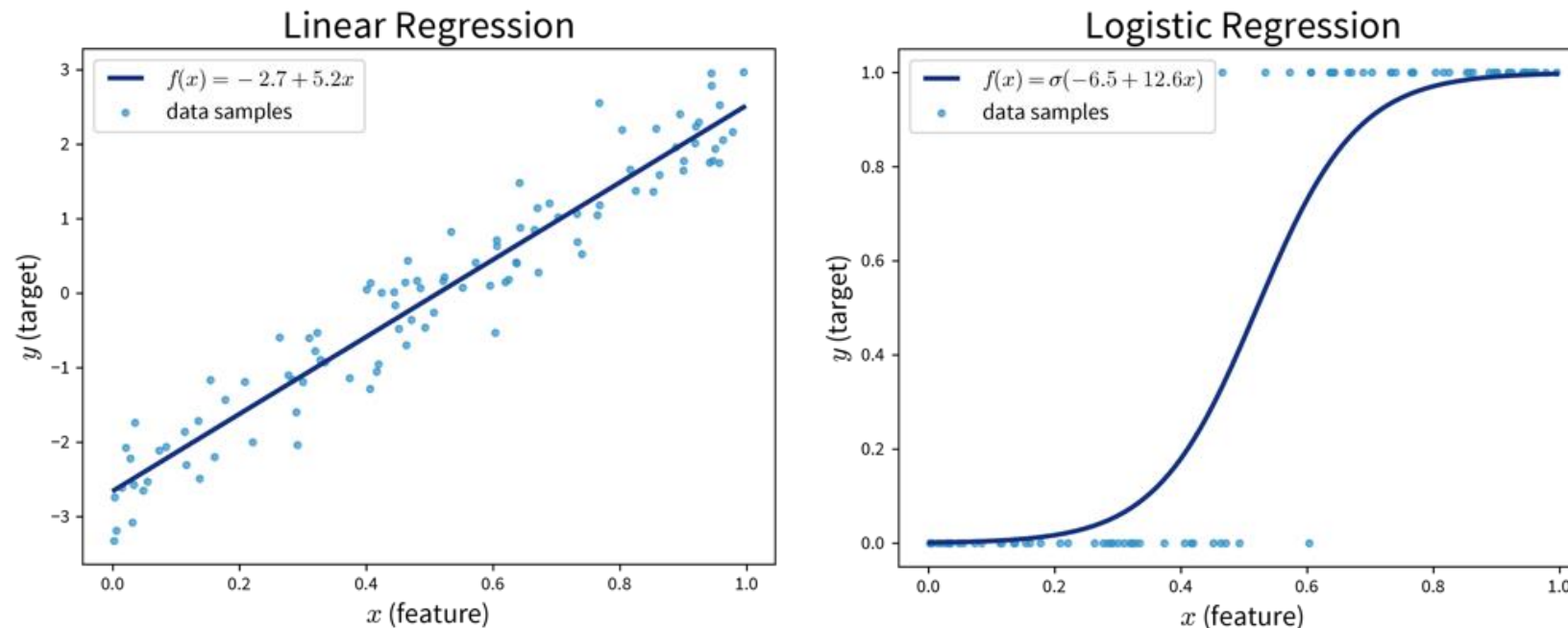
# supervised learning methods:
model.fit(X, y) # pass features and labels for training
y_pred = model.predict(X_test) # generate predictions for new points
# evaluate the model (the internal score function uses the model's preferred evaluation metric)
print("The model is this good:", model.score(X_test, y_test)) # .score() internally calls .predict()
print("Equivalently:", accuracy_score(y_test, y_pred))
```



# Supervised Learning Models

## <Linear Models>

The first type of supervised learning model that we'll look at in more detail are linear models, which are a type of features-based model that are very efficient (i.e., can be used with large datasets), but, as the name suggests, are only capable of describing linear relationships between the input and target variables.

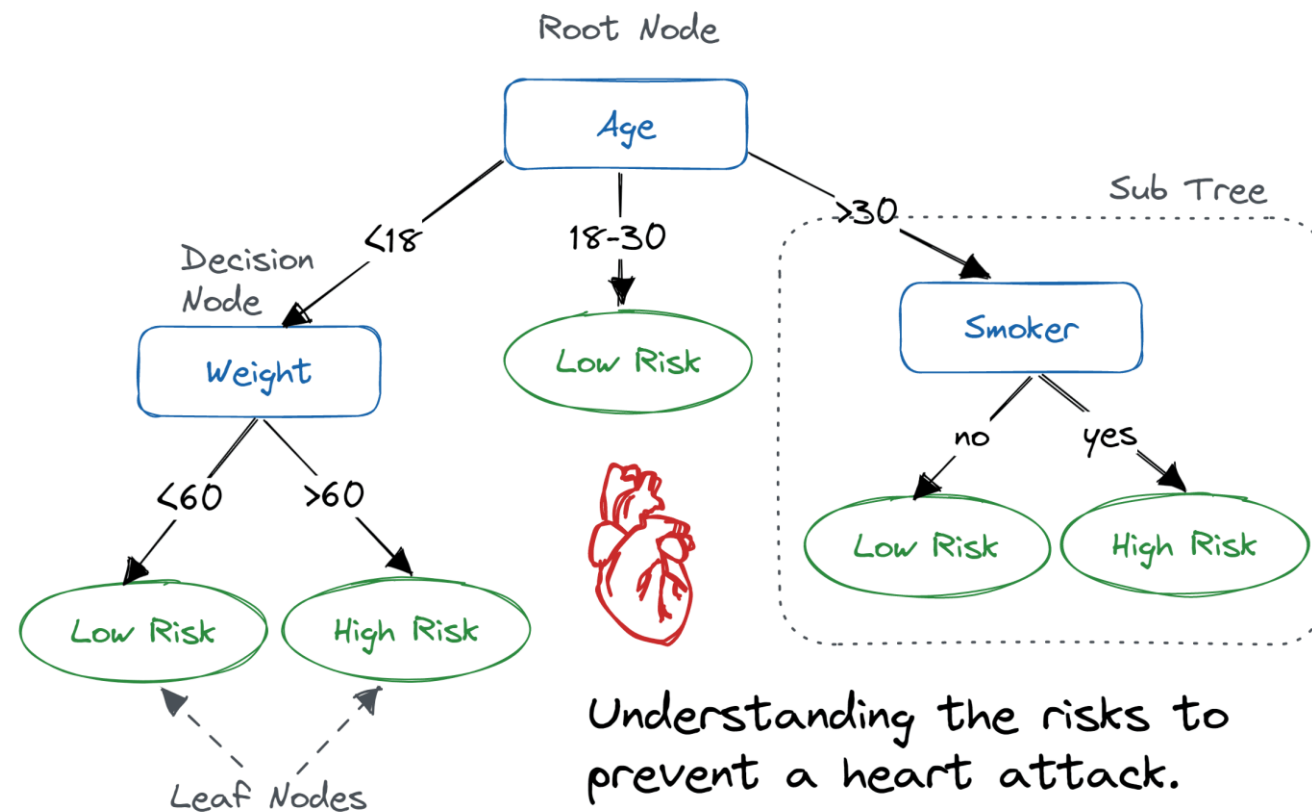


Source: [A Practitioner's Guide to Machine Learning](#)

# Supervised Learning Models

## <Decision Trees>

Creating a tree-like model to make decisions based on input features.

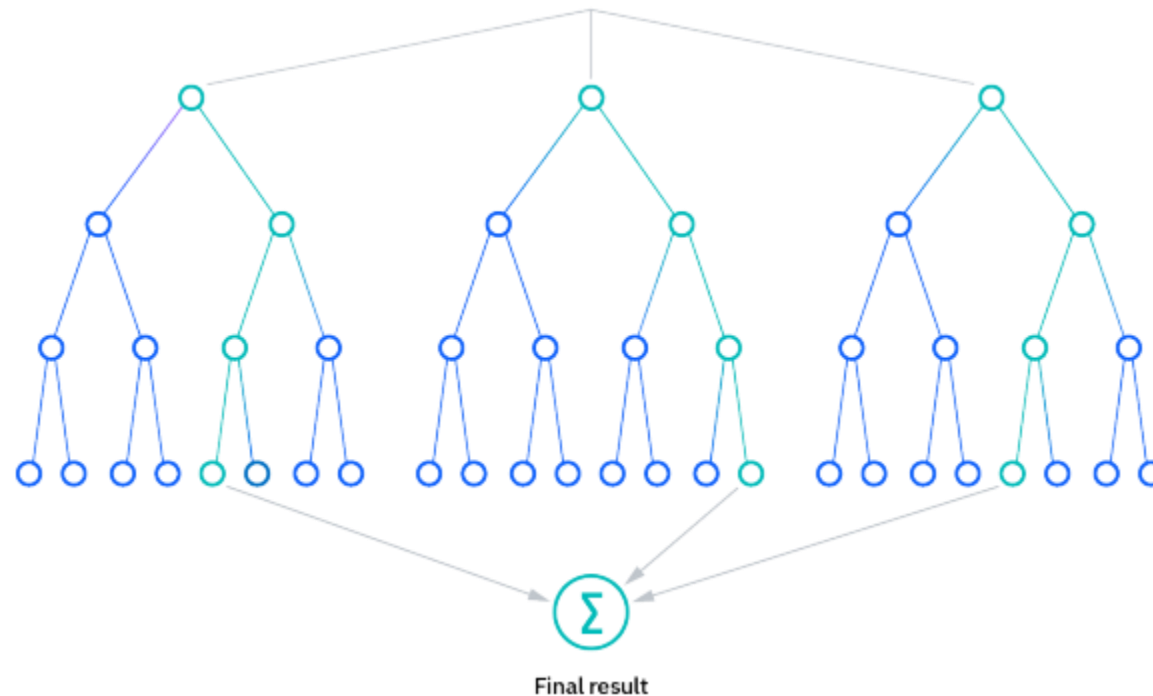


## Supervised Learning Models

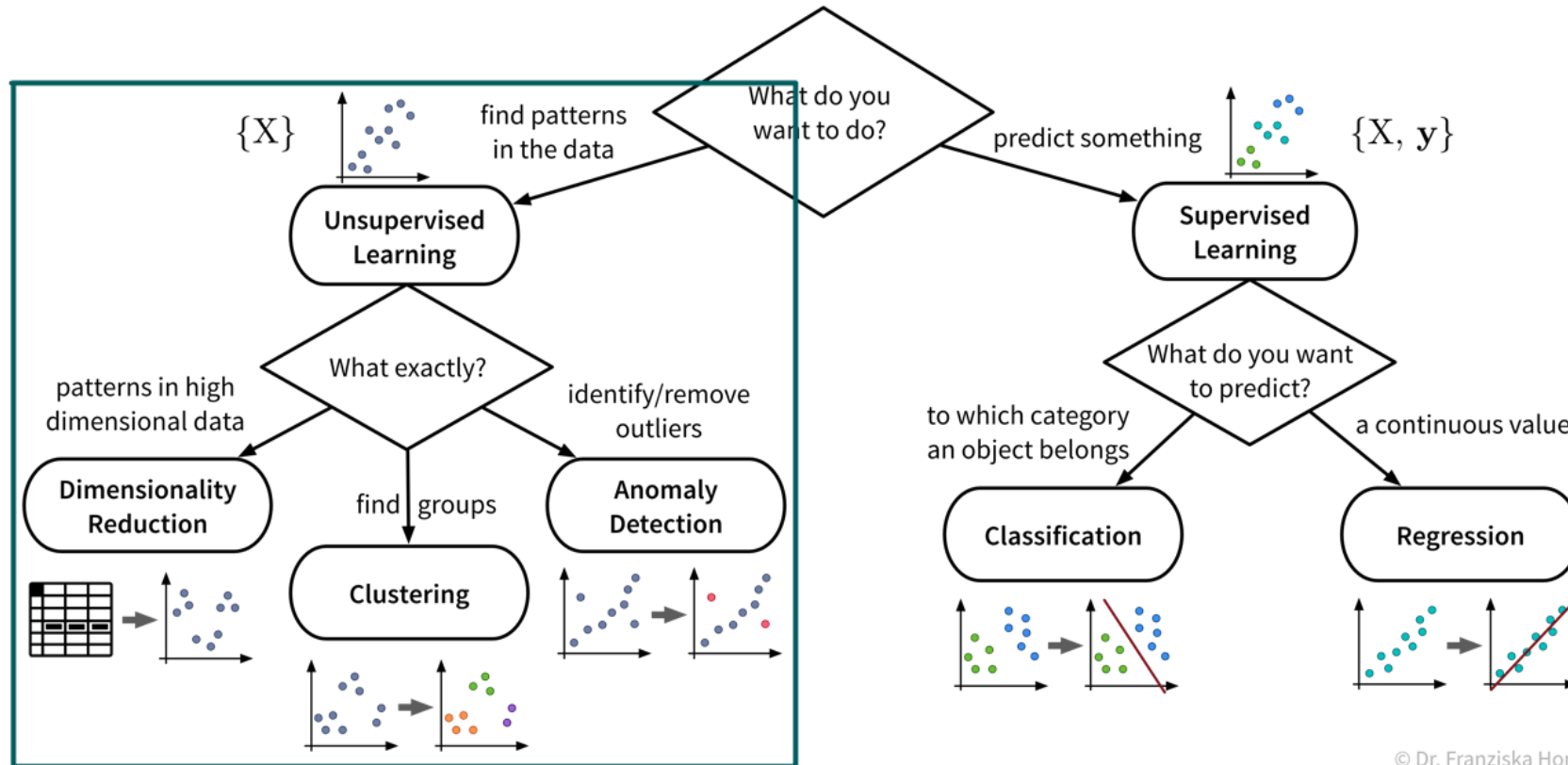
### <Ensemble Models: Random Forest>

What is better than one model? Multiple models!

Train multiple models & combine their predictions (regression: average; classification: most frequent class).



# Unsupervised Learning



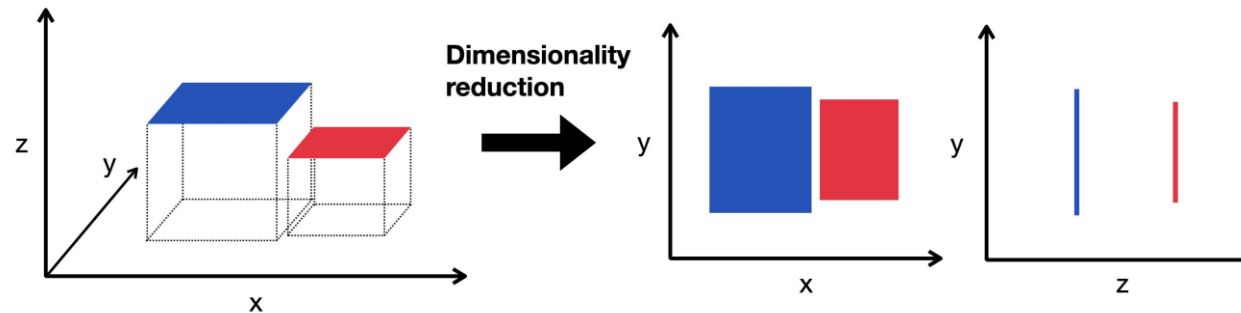
© Dr. Franziska Horn

# Unsupervised Learning: Dimensionality Reduction

**Goal:** Reduce the number of features without losing relevant information.

**Advantages:**

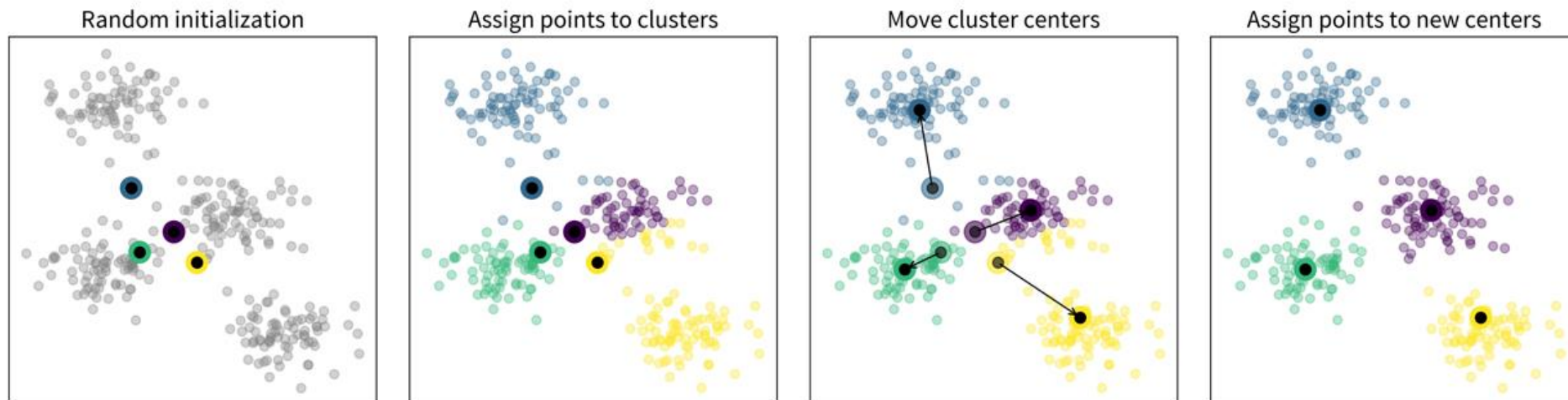
- Reduced data needs less memory (usually not that important anymore today)
- Noise reduction (by focusing on the most relevant signals)
- Create a visualization of the dataset (what we are mostly using these algorithms for)



## Unsupervised Learning: Clustering

Clustering is a type of unsupervised learning technique in machine learning where the goal is to group similar data points together based on certain characteristics or features. Unlike supervised learning, where the algorithm is trained on labeled data with predefined classes, clustering algorithms work on unlabeled data.

The main objective of clustering is to discover hidden patterns or structures within the data, allowing for the identification of natural groupings or clusters. Each cluster ideally contains data points that are more similar to each other than to those in other clusters.



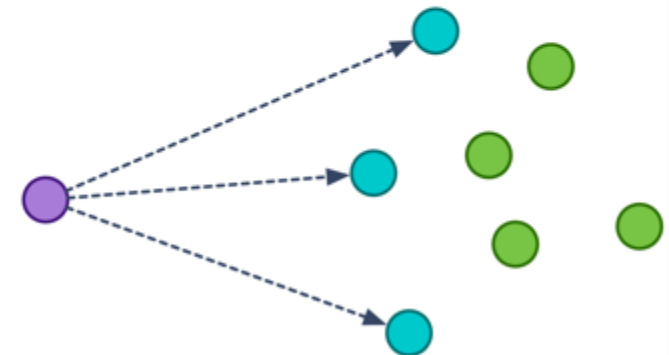
## Unsupervised Learning: Anomaly/Outlier Detection

Anomaly or outlier detection in unsupervised learning is the process of identifying instances in a dataset that deviate significantly from the norm, without relying on labeled data. In other words, the goal is to discover patterns or observations that are different from the majority of the data. Anomalies, also referred to as outliers, can represent unexpected events, errors, or rare occurrences in the dataset.

small average distance  
to nearest neighbors



outlier



## Unsupervised Learning: Anomaly/Outlier Detection

### Different methods for outlier/anomaly detection:

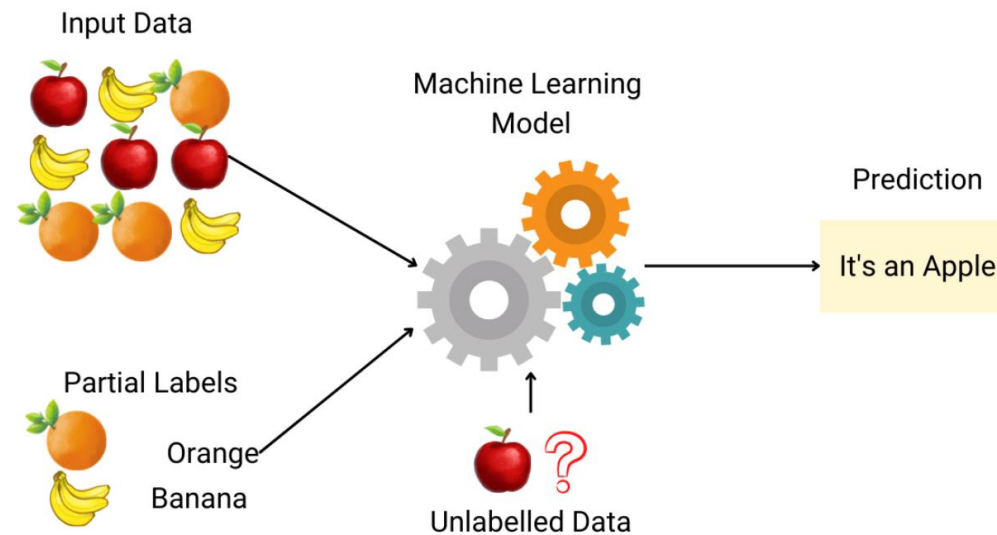
There is no one-size-fits-all solution, but depending on the characteristics of the dataset you might want to experiment with some of these approaches:

- **Detect outliers in individual feature dimensions:**
  - **Compute the interquartile range (IQR)**, i.e., the difference between the 75th and 25th percentiles of the data, as  $IQR = Q3 - Q1$ , where  $Q1$  and  $Q3$  are the lower and upper quartiles. All data points with values outside of the range  $[Q1 - 1.5 \cdot IQR, Q3 + 1.5 \cdot IQR]$  are considered outliers.
- **Detect multidimensional outlier patterns**, especially in datasets with independent data points:
  - $\gamma$ -index (Gamma Index)
  - Local Outlier Factor
  - Isolation Forest
  - One-class SVM
  - DBSCAN (a clustering algorithm that allows for noise, which would be considered outliers)



## Other Types of Machine Learning: Semi-Supervised Learning

Semi-supervised learning is a branch of machine learning that combines supervised and unsupervised learning, using both labeled and unlabeled data to train artificial intelligence (AI) models for classification and regression tasks.



## Other Types of Machine Learning: Semi-Supervised Learning

### Definition:

Uses a combination of labeled and unlabeled data for training.

### Motivation:

Labeled data is often expensive and time-consuming to acquire, while unlabeled data is more abundant.

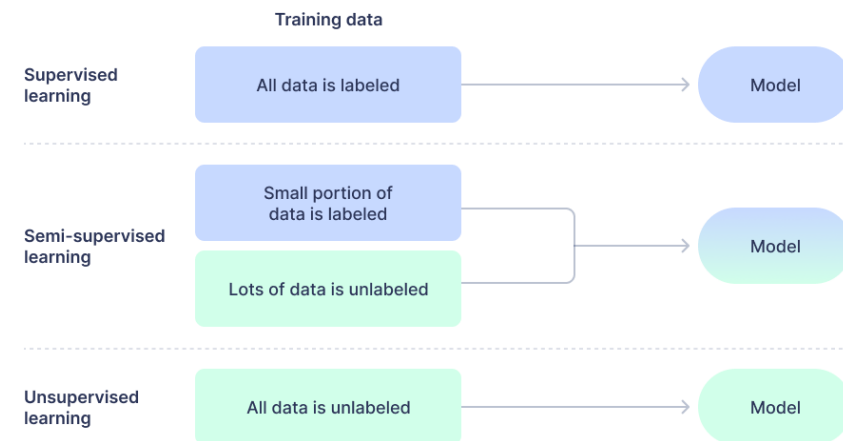
### Approaches:

- **Self-training:** Uses a model to make predictions on unlabeled data and adds confidently predicted instances to the labeled set.
- **Co-training:** Trains multiple models on different views of the data, each providing information on the unlabeled instances.

### Applications:

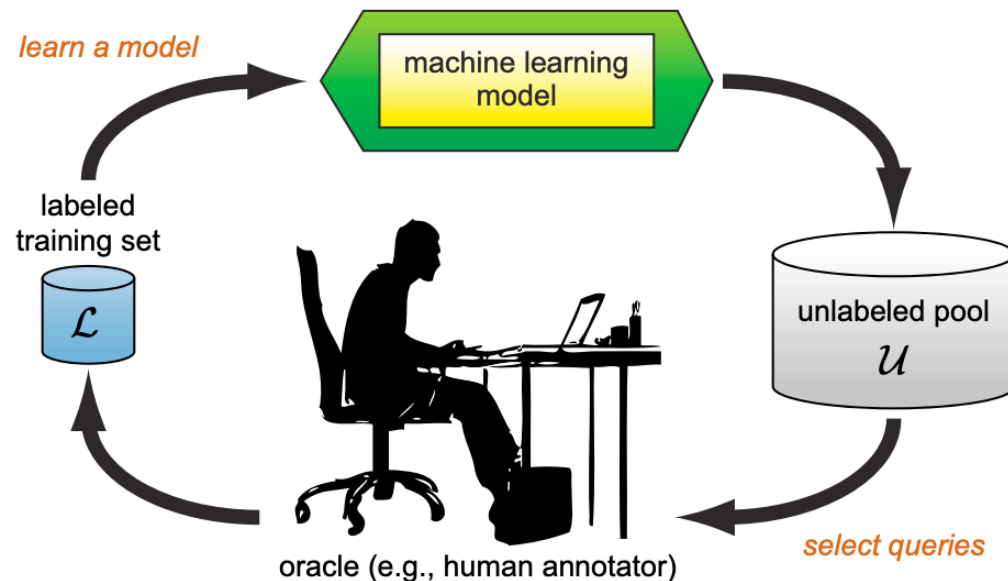
Often used in scenarios where acquiring labeled data is challenging, such as in medical imaging or natural language processing.

### Supervised learning vs Semi-supervised learning vs Unsupervised learning



## Other Types of Machine Learning: Active Learning

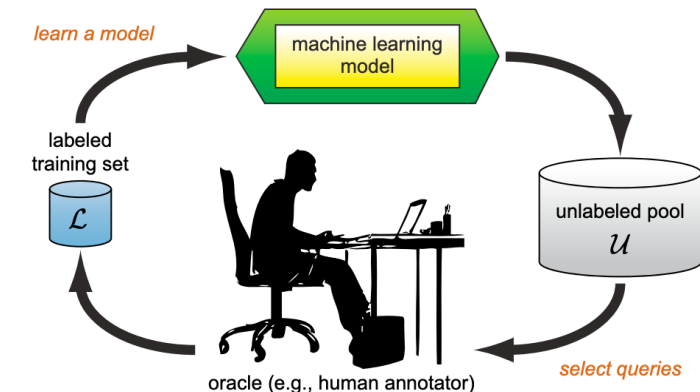
Active learning is a machine learning paradigm that involves an iterative process of selecting the most informative or uncertain data points for labeling and adding them to the training set. The goal is to improve the performance of a model with minimal labeled data. Instead of passively relying on a fixed dataset, active learning enables the model to actively query the user or a labeling oracle for the labels of the most valuable instances.



## Other Types of Machine Learning: Active Learning

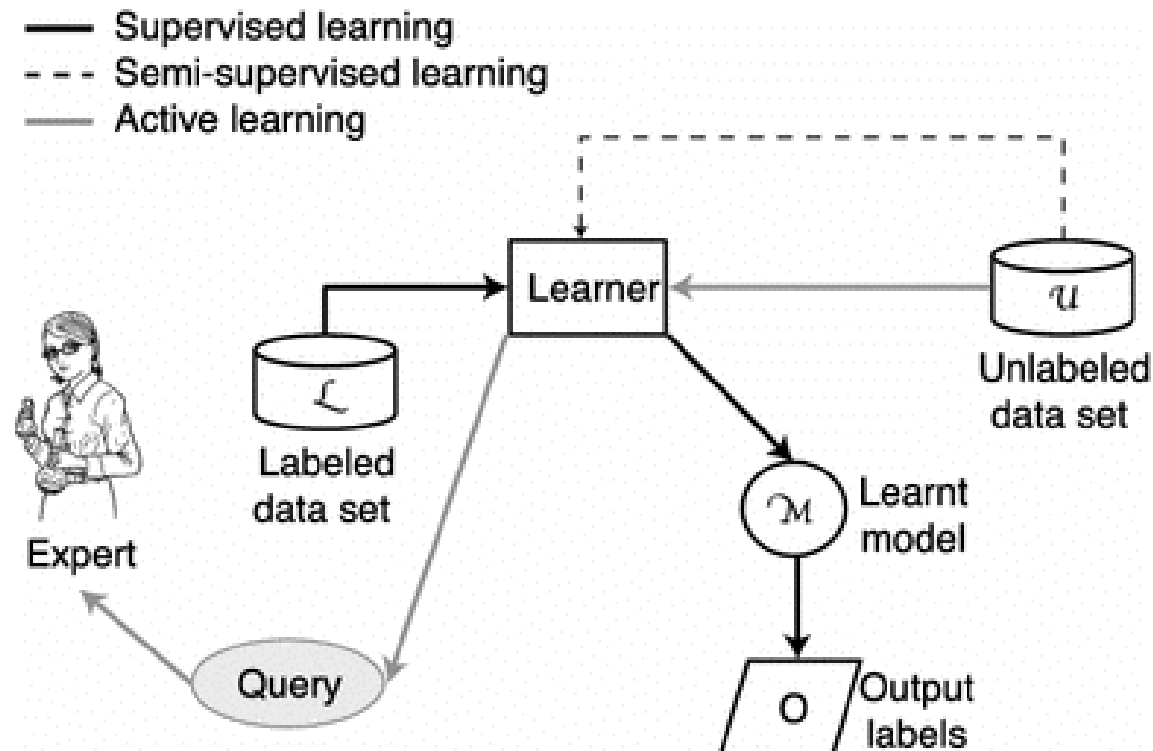
The key steps in an active learning process are as follows:

- **Initial Model Training:** Train an initial model using a small set of labeled examples.
- **Querying:** Identify the instances in the unlabeled dataset that are expected to provide the most valuable information for improving the model. These instances are typically chosen based on some measure of uncertainty or informativeness.
- **Labeling:** Request labels for the selected instances either from human annotators or some other means.
- **Updating the Model:** Incorporate the newly labeled instances into the training set and retrain the model.
- **Iteration:** Repeat the process by going back to step 2 until the model achieves satisfactory performance or resources are exhausted.



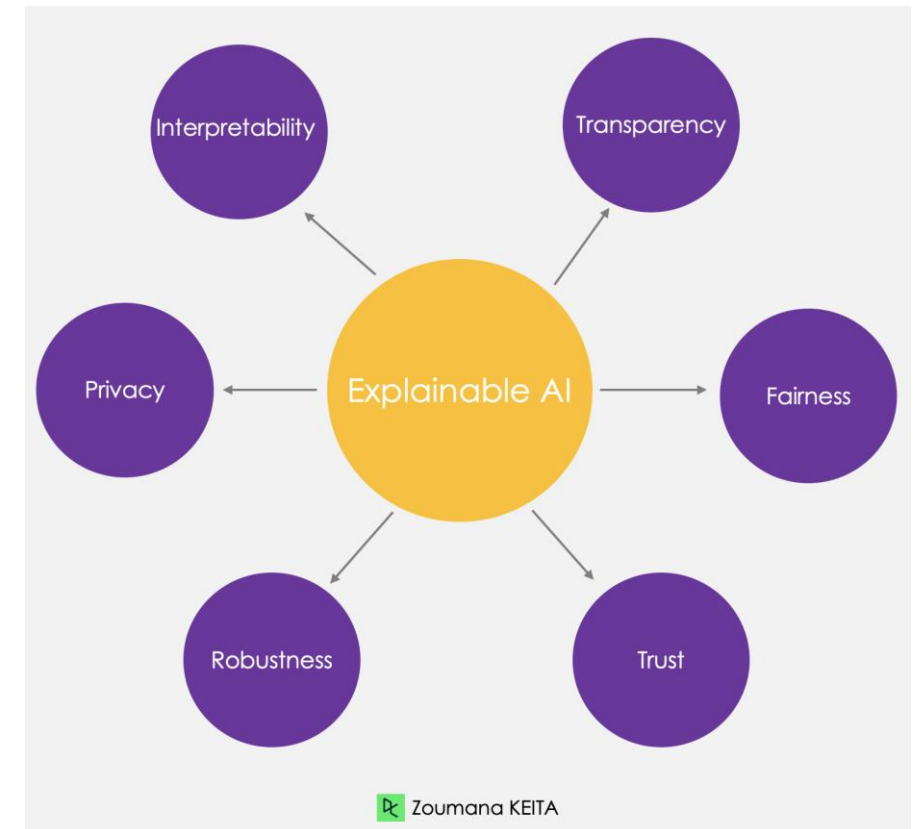
## Semi-Supervised and Active Learning <DEMO>

Demo: [Session 3 – Machine Learning Fundamentals](#)



## eXplainable Artificial Intelligence (XAI)

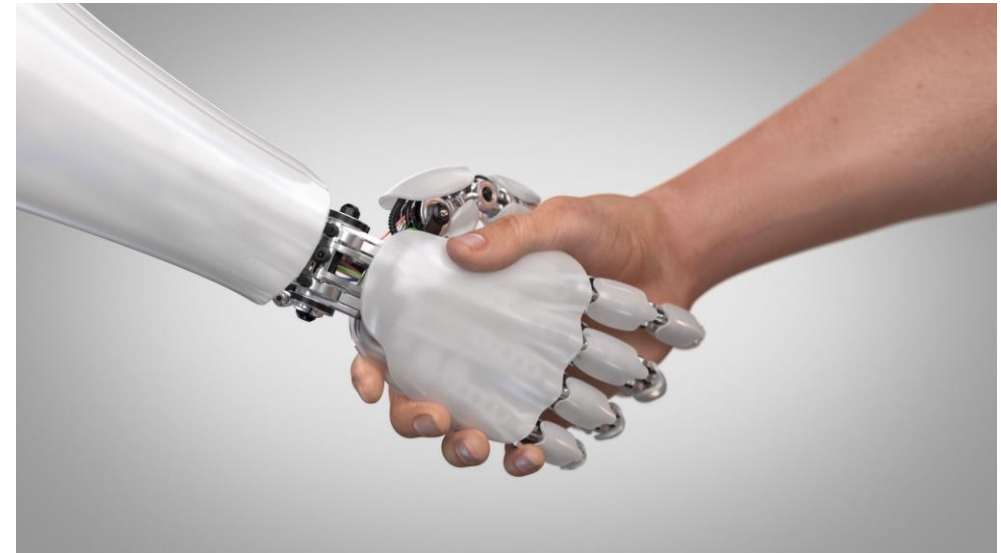
**Explainable AI** refers to a set of processes and methods that aim to provide a clear and human-understandable explanation for the decisions generated by AI and machine learning models. Integrating an explainability layer into these models, Data Scientists and Machine Learning practitioners can create more trustworthy and transparent systems to assist a wide range of stakeholders such as developers, regulators, and end-users.



**Goal of XAI:** The goal of XAI is to make AI systems more transparent and interpretable, enabling users to comprehend the rationale behind the AI's output.

## Why eXplainable AI (XAI)?

1. **Trust and Accountability:** In many applications, especially those with significant consequences such as healthcare, finance, and legal systems, users need to trust AI systems. Understanding how and why a model reaches a particular decision fosters trust and accountability.
2. **Legal and Ethical Compliance:** Regulations and ethical guidelines often require organizations to provide explanations for automated decisions. This is particularly crucial in situations where AI impacts individuals' rights and well-being.
3. **Bias and Fairness:** Explainability helps identify and address biases in AI models. It allows stakeholders to evaluate whether the model is making fair and unbiased decisions, and if not, take corrective measures.
4. **Human Understanding:** Explainable AI is essential for users who may not be experts in machine learning. It enables them to understand the decision-making process and ensures collaboration between AI systems and human users.



**Why should I trust you?**



## Techniques To Achieve explainability in AI

**Interpretable Models:** Choosing models that inherently provide transparency, such as decision trees or linear regression, can make it easier to understand how inputs contribute to outputs.

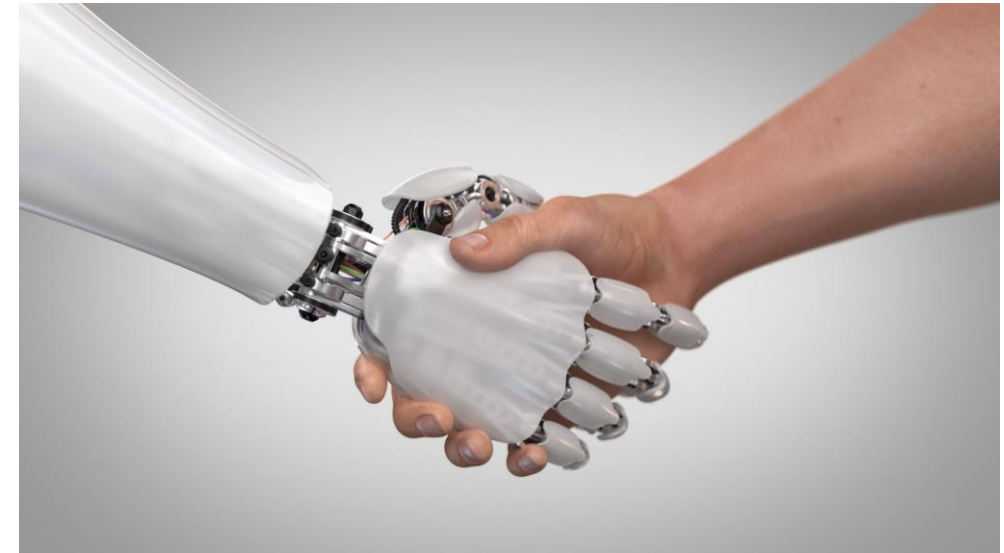
**Feature Importance:** Analyzing the importance of different features in the model's decision-making process helps users understand which inputs have the most significant impact.

**Local Explanations:** Providing explanations for individual predictions rather than the entire model output allows users to understand specific instances of decision-making.

**LIME (Local Interpretable Model-agnostic Explanations):** LIME is a technique that approximates the behavior of a black-box model using a simpler, interpretable model for specific instances.

**SHAP (SHapley Additive exPlanations):** SHAP values allocate contributions of each feature to the prediction, offering a comprehensive and theoretically grounded explanation.

**Visualization:** Graphical representations, such as heatmaps or saliency maps, can help users visually understand the parts of input data that influenced the model's decision.

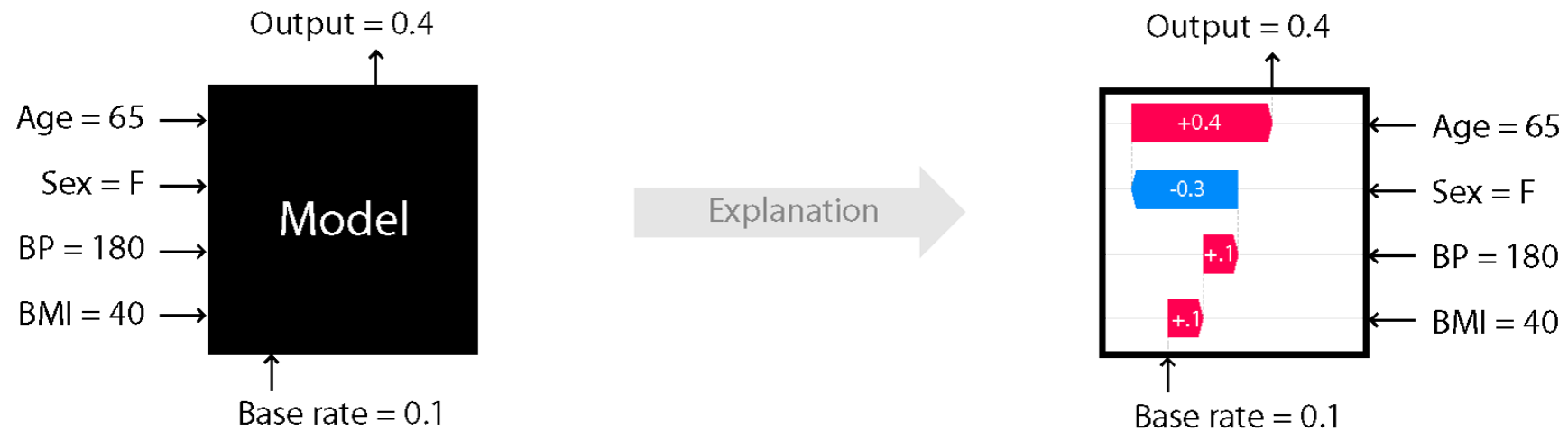


**Why should I trust you?**

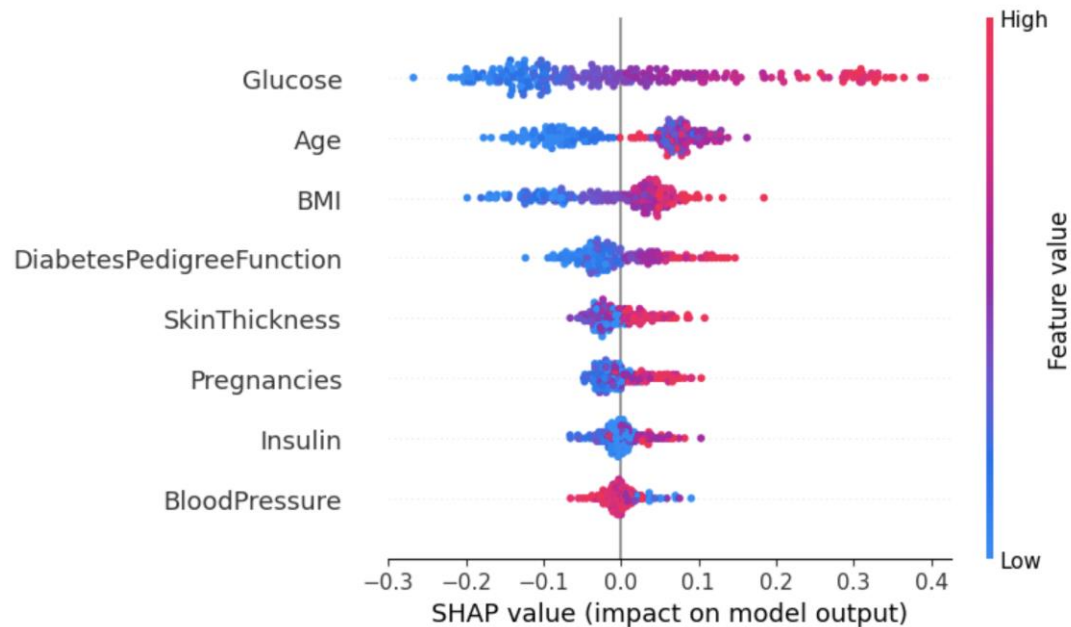
## Techniques To Achieve explainability in AI <SHAP>



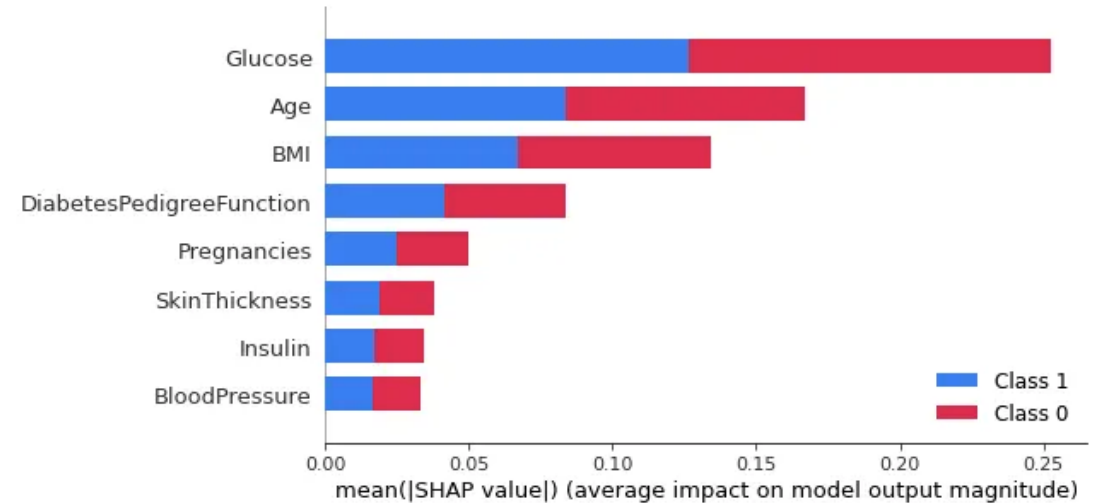
SHAP



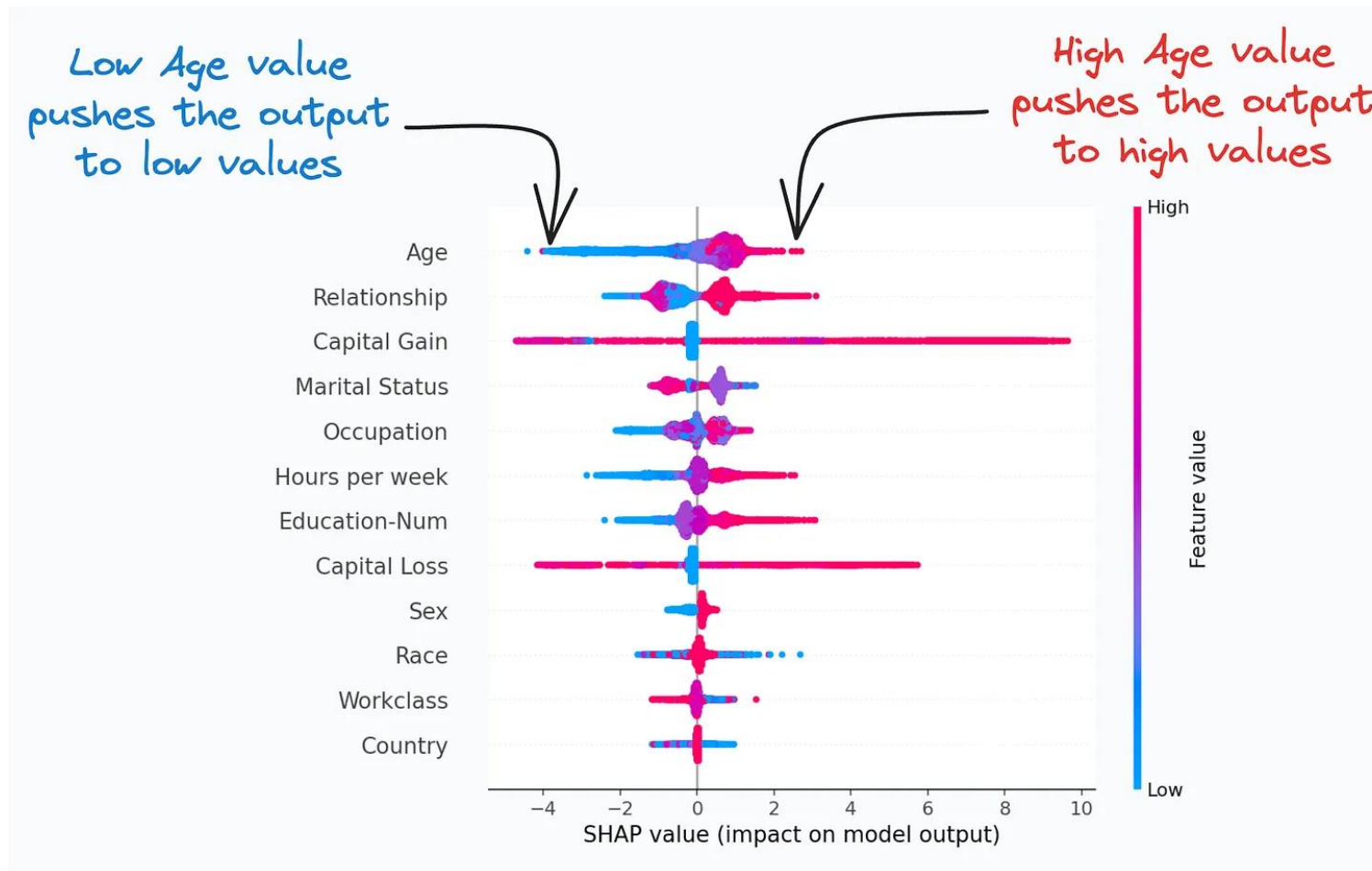
# Techniques To Achieve explainability in AI <SHAP>



Variable Importance Plot - Global Interpretation



## Techniques To Achieve explainability in AI <SHAP>



## eXplainable AI using SHAP < DEMO>

Demo: [Session 3 – Machine Learning Fundamentals](#)

