Introduction to Machine Learning

Guest Lectures at TTK University of Applied Sciences, Tallinn, Estonia

8 – 12 December 2025

Andy Louwyck & Dominique Stove

Vives University of Applied Sciences, Kortrijk, Belgium

TALLINNA TEHNIKAKÕRGKOOL
TTK UNIVERSITY OF APPLIED SCIENCES

vives University of Applied Sciences

Who’s Teaching Today?

Andy Louwyck

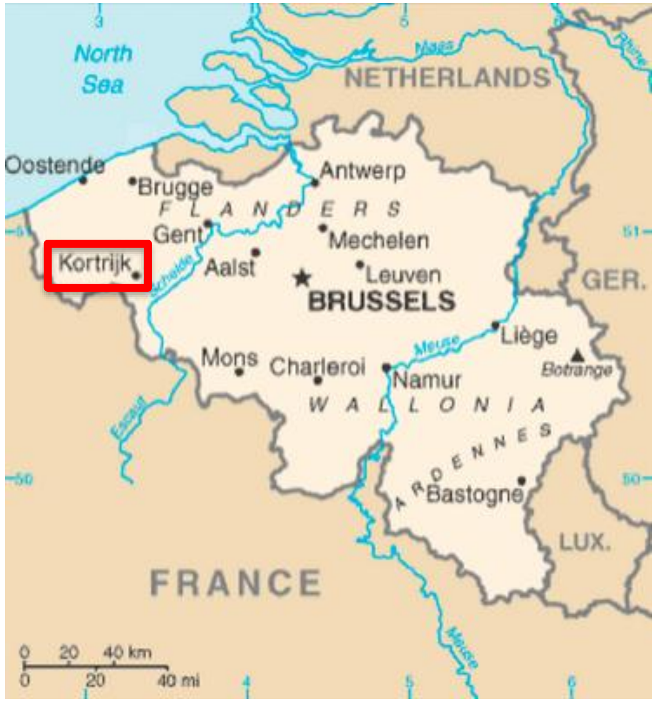
- Master and Doctor in Science: Geology
- Associate Degree in Programming
- Micro Degree in AI & Data Science
- Lecturer in IT at Vives University of Applied Sciences
- AI Coordinator at Flanders Environment Agency

Dominique Stove

- Master in Applied Economics
- Teaching Master’s Degree in Economics, Mathematics, and Physics
- Lecturer in IT at Vives University of Applied Sciences
- IT Consultant in Infrastructure
- Founder and Business Owner of IqPro

vives University of Applied Sciences

Vives Campus in the City of Kortrijk



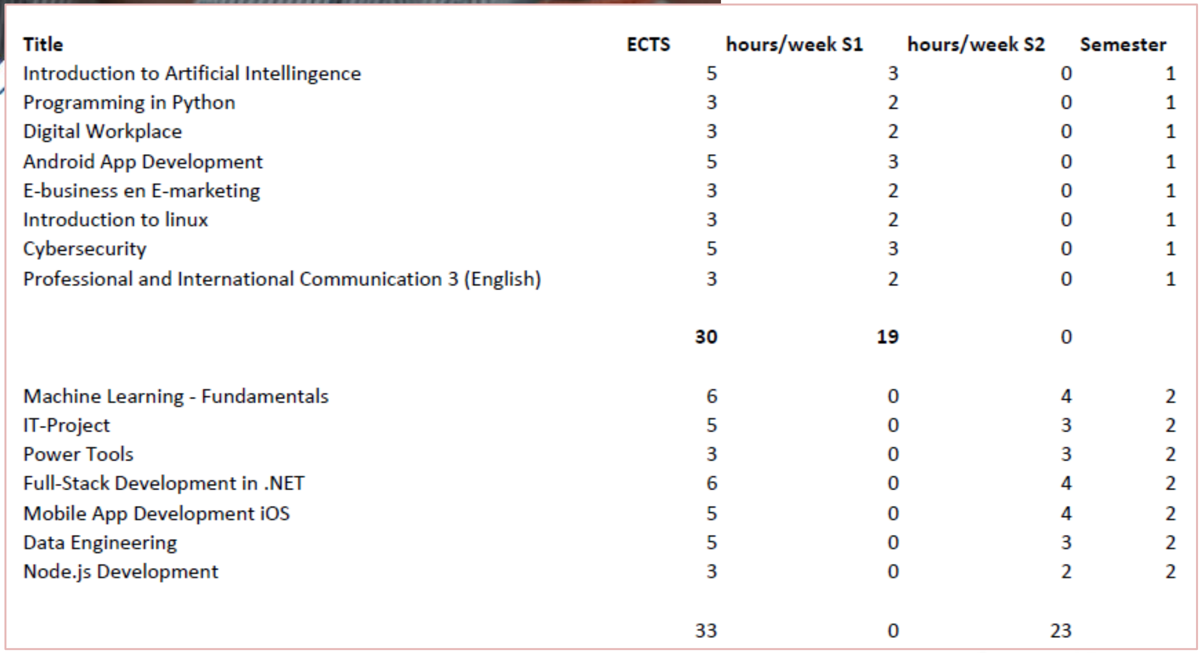
Informatics Program for Exchange Students

_INFORMATICS (KORTRIJK)
1st semester (Autumn) or 2nd semester (Spring) programme at VIVES Kortrijk

https://www.vives.be/en/commercial-sciences-business-management-and-informatics/informatics-kortrijk

The Informatics-programme is a programme consisting of lectures, group work, visits and projects in the field of Business and Informatics. Evaluation follows the rules of the European Credit Transfer System (ECTS). Incoming students can select a programme of up to 30 ECTS credits per semester.

New full-year program!



Title	ECTS	hours/week S1	hours/week S2	Semester
Introduction to Artificial Intellingence	5	3	0	1
Programming in Python	3	2	0	1
Digital Workplace	3	2	0	1
Android App Development	5	3	0	1
E-business en E-marketing	3	2	0	1
Introduction to linux	3	2	0	1
Cybersecurity	5	3	0	1
Professional and International Communication 3 (English)	3	2	0	1
	30	**19**	0	
Machine Learning - Fundamentals	6	0	4	2
IT-Project	5	0	3	2
Power Tools	3	0	3	2
Full-Stack Development in .NET	6	0	4	2
Mobile App Development iOS	5	0	4	2
Data Engineering	5	0	3	2
Node.js Development	3	0	2	2
	33	0	23	

Let’s Dive In!

1. What is Machine Learning?
2. Applications and tasks
3. Supervised learning
4. Unsupervised learning
5. Training and evaluation
6. Demo

Guest Lectures at TTK University of Applied Sciences, Tallinn, Estonia

Introduction to Machine Learning

WHAT IS MACHINE LEARNING?

Data

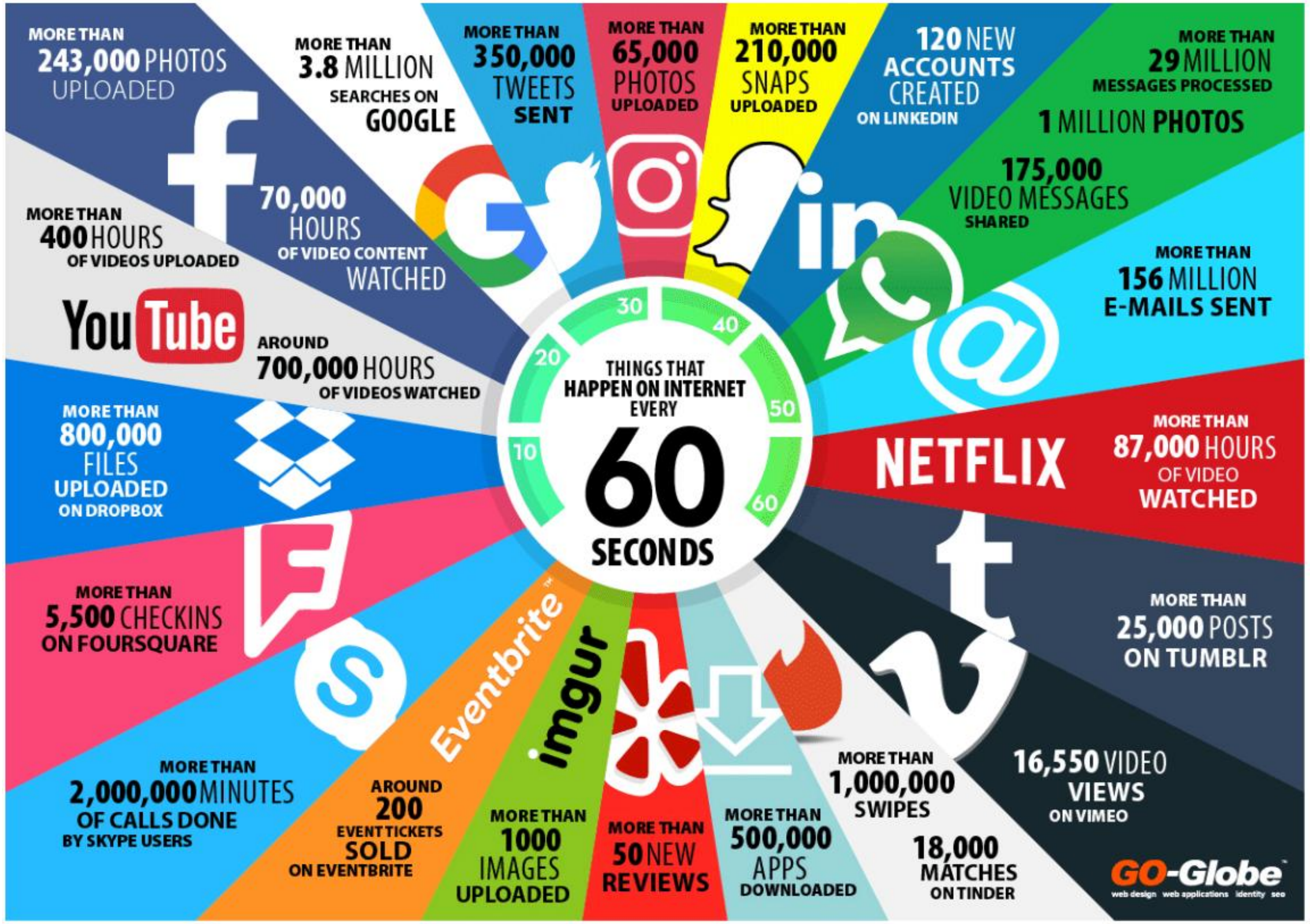
“We are drowning in data but starving for knowledge”

[Naisbitt, 1982]

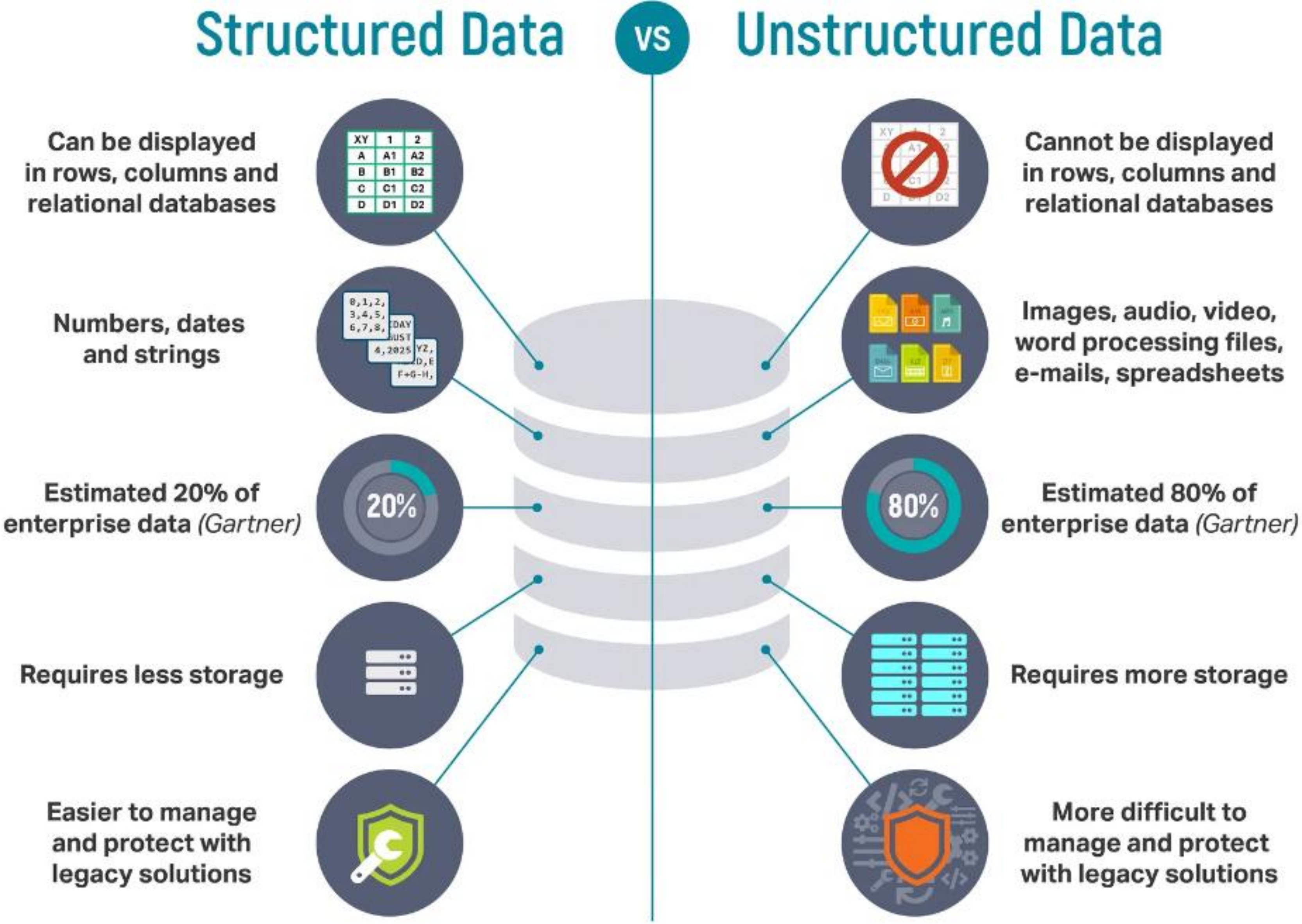
– A lot of data is gathered, but never used

– It is easier to generate data than to analyze data

→ **MACHINE LEARNING**



MORE THAN 243,000 PHOTOS UPLOADED
MORE THAN 3.8 MILLION SEARCHES ON GOOGLE
MORE THAN 350,000 TWEETS SENT
MORE THAN 65,000 PHOTOS UPLOADED
MORE THAN 210,000 SNAPS UPLOADED
120 NEW ACCOUNTS CREATED ON LINKEDIN
MORE THAN 29 MILLION MESSAGES PROCESSED
1 MILLION PHOTOS
175,000 VIDEO MESSAGES SHARED
MORE THAN 400 HOURS OF VIDEOS UPLOADED
70,000 HOURS OF VIDEO CONTENT WATCHED
YouTube
AROUND 700,000 HOURS OF VIDEOS WATCHED
MORE THAN 156 MILLION E-MAILS SENT
THINGS THAT HAPPEN ON INTERNET EVERY
60
SECONDS
10
20
30
40
50
60
MORE THAN 800,000 FILES UPLOADED ON DROPBOX
NETFLIX
MORE THAN 87,000 HOURS OF VIDEO WATCHED
MORE THAN 5,500 CHECKINS ON FOURSQUARE
MORE THAN 25,000 POSTS ON TUMBLR
Eventbrite
imgur
MORE THAN 2,000,000 MINUTES OF CALLS DONE BY SKYPE USERS
AROUND 200 EVENT TICKETS SOLD ON EVENTBRITE
MORE THAN 1000 IMAGES UPLOADED
MORE THAN 50 NEW REVIEWS
MORE THAN 500,000 APPS DOWNLOADED
MORE THAN 1,000,000 SWIPES
18,000 MATCHES ON TINDER
16,550 VIDEO VIEWS ON VIMEO
GO-Globe
web design web applications identity seo

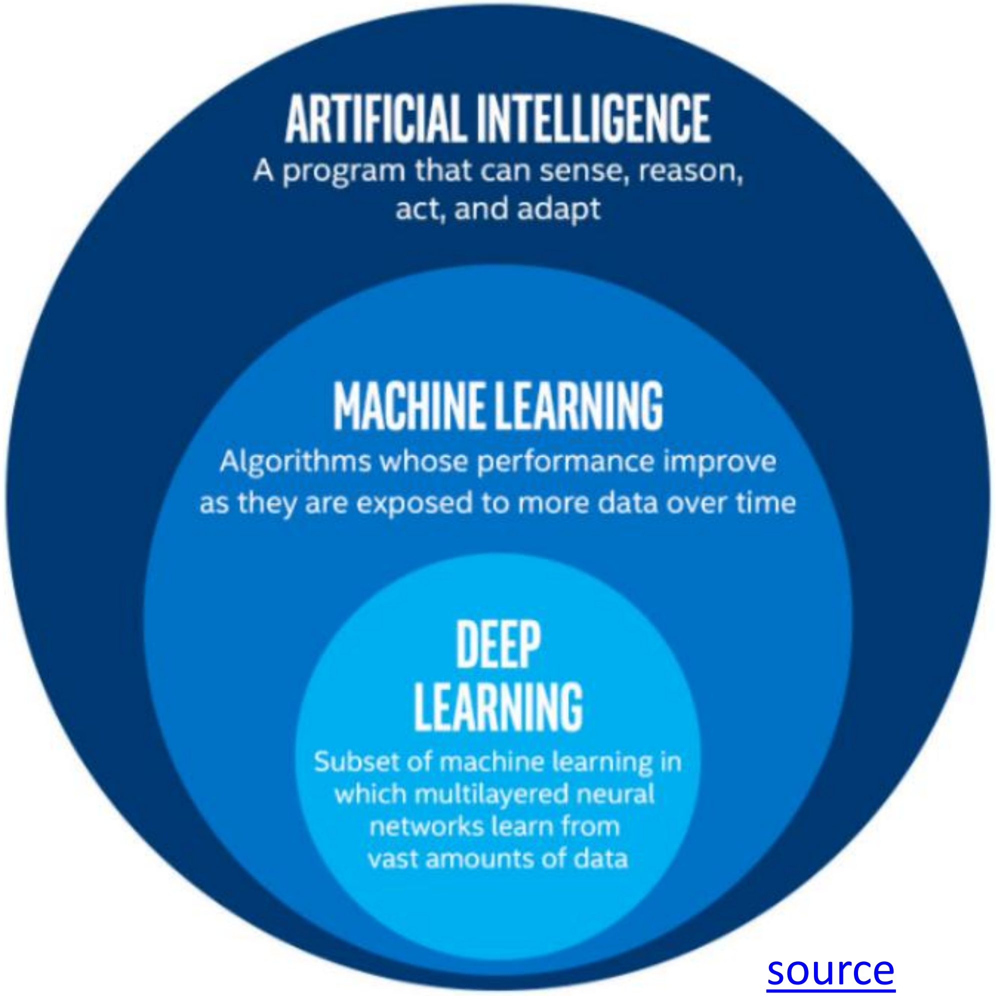


Structured Data VS Unstructured Data

Structured Data	Unstructured Data
Can be displayed in rows, columns and relational databases	Cannot be displayed in rows, columns and relational databases
Numbers, dates and strings	Images, audio, video, word processing files, e-mails, spreadsheets
Estimated 20% of enterprise data *(Gartner)*	Estimated 80% of enterprise data *(Gartner)*
Requires less storage	Requires more storage
Easier to manage and protect with legacy solutions	More difficult to manage and protect with legacy solutions

https://www.igneous.io/blog/structured-data-vs-unstructured-data

Machine Learning ⊂ Artificial Intelligence


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

source

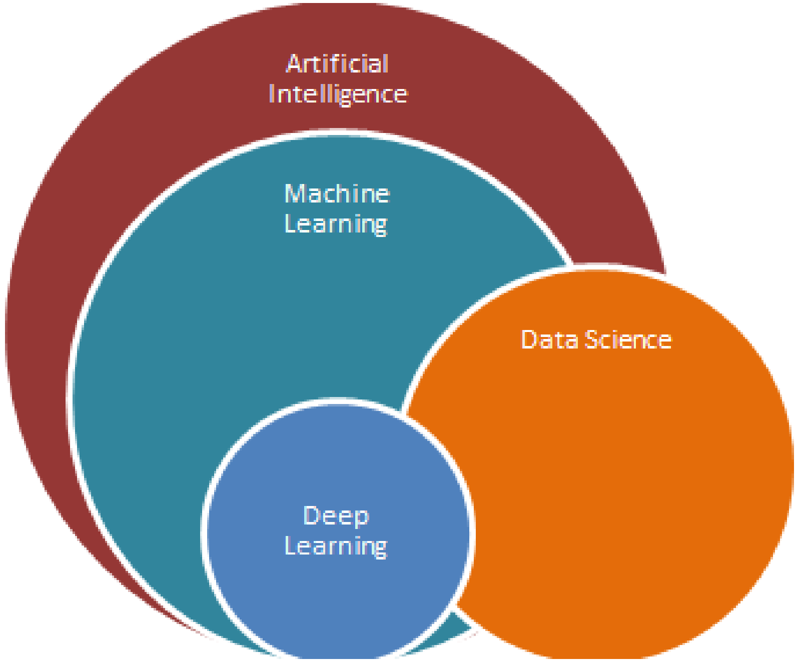
- **Artificial Intelligence** (AI):

  "The set of all tasks in which a computer can make *decisions*."
- **Machine Learning** (ML):

  "The set of all tasks in which a computer can make decisions *based on data*."
- **Deep Learning** (DL):

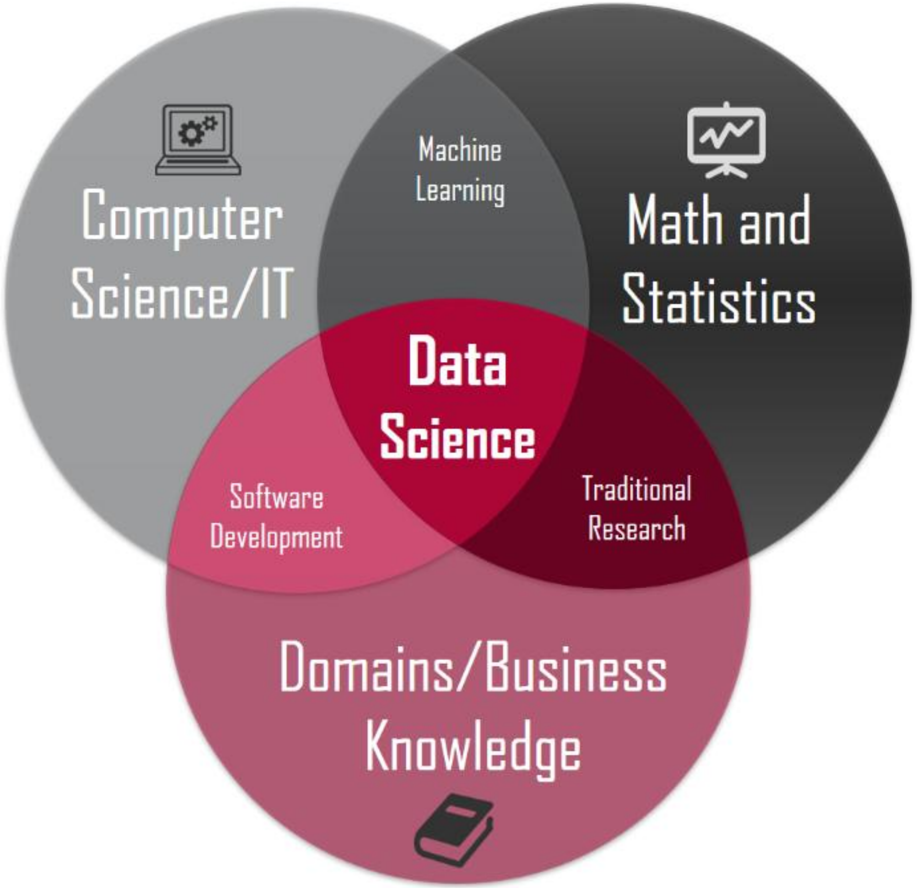
  "The field of machine learning that uses certain objects called *neural networks*."

Machine Learning ≠ Data Science



In practice:

- ML team: delivers software
- DS team: provides new insights

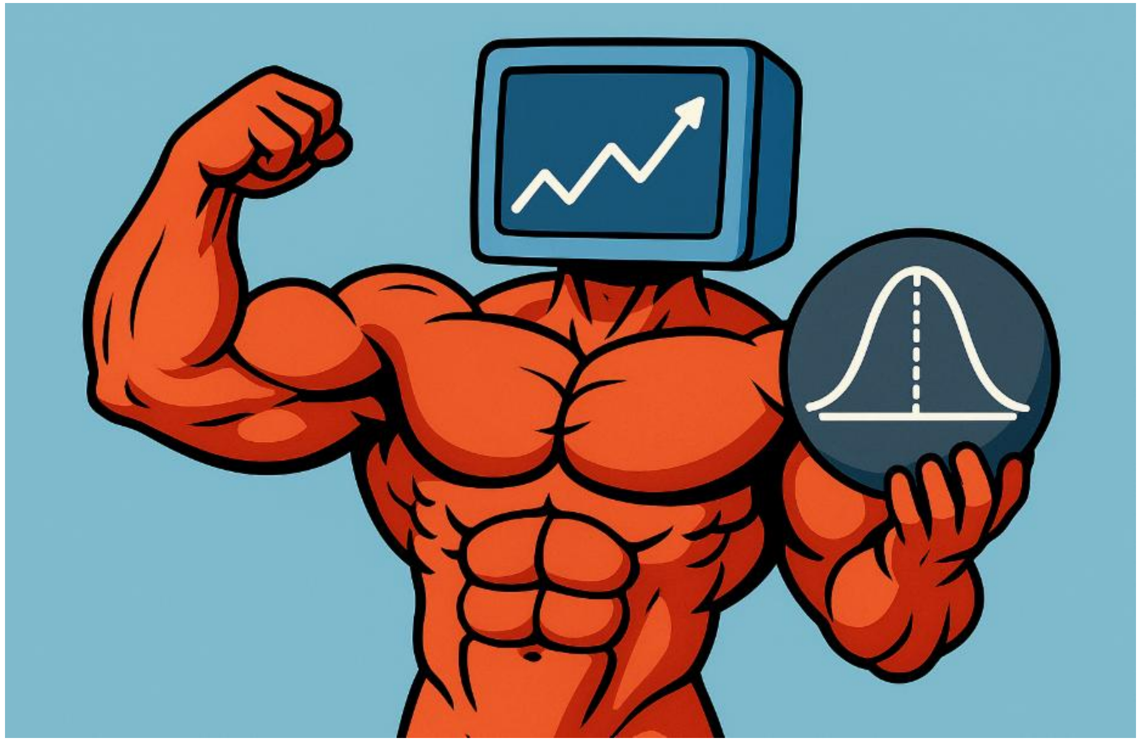


source

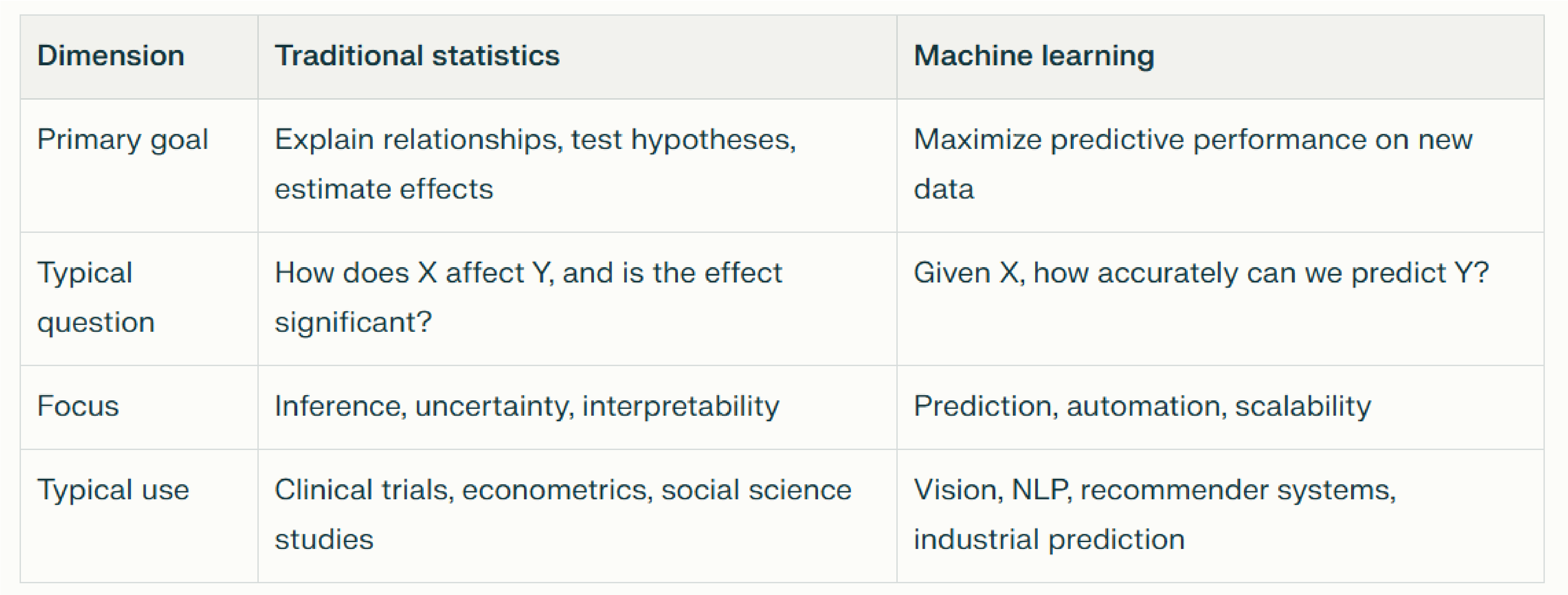
Machine Learning = Statistics on Steroids?

Machine learning takes **core statistical ideas** and supercharges them with **computational power** and **flexibility**:

- ML builds on the **foundations of statistics** but amplifies its capabilities.
- ML uses **larger datasets**, more complex models, and automated learning.
- ML focuses on **prediction and scalability**, often with less emphasis on interpretability.
- ML applies **fewer explicit assumptions**, letting algorithms discover patterns directly from data.
- ML excels in domains like **computer vision**, **NLP**, and industrial **prediction**.



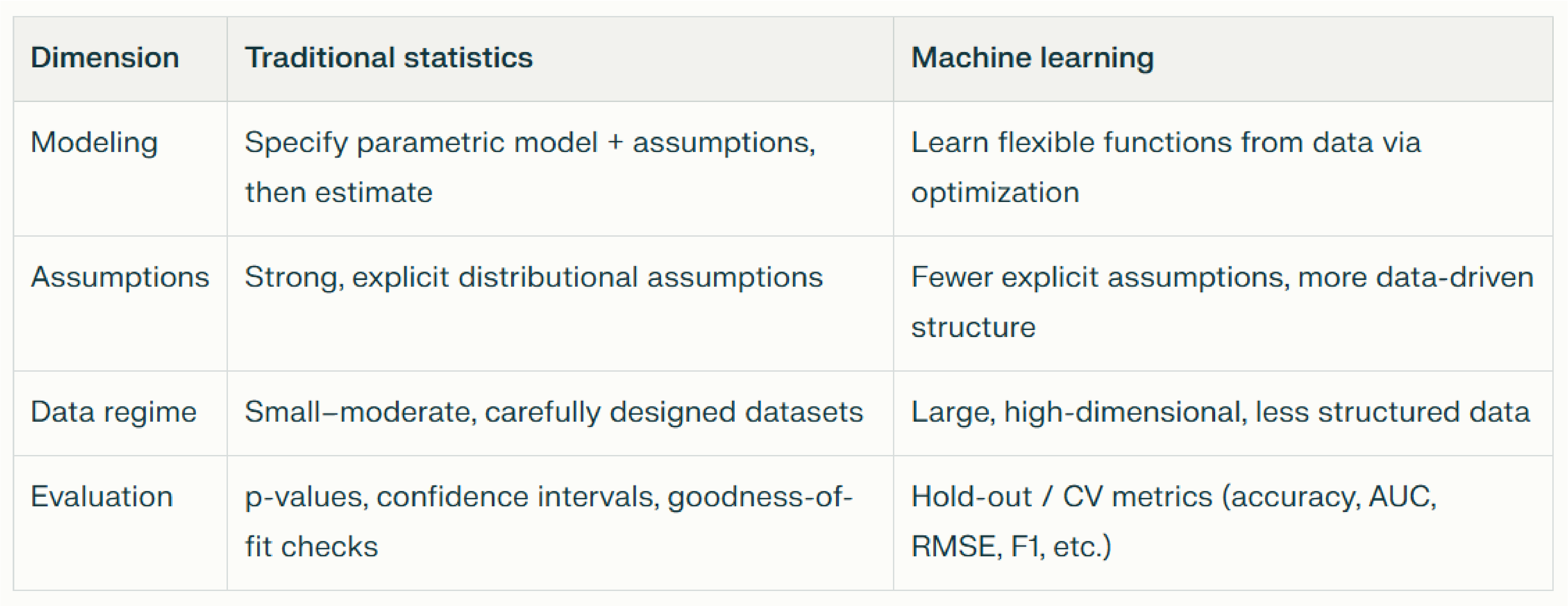
Machine Learning vs Statistics: Goals & Questions



Dimension	Traditional statistics	Machine learning
Primary goal	Explain relationships, test hypotheses, estimate effects	Maximize predictive performance on new data
Typical question	How does X affect Y, and is the effect significant?	Given X, how accurately can we predict Y?
Focus	Inference, uncertainty, interpretability	Prediction, automation, scalability
Typical use	Clinical trials, econometrics, social science studies	Vision, NLP, recommender systems, industrial prediction

source

Machine Learning vs Statistics: Modeling & Data



Dimension	Traditional statistics	Machine learning
Modeling	Specify parametric model + assumptions, then estimate	Learn flexible functions from data via optimization
Assumptions	Strong, explicit distributional assumptions	Fewer explicit assumptions, more data-driven structure
Data regime	Small–moderate, carefully designed datasets	Large, high-dimensional, less structured data
Evaluation	p-values, confidence intervals, goodness-of-fit checks	Hold-out / CV metrics (accuracy, AUC, RMSE, F1, etc.)

source

Intuitive Explanation of Machine Learning

I make **decisions** based on **experience**.
I make **prediction** based on **data**.

Example: buying a new car

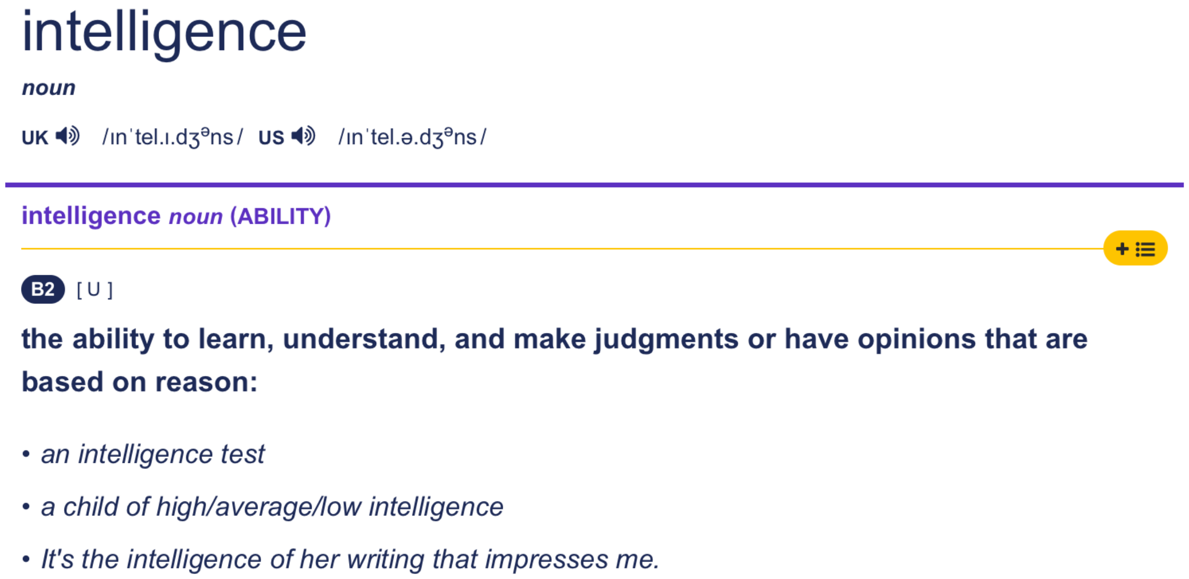
- How do we make decisions?
  - by logical **reasoning**
  - by relying on previous **experiences** (either our own or those of others)

- For a computer: **experiences = data**

“Machine learning is common sense, except done by a computer”

Formal Explanation of Machine Learning

- Core domain of AI, concerned with automatic learning



intelligence

noun

UK /ɪnˈtel.ɪ.dʒᵊns/ US /ɪnˈtel.ə.dʒᵊns/

intelligence ***noun*** **(ABILITY)**

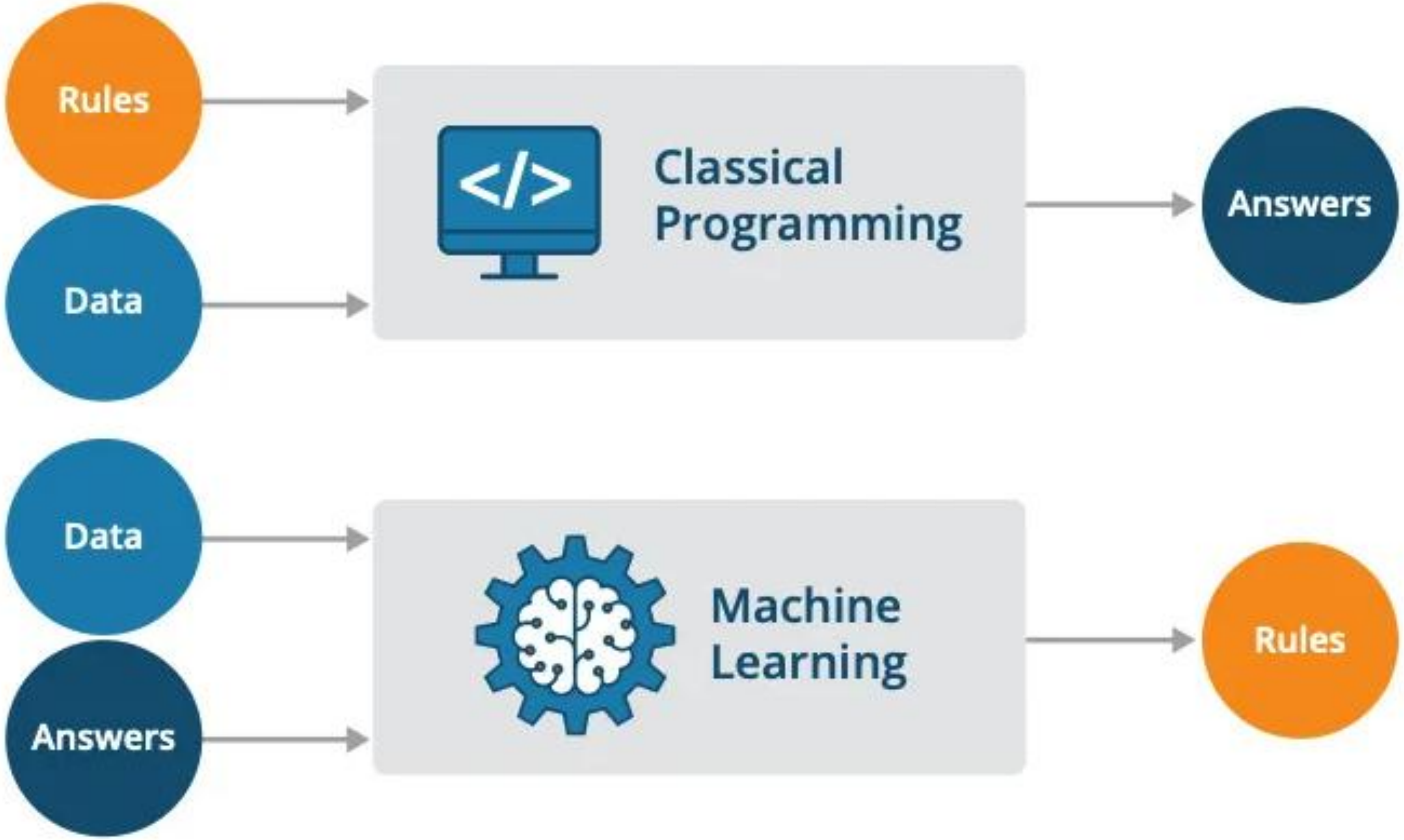
B2 [U]

the ability to learn, understand, and make judgments or have opinions that are based on reason:

- *an intelligence test*
- *a child of high/average/low intelligence*
- *It's the intelligence of her writing that impresses me.*

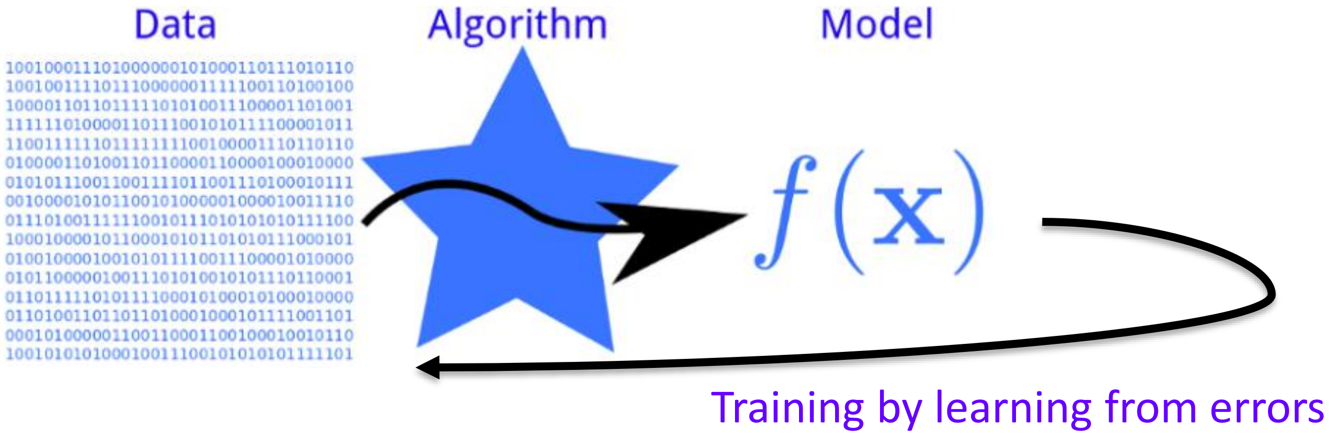
- A computer is said to be able to learn if its performance in solving some task improves with its experience

Machine Learning versus Classical Programming



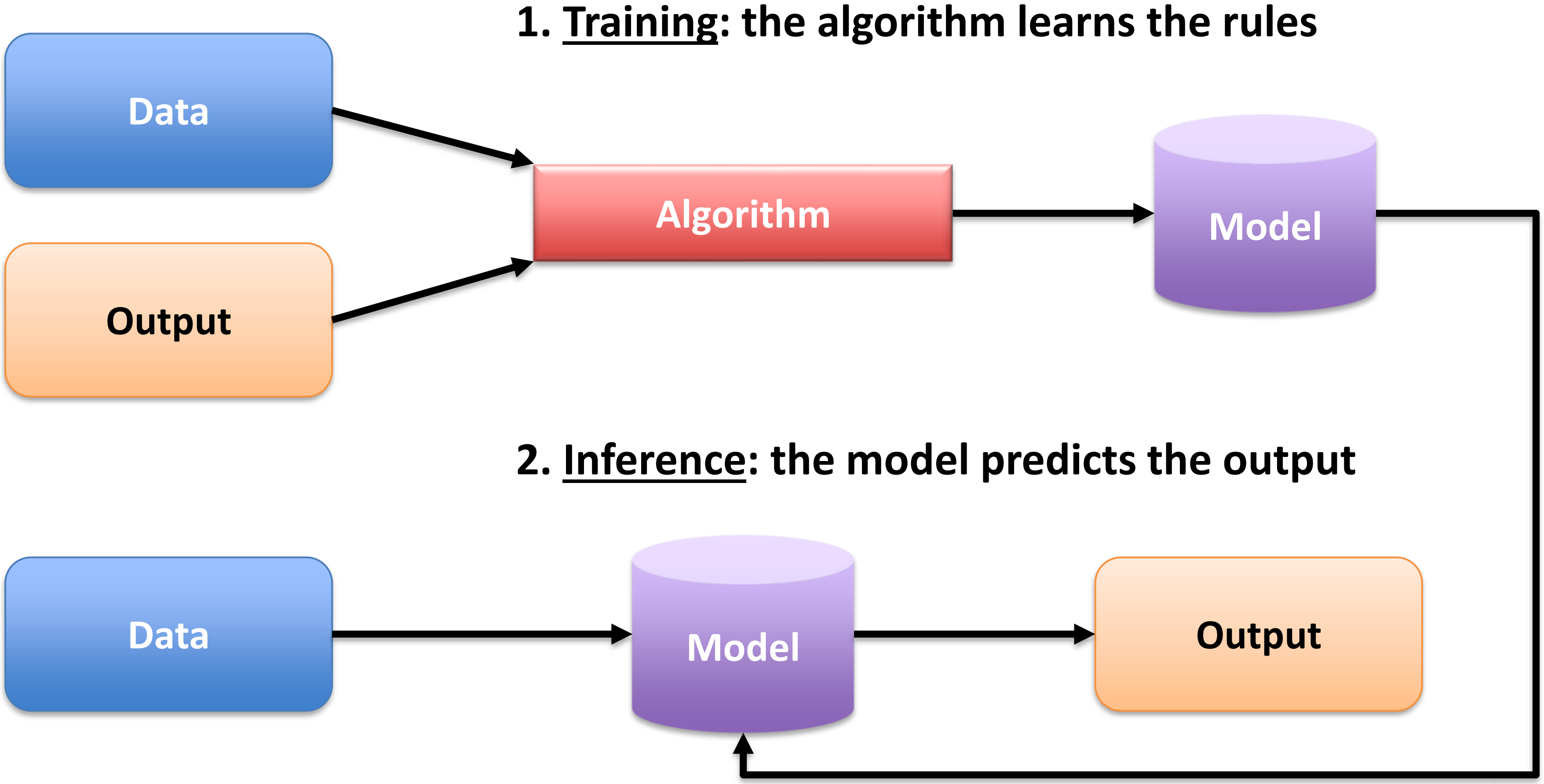
Machine Learning: Algorithm versus Model

- **Algoritme**: A procedure, or a set of steps, used to build a model.
- **Model**: A set of rules that represent the data and can be used to make predictions.
- **Learning**: The process in which the algorithm improves the model by analyzing errors.



*“An algorithm is run on the data to create a model.
By learning from errors, the model improves itself.”*

Machine Learning: Training versus Inference



1. Training: the algorithm learns the rules

Data

Output

Algorithm

Model

2. Inference: the model predicts the output

Data

Model

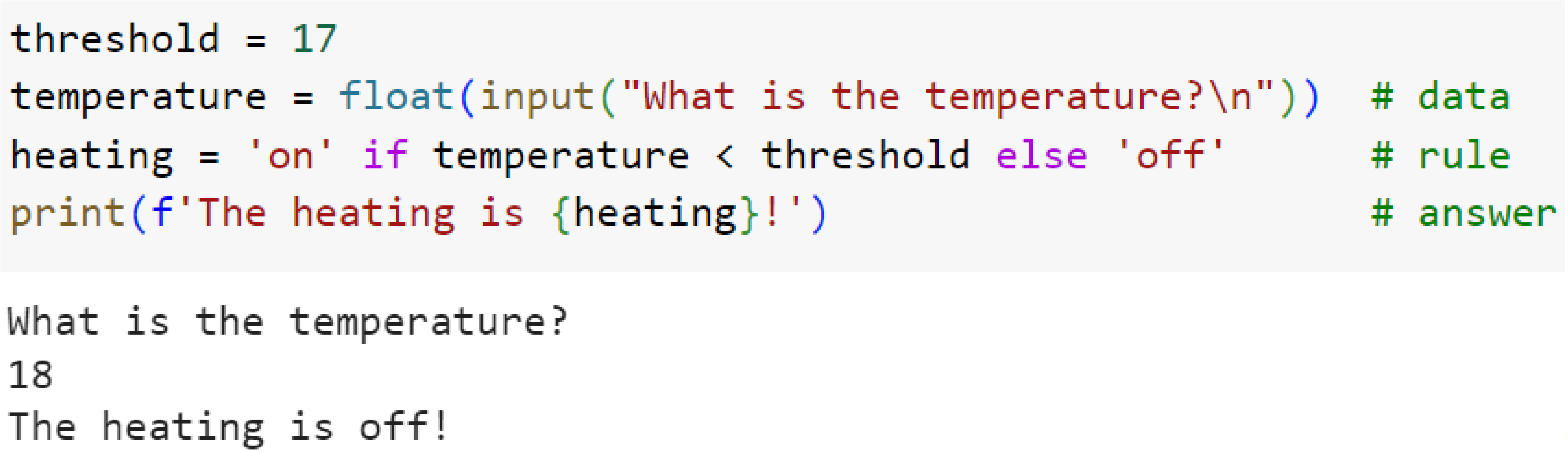
Output

Thermostat example

Traditional Approach

- **The rule is given**:

  “If temperature is smaller than 17°C, then heating is on, otherwise it’s off”
- The algorithm implements the rule
- No data required to derive the rule!

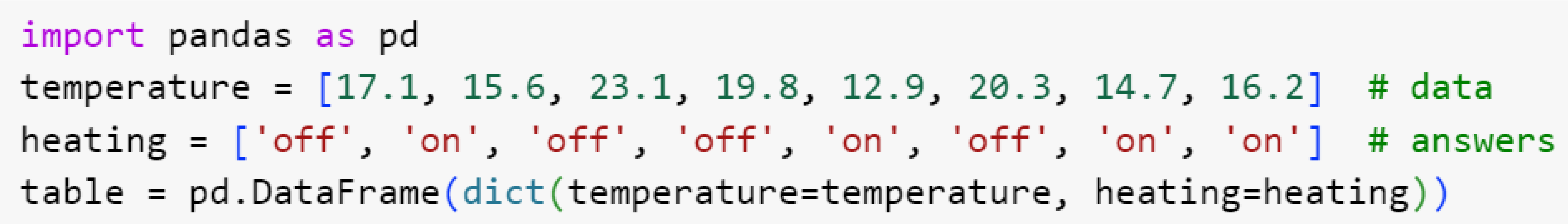


```
threshold = 17
temperature = float(input("What is the temperature?\n"))  # data
heating = 'on' if temperature < threshold else 'off'      # rule
print(f'The heating is {heating}!')                       # answer
```

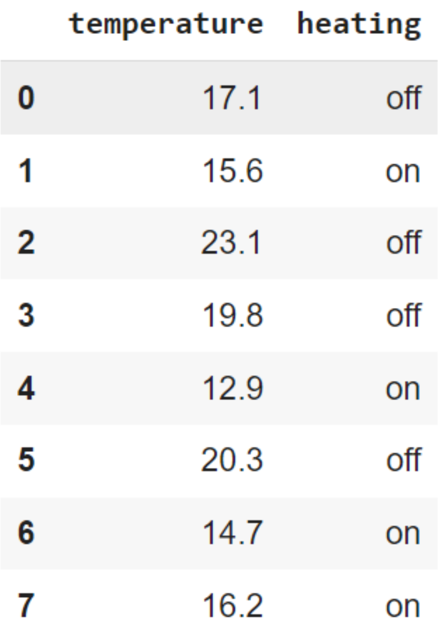
```
What is the temperature?
18
The heating is off!
```

Machine Learning Approach

The rule is not known and must be derived from data!

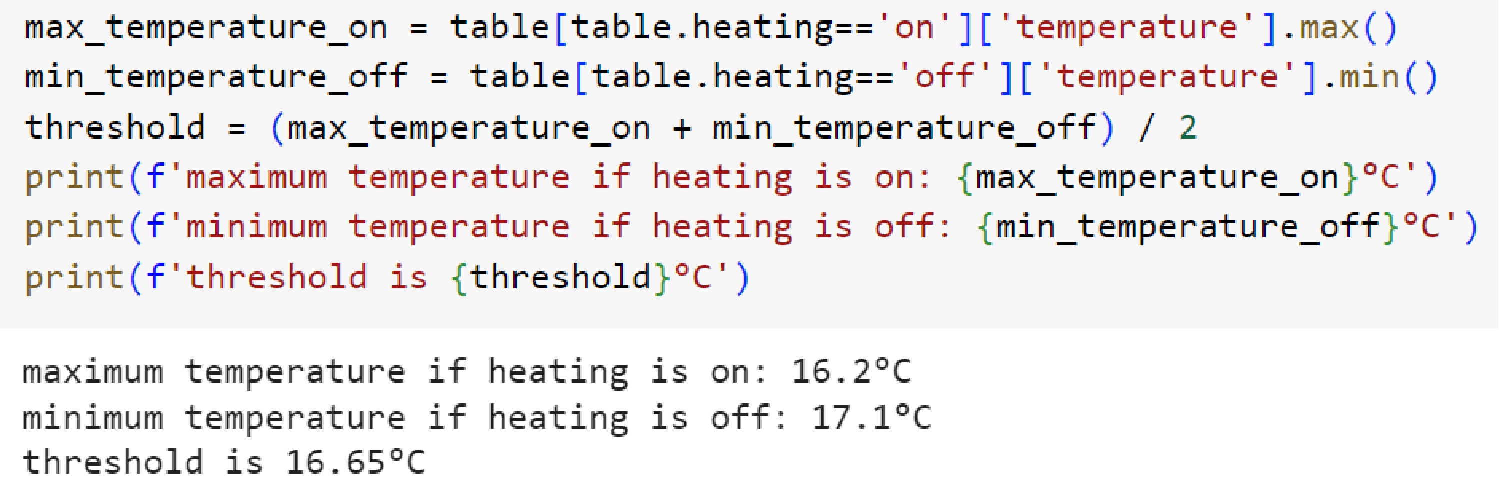


```python
import pandas as pd
temperature = [17.1, 15.6, 23.1, 19.8, 12.9, 20.3, 14.7, 16.2]  # data
heating = ['off', 'on', 'off', 'off', 'on', 'off', 'on', 'on']  # answers
table = pd.DataFrame(dict(temperature=temperature, heating=heating))
```



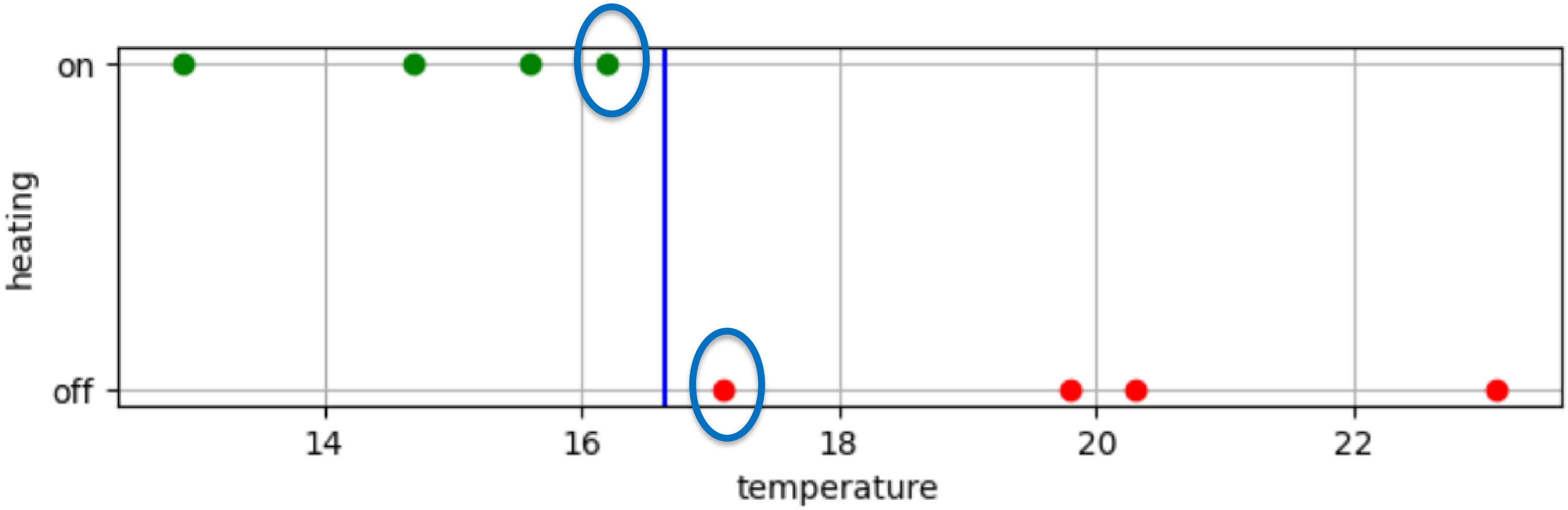
	temperature	heating
0	17.1	off
1	15.6	on
2	23.1	off
3	19.8	off
4	12.9	on
5	20.3	off
6	14.7	on
7	16.2	on

Intuitive Algorithm

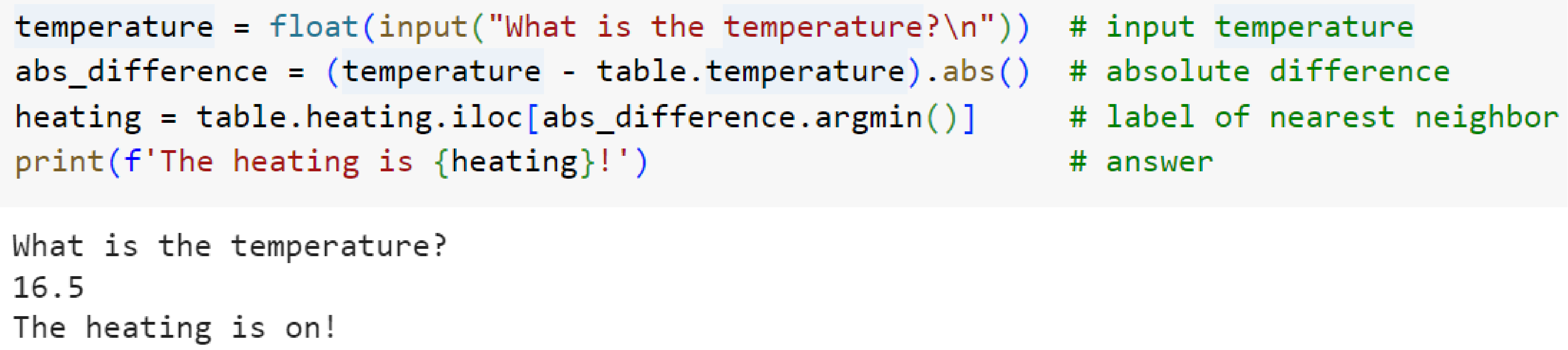


```
max_temperature_on = table[table.heating=='on']['temperature'].max()
min_temperature_off = table[table.heating=='off']['temperature'].min()
threshold = (max_temperature_on + min_temperature_off) / 2
print(f'maximum temperature if heating is on: {max_temperature_on}°C')
print(f'minimum temperature if heating is off: {min_temperature_off}°C')
print(f'threshold is {threshold}°C')
```

```
maximum temperature if heating is on: 16.2°C
minimum temperature if heating is off: 17.1°C
threshold is 16.65°C
```

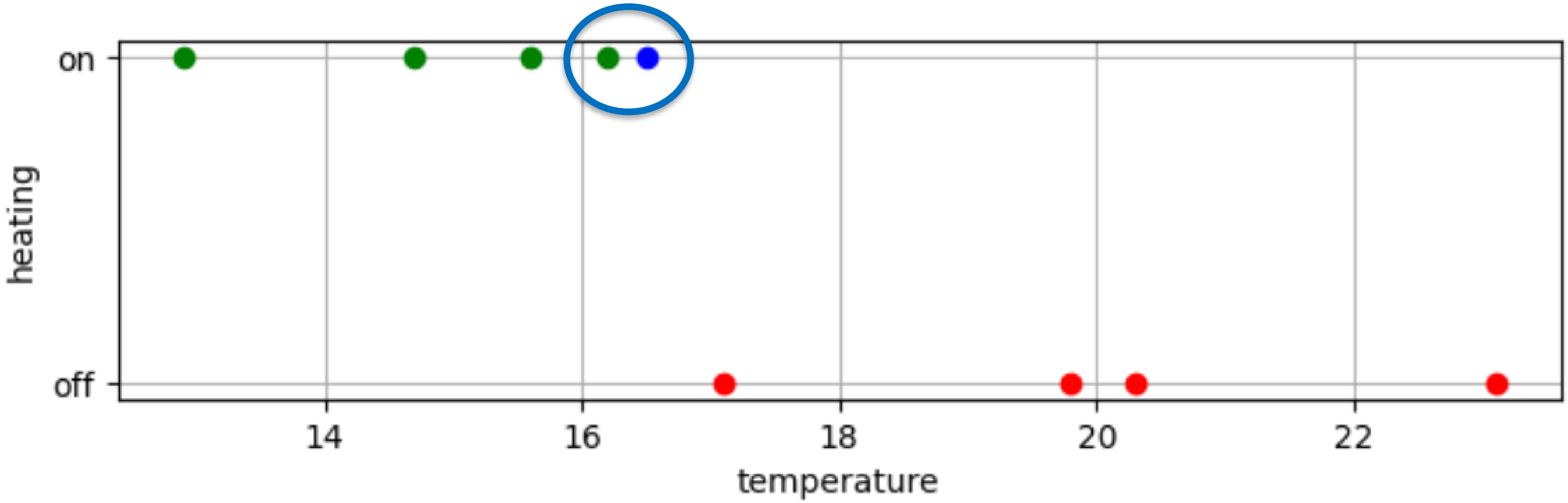


Nearest Neighbor



```
temperature = float(input("What is the temperature?\n"))  # input temperature
abs_difference = (temperature - table.temperature).abs()  # absolute difference
heating = table.heating.iloc[abs_difference.argmin()]     # label of nearest neighbor
print(f'The heating is {heating}!')                       # answer
```

```
What is the temperature?
16.5
The heating is on!
```



Why simple models are not enough

- Real-life datasets are typically much larger:
  - more data points
  - more variables
- Real-life datasets often contain outliers or errors

- **Therefore we need more robust algorithms**
  - that take all samples into account
  - that measure and minimize the errors
- Examples:
  - **Decision Tree**: splits the data repeatedly into smaller, more homogeneous groups
  - **Logistic Regression**: finds the best boundary that separates the data
  - **K Nearest Neighbors**: considers K nearest data points instead of 1

Guest Lectures at TTK University of Applied Sciences, Tallinn, Estonia

Introduction to Machine Learning

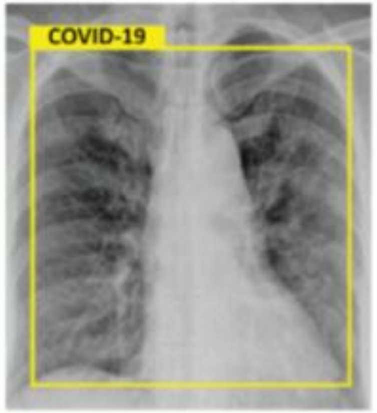
APPLICATIONS & TASKS

TALLINNA TEHNIKAKÕRGKOOL
TTK UNIVERSITY OF APPLIED SCIENCES

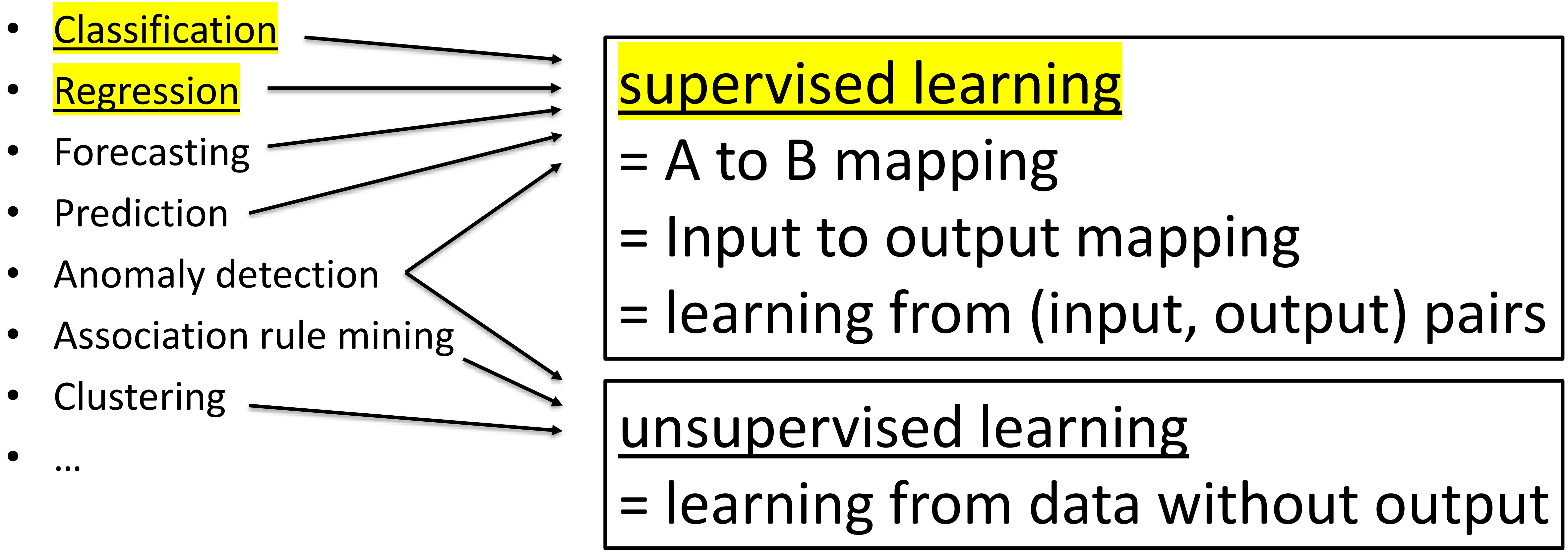
vives University of Applied Sciences

ML Applications

- Spam filters
- Recommender systems
- Personalized shopping
- Voice assistants
- Self-driving cars
- Search engines
- Chatbots
- Fraud prevention
- Face recognition
- Medical imaging
- Robotics
- Route planning
- Sales forecasting
- Deepfakes
- …



Machine Learning Tasks

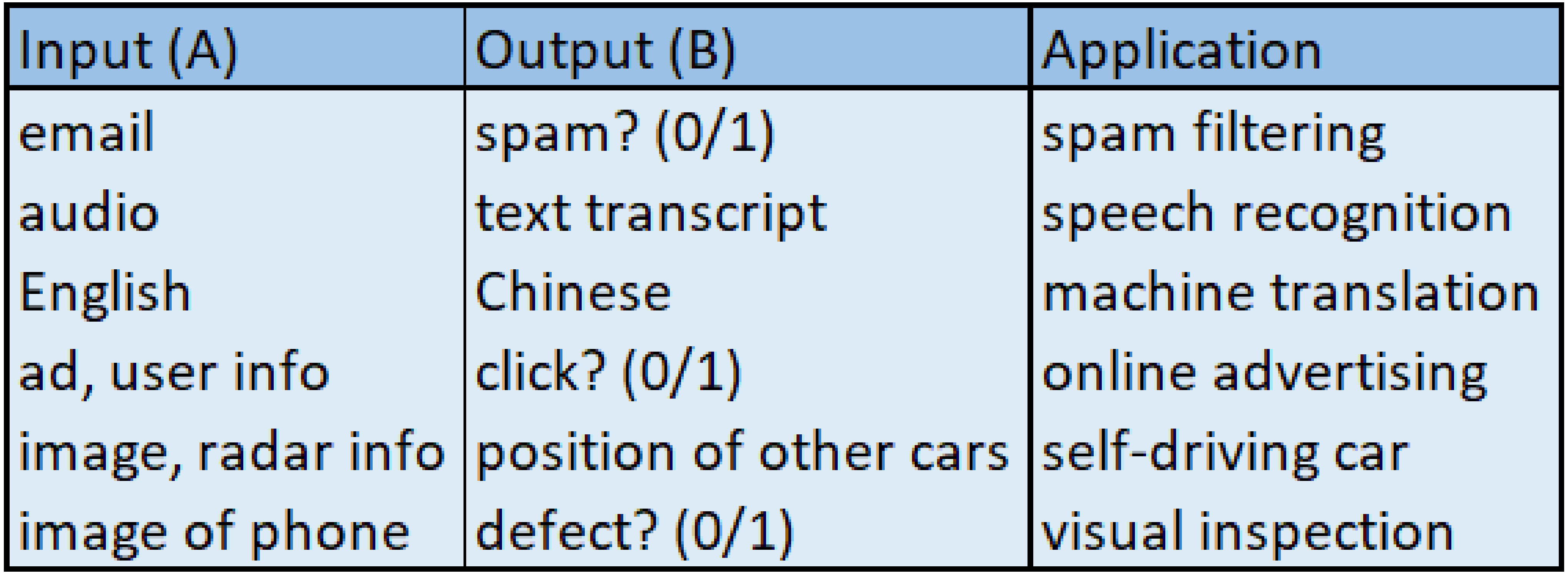


- Classification
- Regression
- Forecasting
- Prediction
- Anomaly detection
- Association rule mining
- Clustering
- …

supervised learning
= A to B mapping
= Input to output mapping
= learning from (input, output) pairs

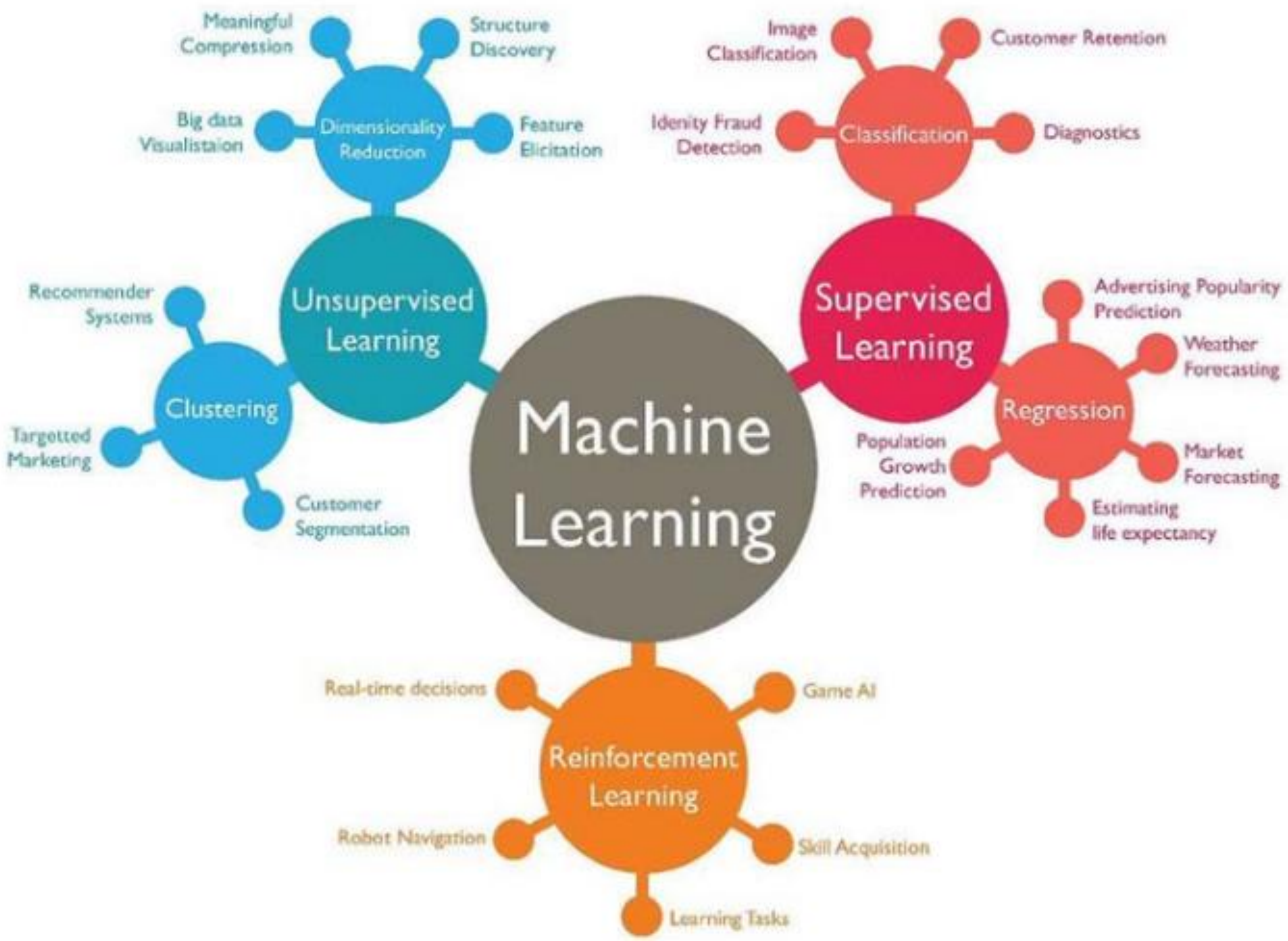
unsupervised learning
= learning from data without output

Supervised Learning



Input (A)	Output (B)	Application
email	spam? (0/1)	spam filtering
audio	text transcript	speech recognition
English	Chinese	machine translation
ad, user info	click? (0/1)	online advertising
image, radar info	position of other cars	self-driving car
image of phone	defect? (0/1)	visual inspection

The Big Three



Category of machine learning. Image by https://www.techleer.com/articles/203-machine-learning-algorithm-backbone-of-emerging-technologies/

What about GenAI?

- **Supervised learning**

  = learning from labeled data

- **Unsupervised learning**

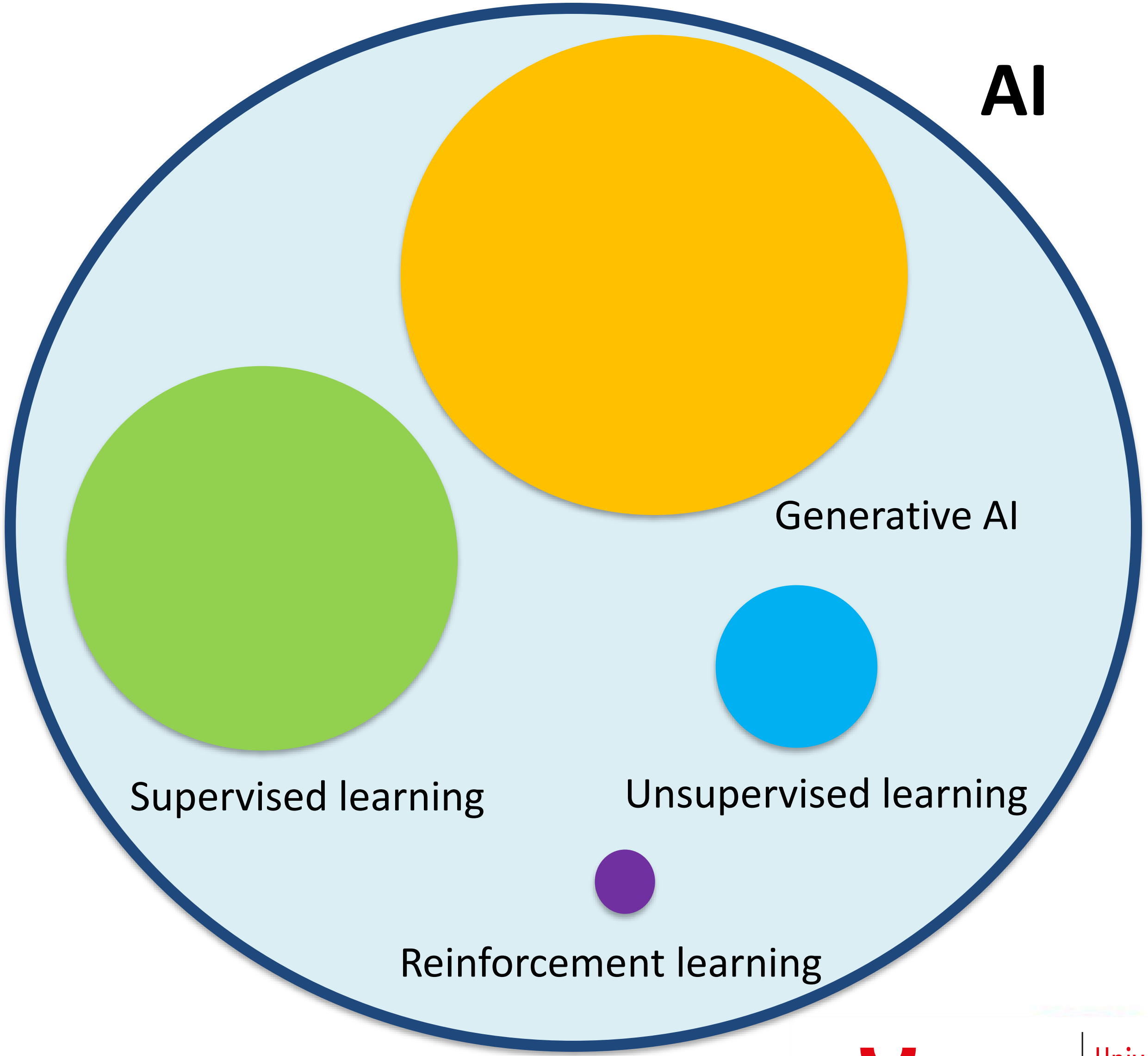
  = learning from unlabeled data

- **Reinforcement learning**

  = learning from rewards

- **Generative AI**

  = generating new data



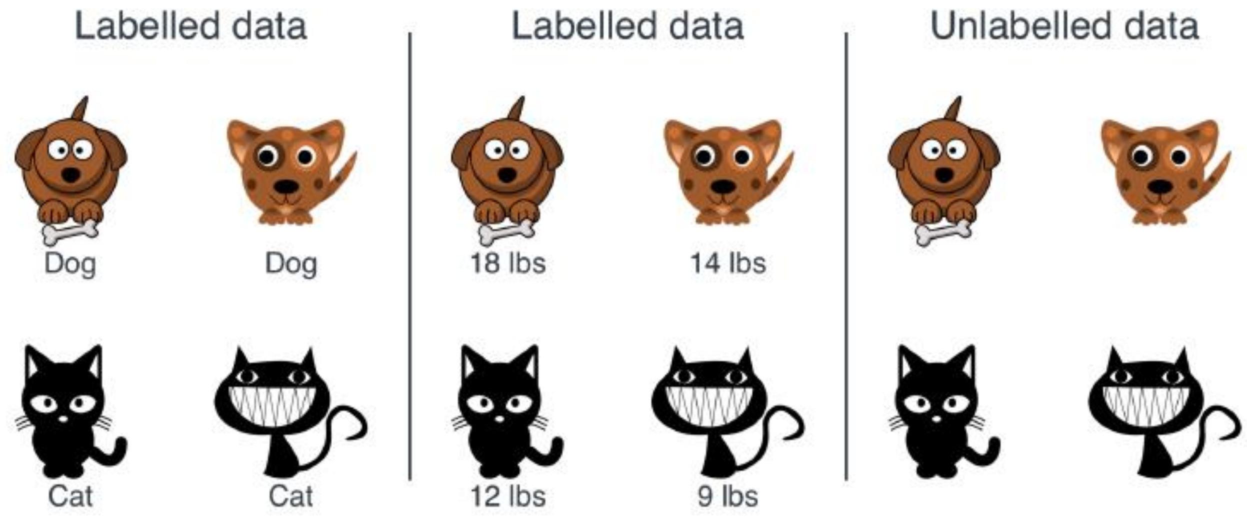
Supervised Learning versus Unsupervised Learning

- **Labeled data**: data with label

  → **SUPERVISED** LEARNING

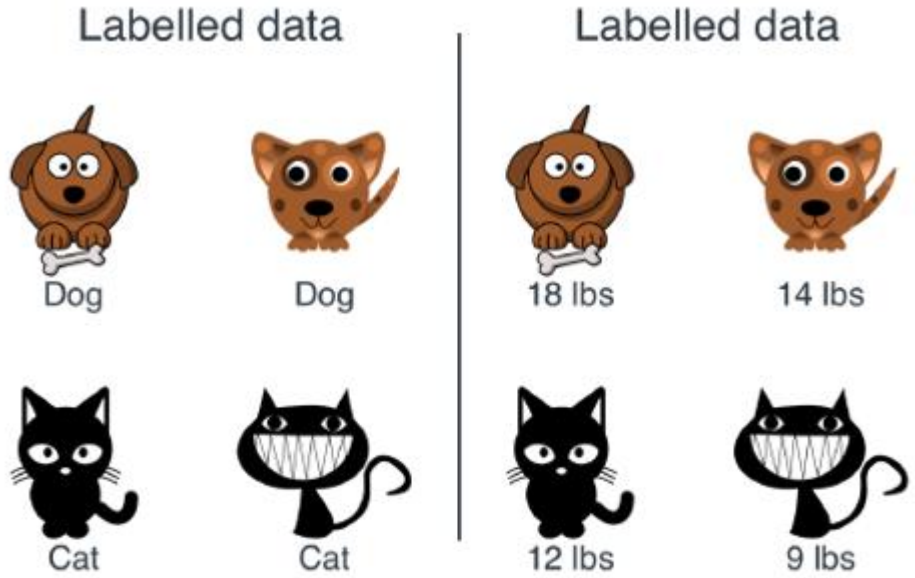
- **Unlabeled data**: data without label

  → **UNSUPERVISED** LEARNING

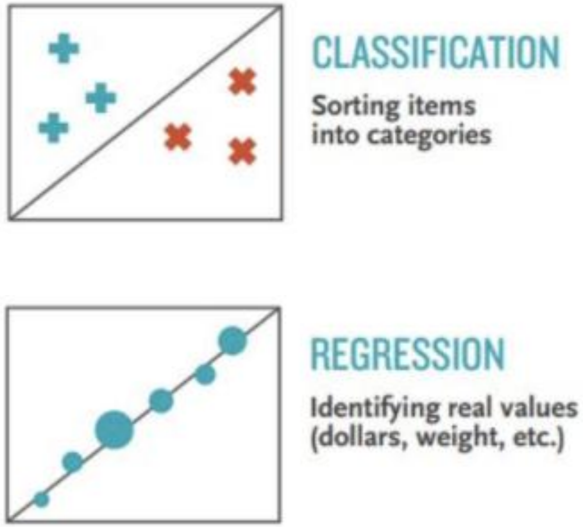


Classification vs Regression

- **Qualitative or Categorical** target ➔ **Classification**
- **Quantitative or Numerical** target ➔ **Regression**

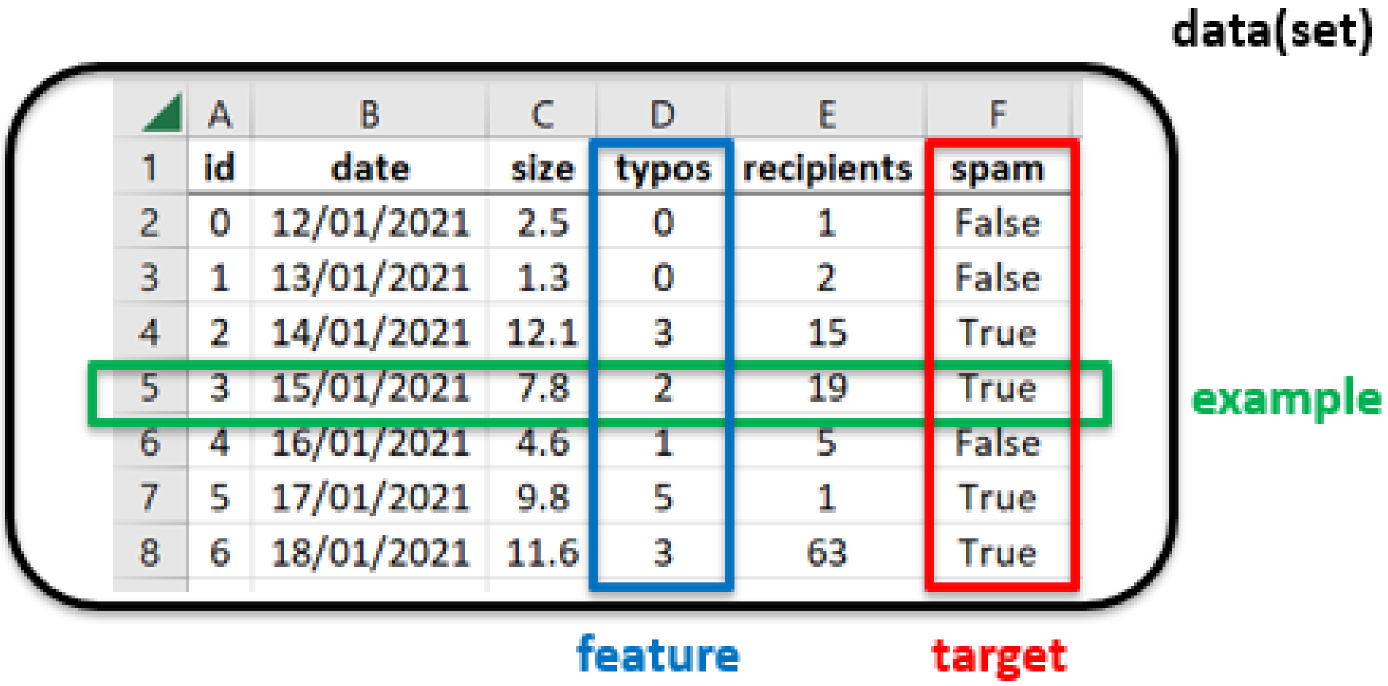


Categorical **Numerical**

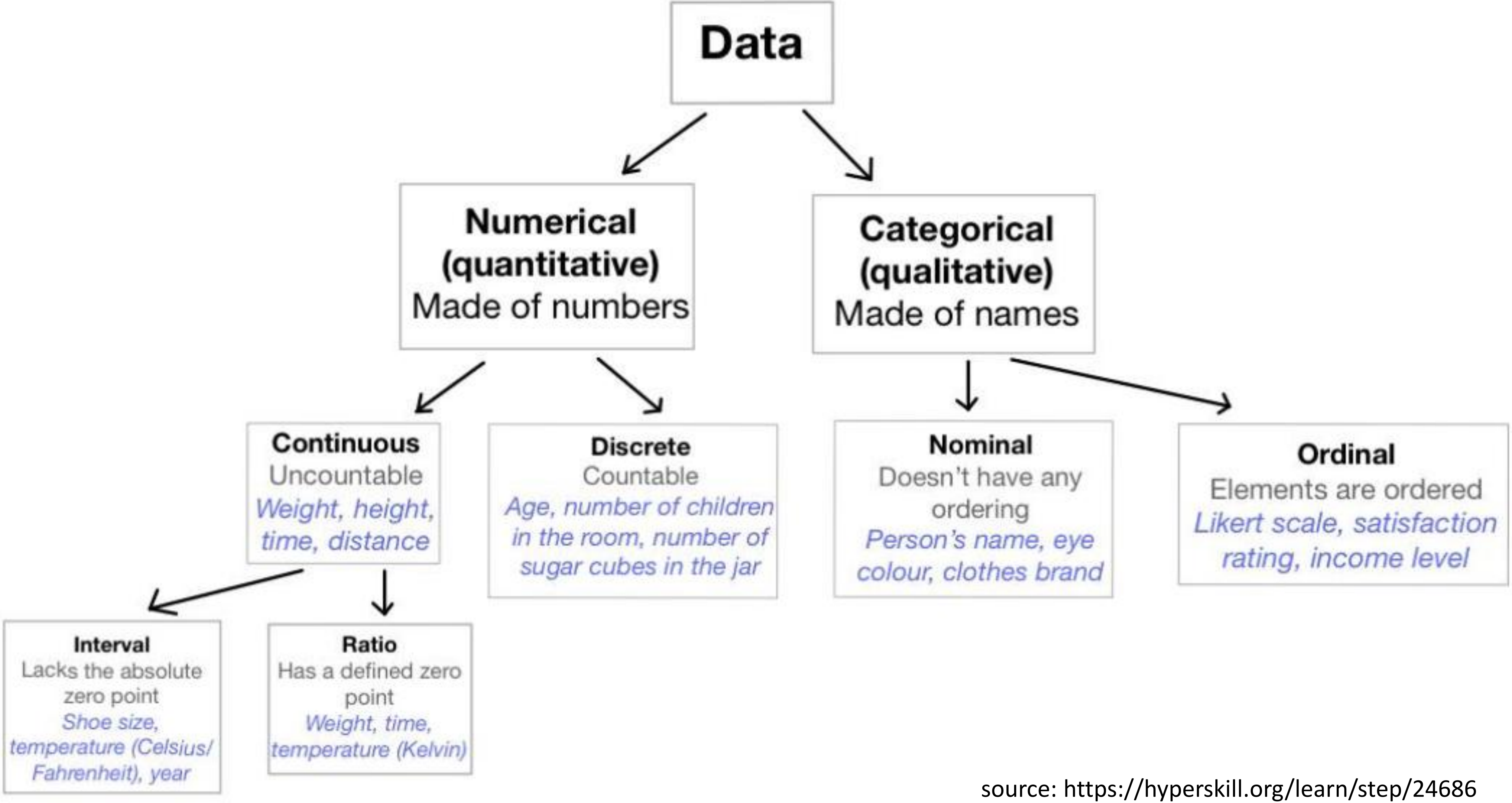

CLASSIFICATION
Sorting items into categories
REGRESSION
Identifying real values (dollars, weight, etc.)

Structured Data

- **Data** = dataset = information (= table)
- **Example** = sample = instance = data point = observation (= table row/record)
- **Feature** = predictor = independent variable = input variable (= table column/attribute)
- **Target** = labels = dependent variable = response variable = output variable



	A	B	C	D	E	F
1	id	date	size	typos	recipients	spam
2	0	12/01/2021	2.5	0	1	False
3	1	13/01/2021	1.3	0	2	False
4	2	14/01/2021	12.1	3	15	True
5	3	15/01/2021	7.8	2	19	True
6	4	16/01/2021	4.6	1	5	False
7	5	17/01/2021	9.8	5	1	True
8	6	18/01/2021	11.6	3	63	True



Data

Numerical (quantitative)
Made of numbers

Continuous
Uncountable
Weight, height, time, distance

Interval
Lacks the absolute zero point
Shoe size, temperature (Celsius/ Fahrenheit), year

Ratio
Has a defined zero point
Weight, time, temperature (Kelvin)

Discrete
Countable
Age, number of children in the room, number of sugar cubes in the jar

Categorical (qualitative)
Made of names

Nominal
Doesn't have any ordering
Person's name, eye colour, clothes brand

Ordinal
Elements are ordered
Likert scale, satisfaction rating, income level

source: https://hyperskill.org/learn/step/24686

Guest Lectures at TTK University of Applied Sciences, Tallinn, Estonia

Introduction to Machine Learning

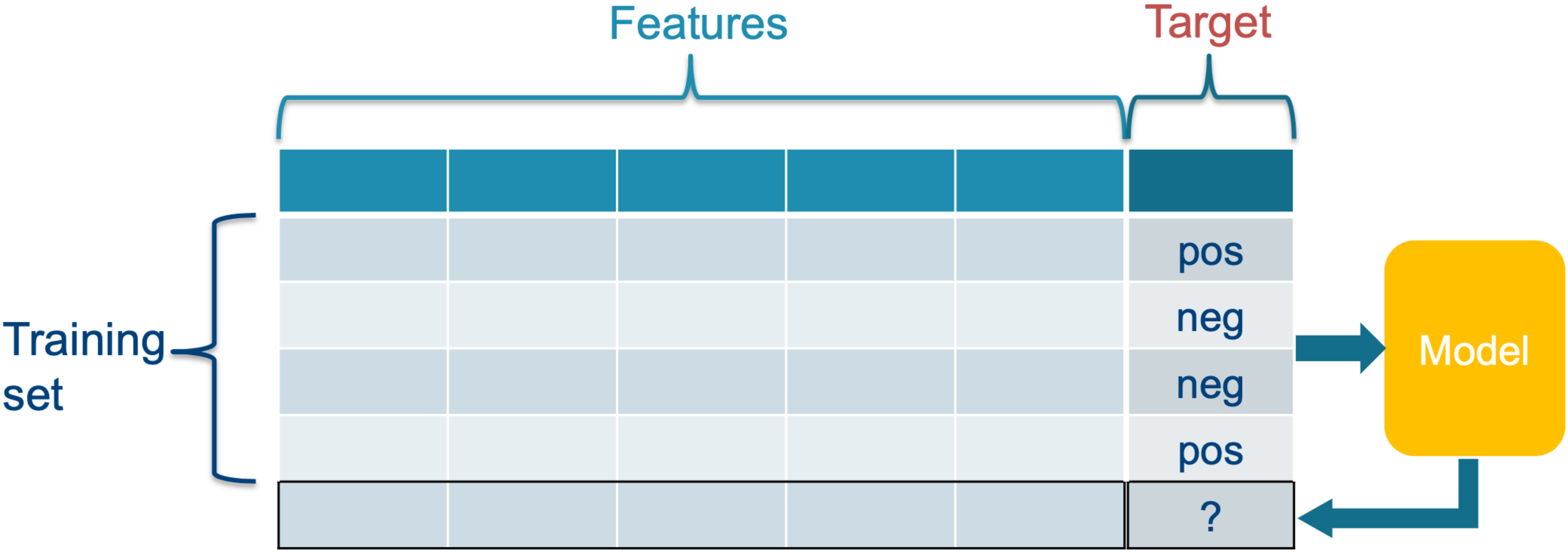
SUPERVISED LEARNING

TALLINNA TEHNIKAKÕRGKOOL
TTK UNIVERSITY OF APPLIED SCIENCES

vives University of Applied Sciences

Supervised Learning Task

- Task: learn a model to predict a target for new data instances,
  based on a training set of data instances for which the target is known



Features

Target

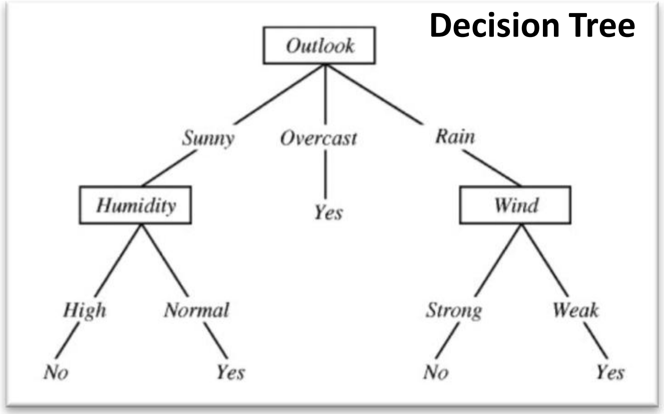
					pos
					neg
					neg
					pos
					?

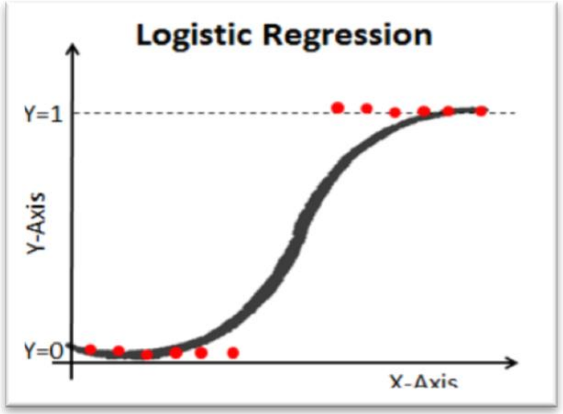
Training set

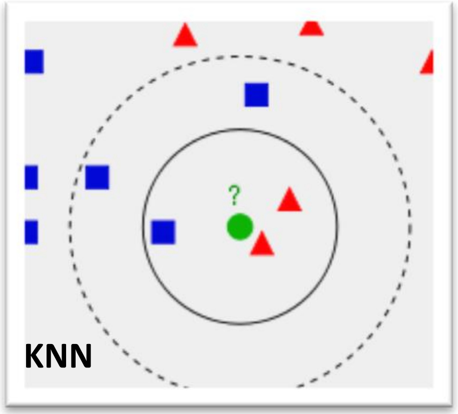
Model

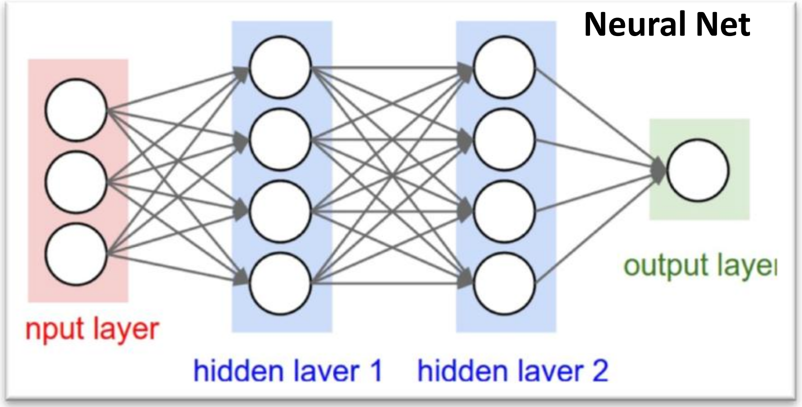
Supervised Learning Algorithms

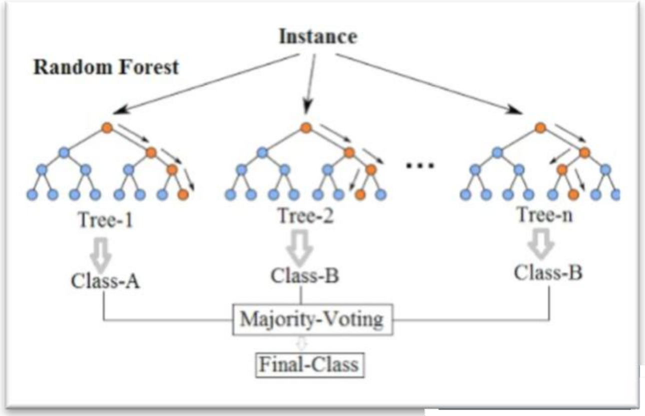
- There exist plenty of supervised learning algorithms
- **No free lunch**: there is no algorithm that works best for every problem







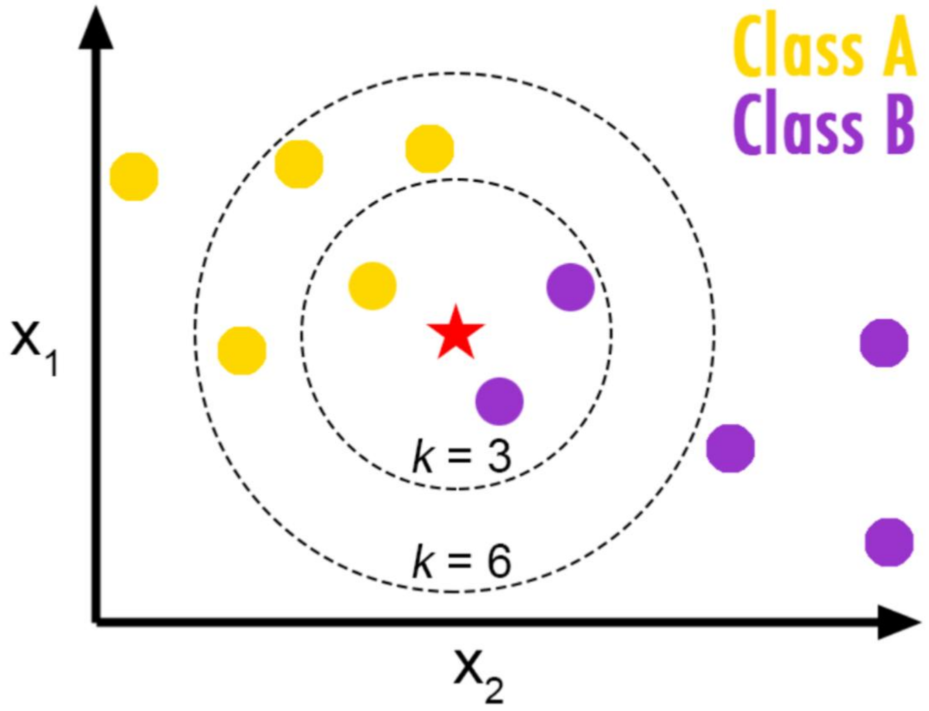




...

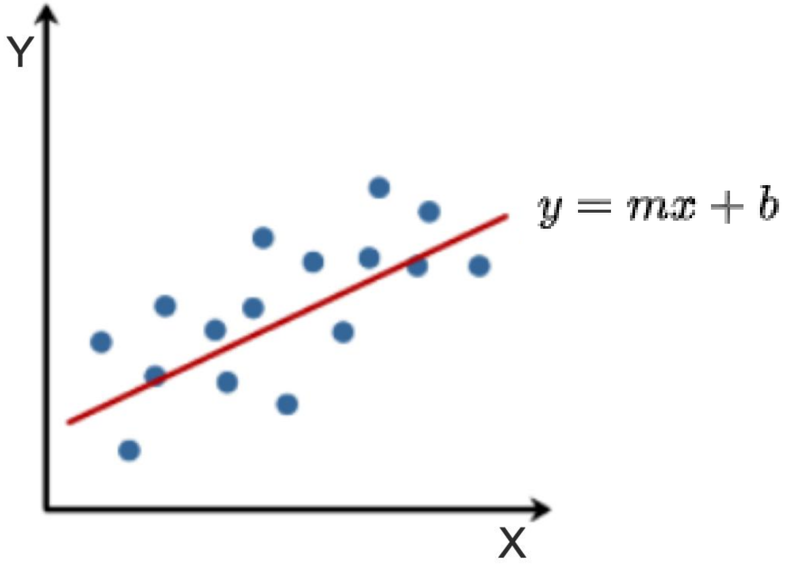
K Nearest Neighbors (KNN)

- Classification (regression is also possible)
- Requires no training (= lazy learning, as opposed to eager learning)
- Main task: find suitable distance function (Euclidean, Manhattan, …)



Simple Linear Regression

- Regression for numeric targets
- 1 independent variable (feature *x*) and 1 dependent variable (target *y*)
- Main task: estimate parameters *m* and *b*, such that predictions (red line) and targets (blue dots) are as close as possible (= best-fitting straight line)



Simple Linear Regression: Algorithm versus Model

Example: predicting weight from height

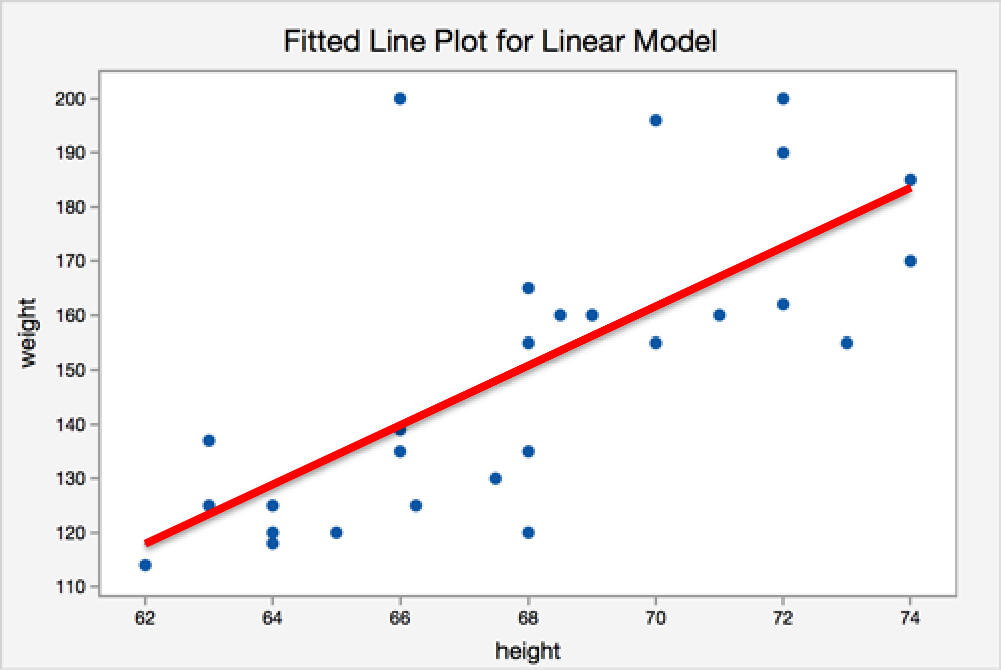
- **Input**: (*x*, *y*) data with *x* = height (inches) and *y* = weight (pounds)
- **Output**: predicted weight
- **Algorithm**: *calculates* the best fitting line

  find parameters m and b in

  $y = mx + b$

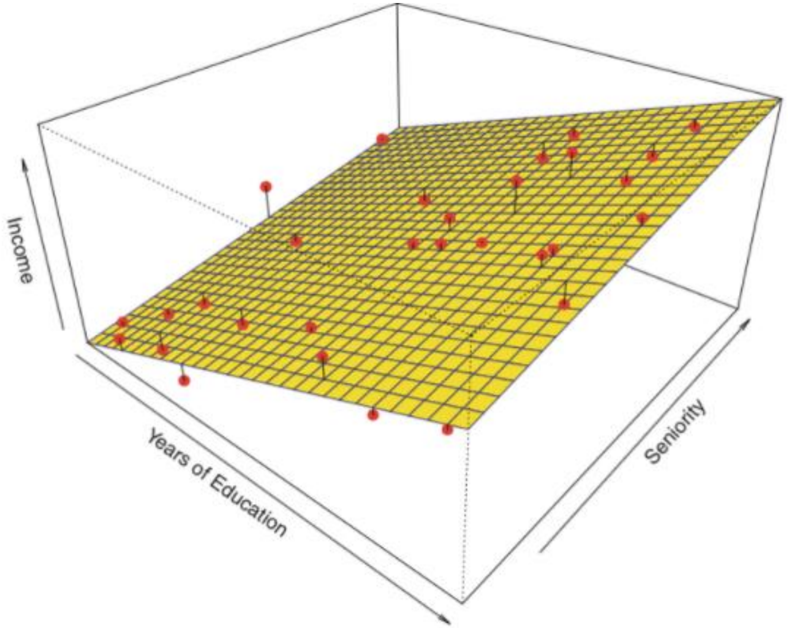
- **Model**: the best fitting line

  $\text{weight} = -222.5 + 5.49 \text{ height}$

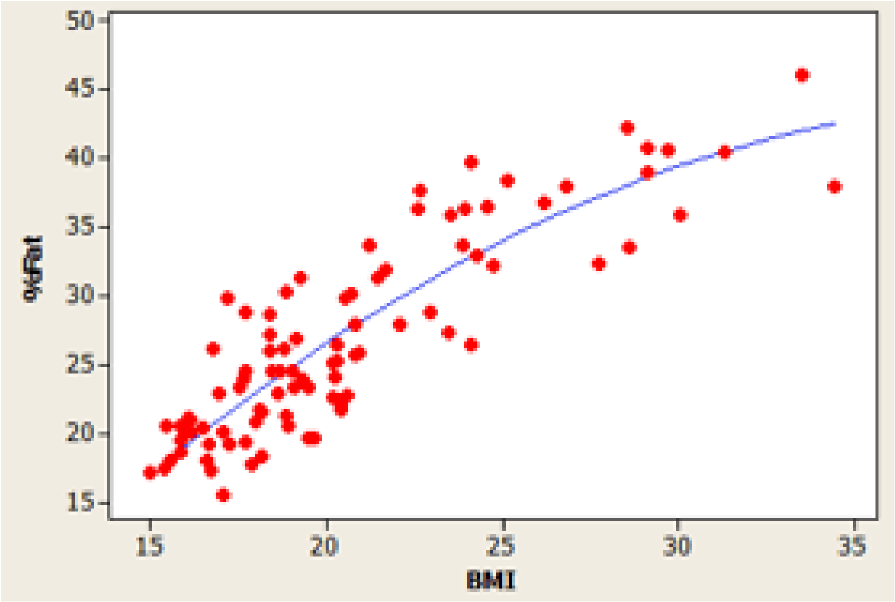


Linear & Nonlinear Regression

- Linear Regression: 2 features and 1 target (left)
- Nonlinear regression: 1 feature and 1 target (right)



Source: James et al. *Introduction to Statistical Learning* (Springer 2013)

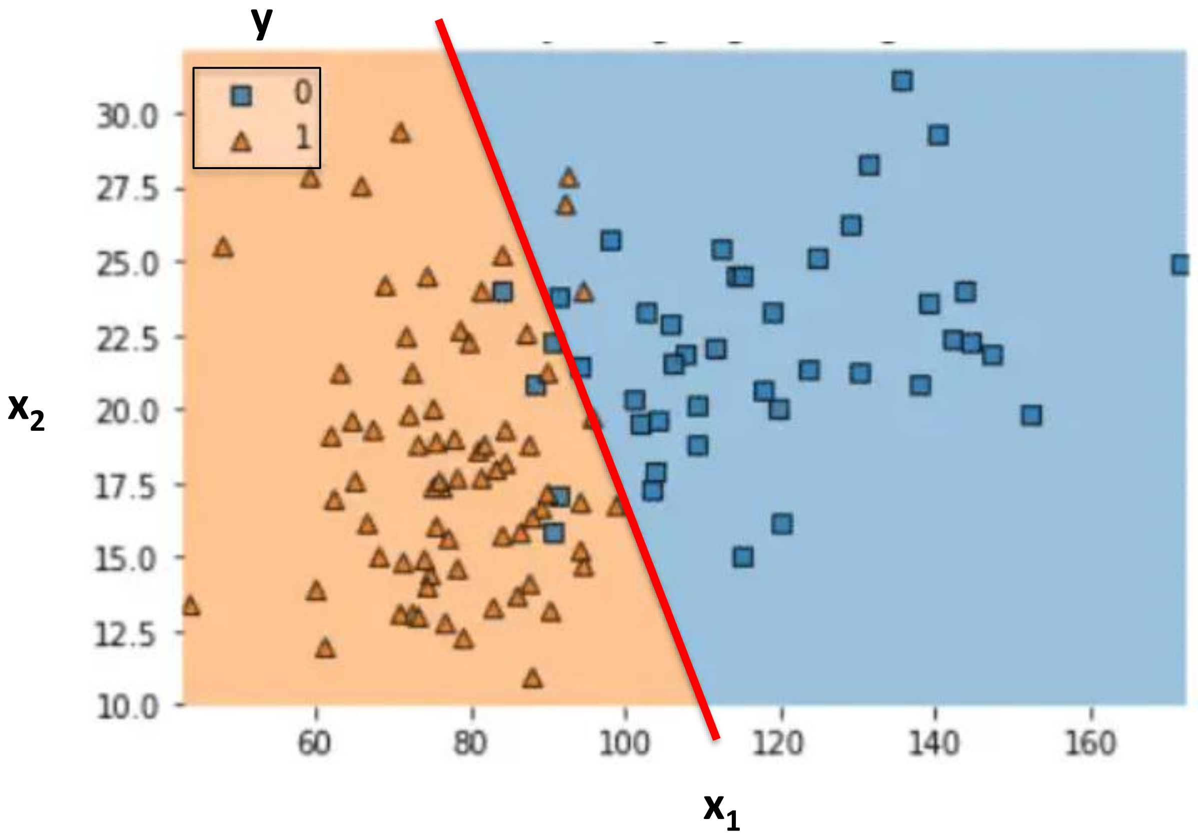


Source: the minitab blog

Logistic Regression

- Regression for binary targets
- Features x_i and target y
- Main task: find a separating straight line

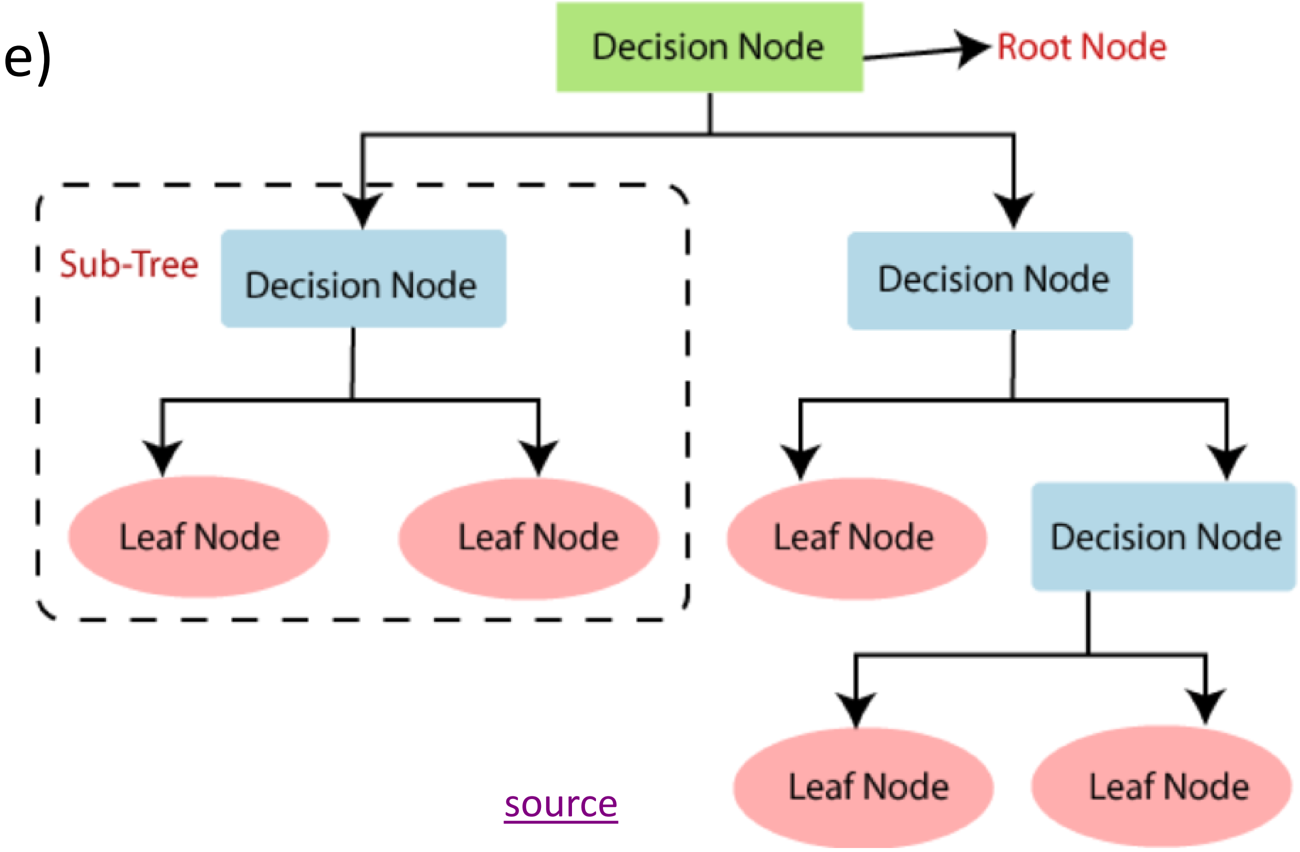
  = binary classification
- N dimensions: separating hyperplane



source: https://www.jcchouinard.com/logistic-regression/

Decision Trees

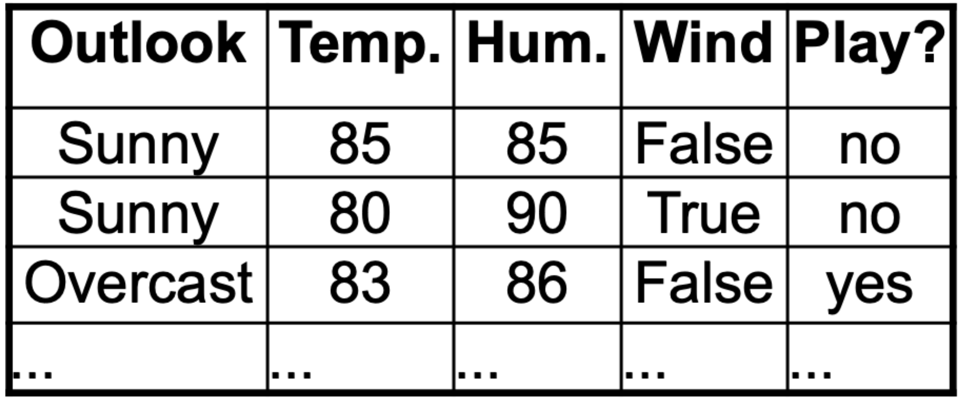
- Very popular data mining methods
- High interpretability/explainability (e.g.medicine)
- Efficient learning and prediction procedures
- Classification (regression is also possible)



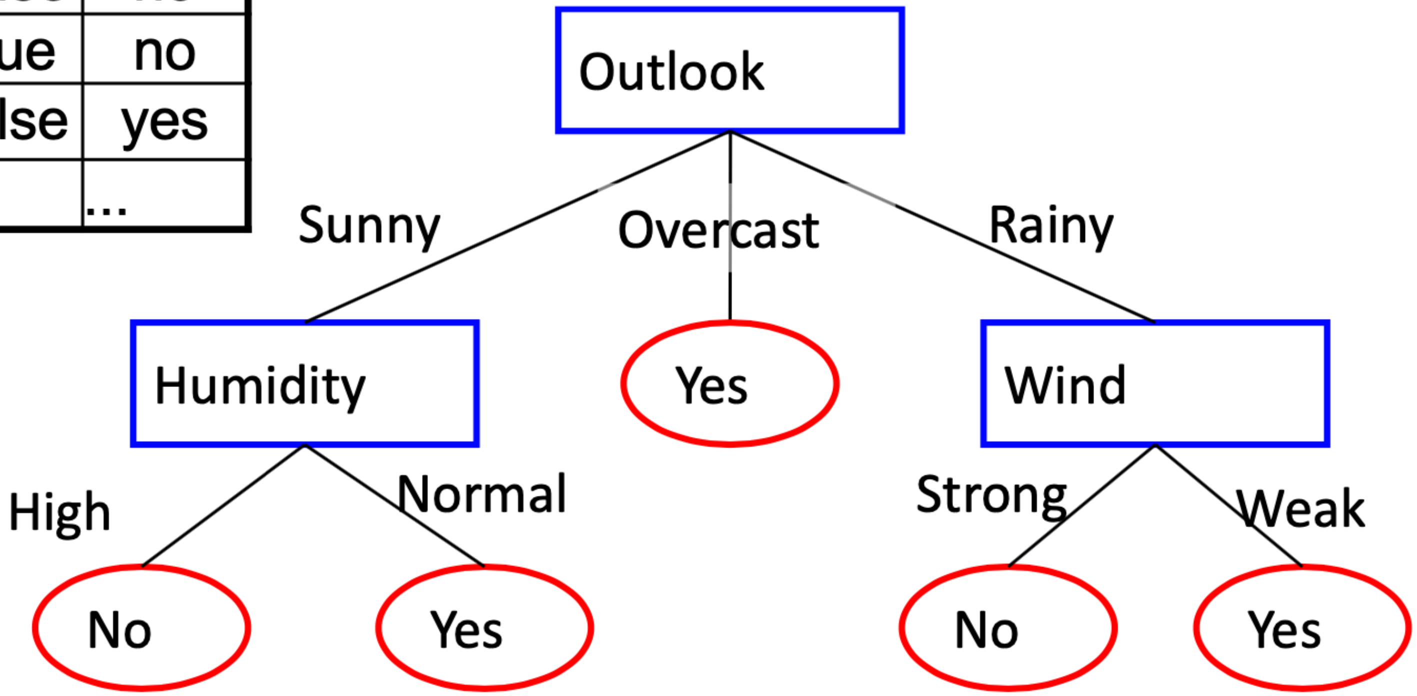
source

Decision Tree

- Example: Play tennis or not? (depending on weather conditions)



Outlook	Temp.	Hum.	Wind	Play?
Sunny	85	85	False	no
Sunny	80	90	True	no
Overcast	83	86	False	yes
…	…	…	…	…



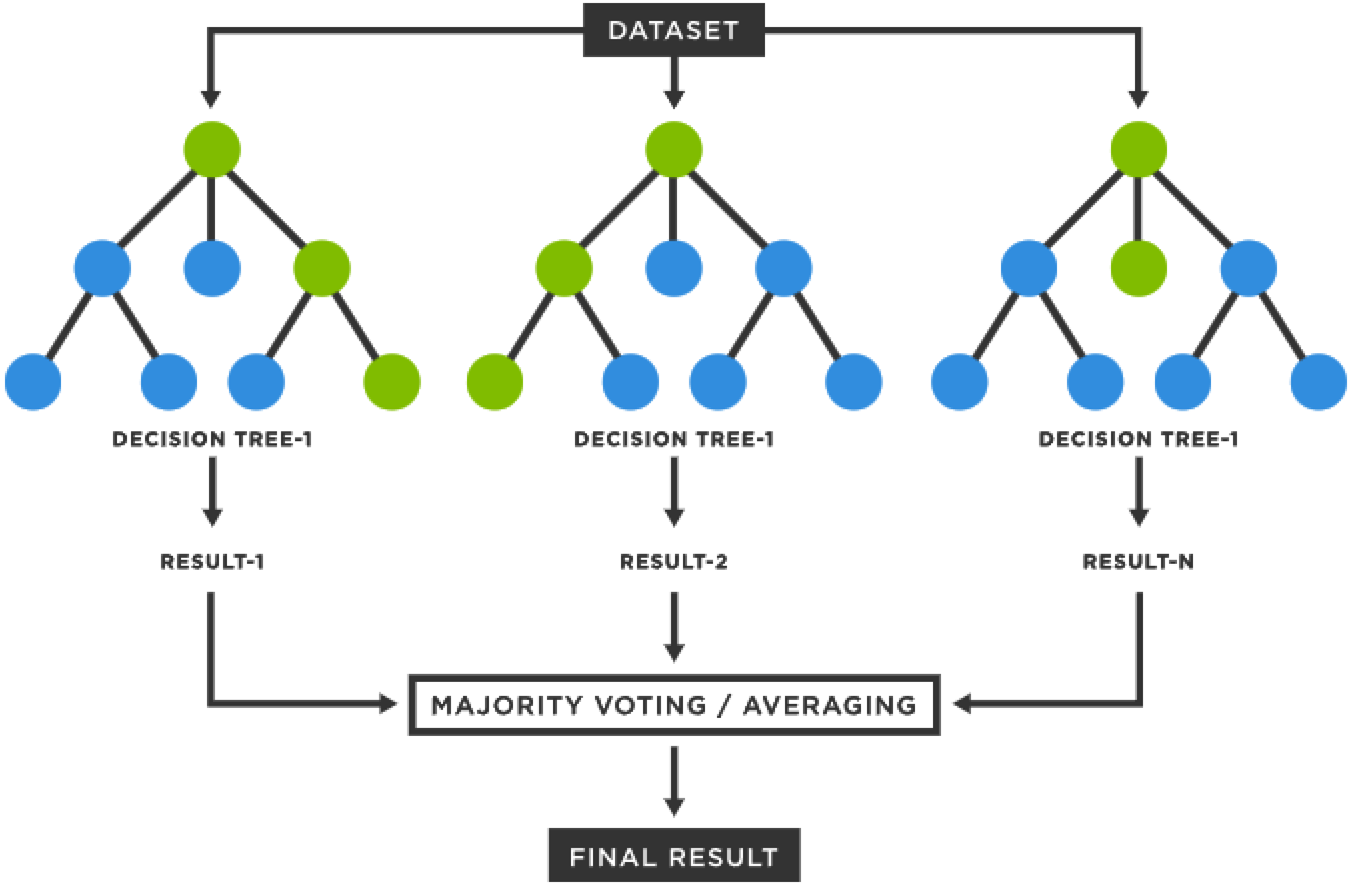
- Leaf nodes versus internal nodes

  = labels = features

Random Forests

= Ensemble of decision trees

- Each tree is trained on a random subset (= **bagging**)
- Voting mechanism:
  - Classification: majority voting
  - Regression: averaging



https://www.spotfire.com/glossary/what-is-a-random-forest

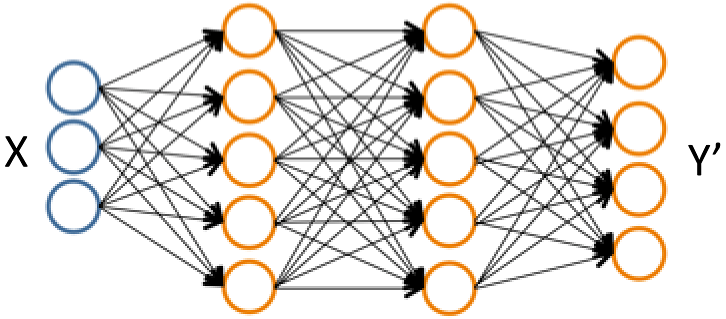
Artificial Neural Network

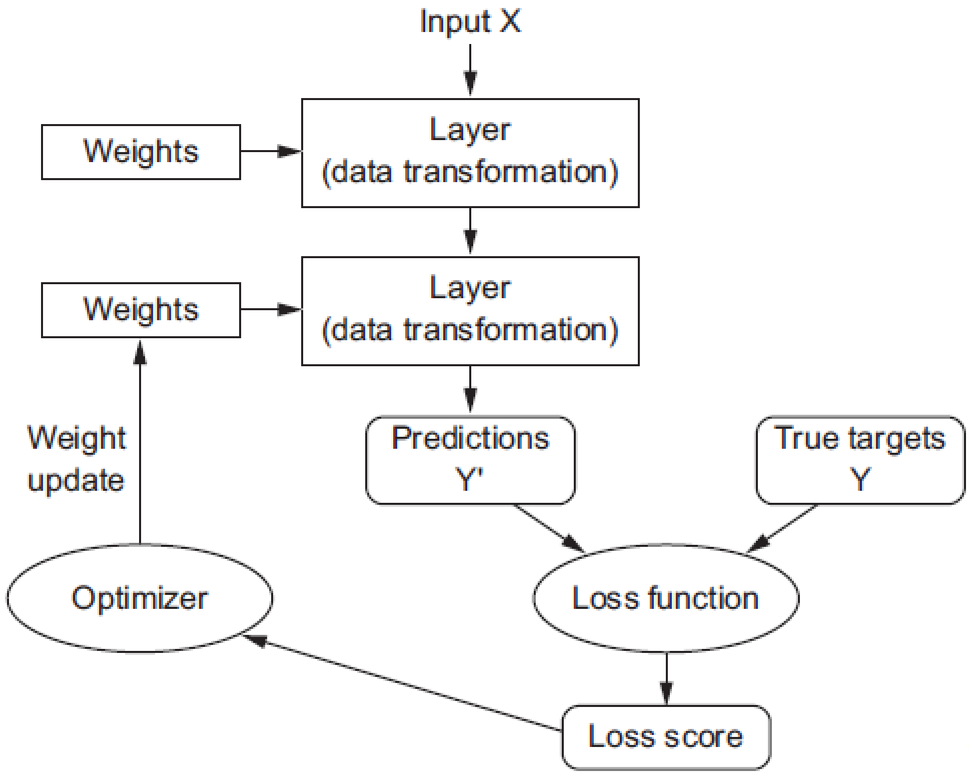
- Regression or classification

- Features X and targets Y

- **Loss**: function quantifying differences
  between targets Y and predictions Y’

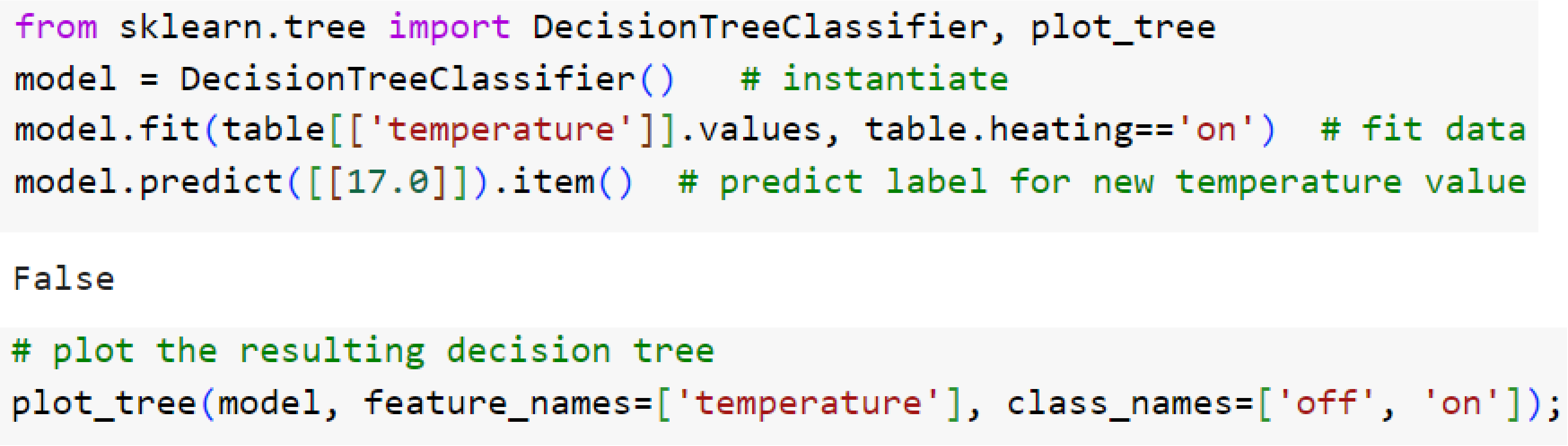
- Main task: find optimal weights
  that minimize the loss





Thermostat example

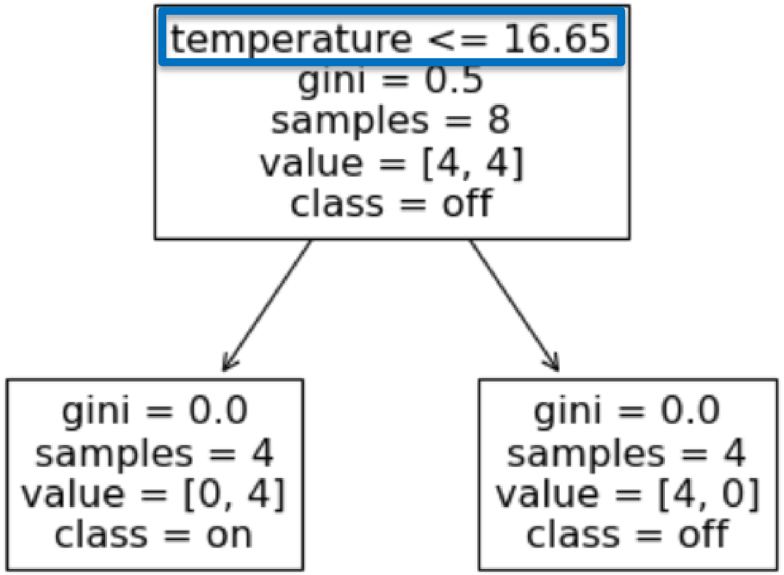
Decision Tree



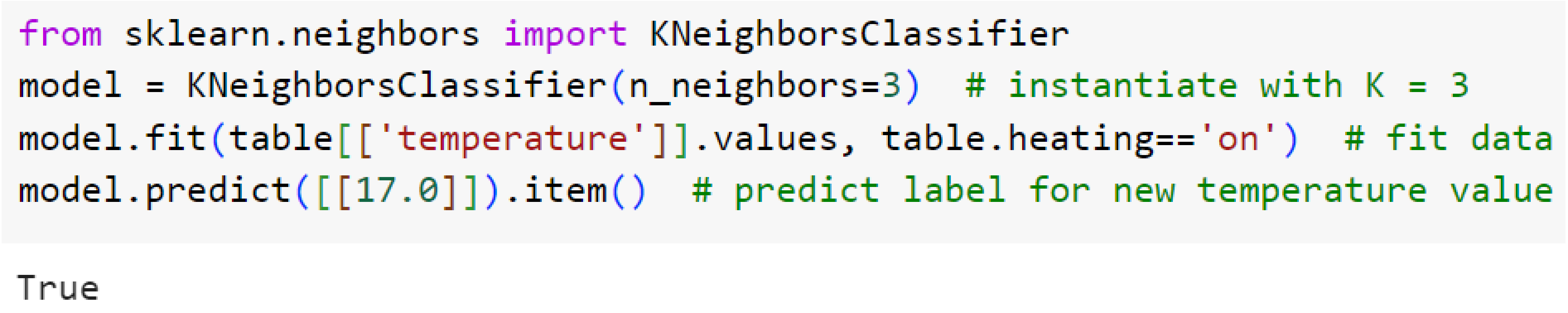
```python
from sklearn.tree import DecisionTreeClassifier, plot_tree
model = DecisionTreeClassifier()   # instantiate
model.fit(table[['temperature']].values, table.heating=='on')  # fit data
model.predict([[17.0]]).item()  # predict label for new temperature value
```

```
False
```

```python
# plot the resulting decision tree
plot_tree(model, feature_names=['temperature'], class_names=['off', 'on']);
```

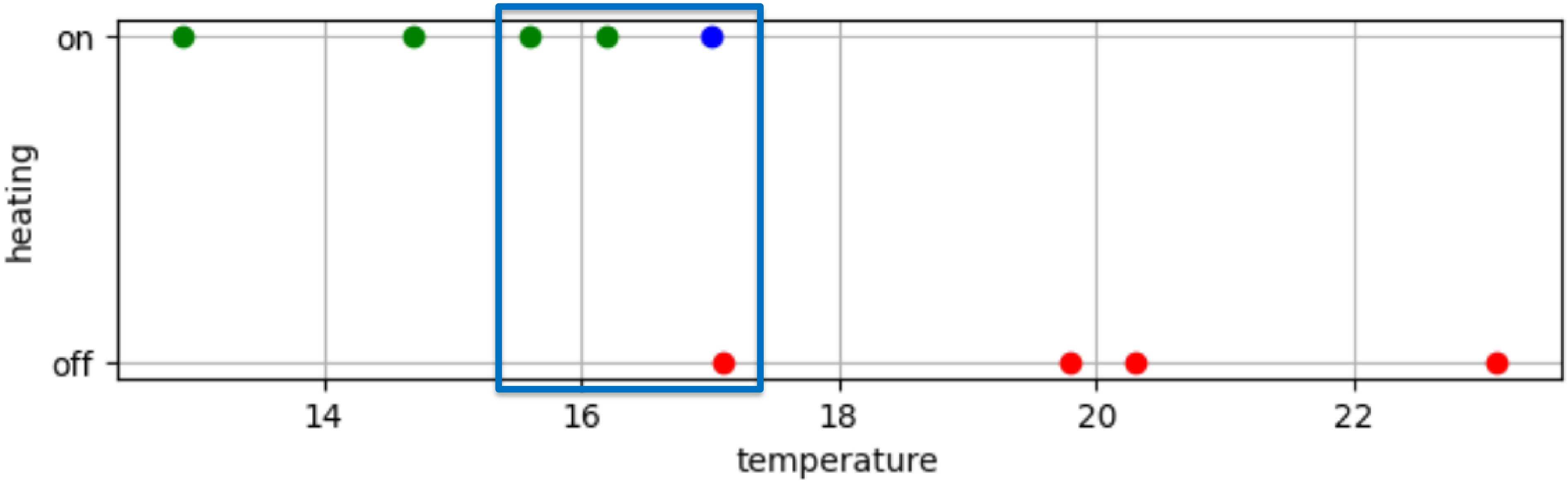


K Nearest Neighbors

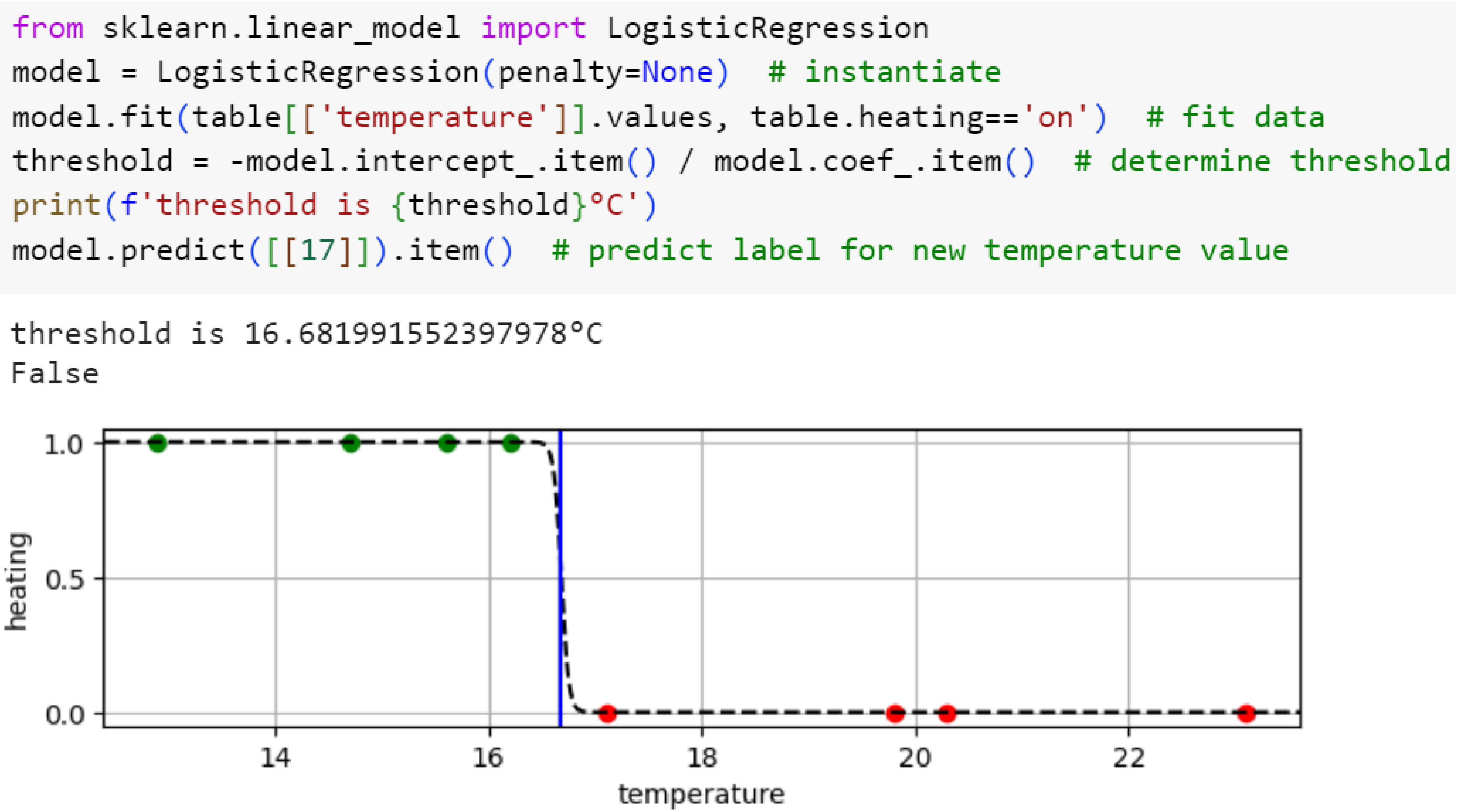


```
from sklearn.neighbors import KNeighborsClassifier
model = KNeighborsClassifier(n_neighbors=3)  # instantiate with K = 3
model.fit(table[['temperature']].values, table.heating=='on')  # fit data
model.predict([[17.0]]).item()  # predict label for new temperature value
```

```
True
```



Logistic Regression



```python
from sklearn.linear_model import LogisticRegression
model = LogisticRegression(penalty=None)  # instantiate
model.fit(table[['temperature']].values, table.heating=='on')  # fit data
threshold = -model.intercept_.item() / model.coef_.item()  # determine threshold
print(f'threshold is {threshold}°C')
model.predict([[17]]).item()  # predict label for new temperature value
```

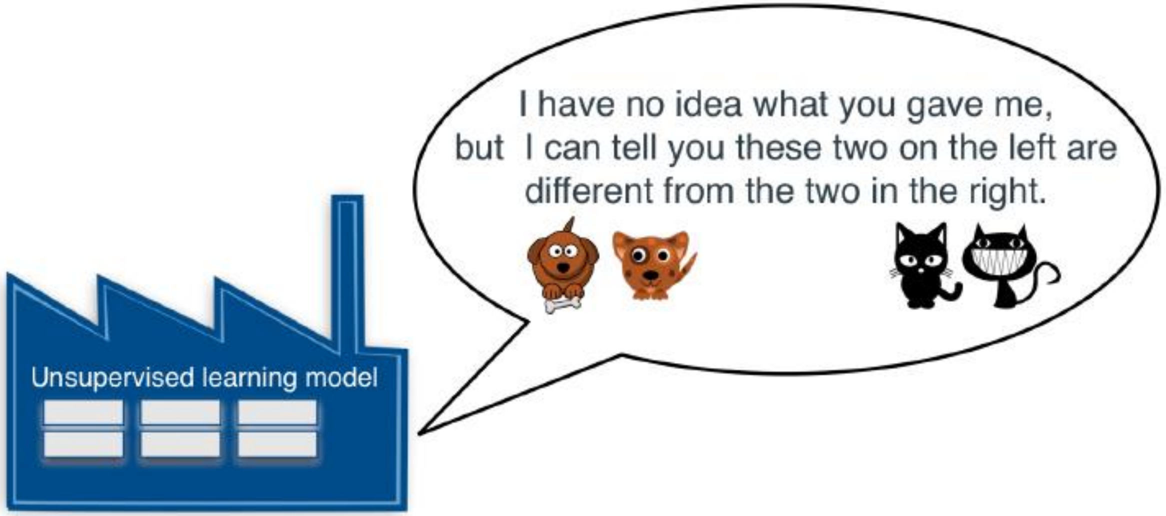
```
threshold is 16.681991552397978°C
False
```

Guest Lectures at TTK University of Applied Sciences, Tallinn, Estonia

Introduction to Machine Learning

UNSUPERVISED LEARNING

Unsupervised Learning


I have no idea what you gave me,
but I can tell you these two on the left are different from the two in the right.
Unsupervised learning model

- Data are **not labeled**
- Often used during data **preprocessing**

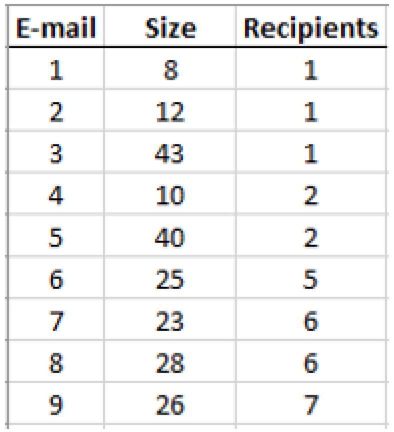
- Clustering: grouping data based on similarities
- Dimensionality reduction: reducing the number of features while retaining as much meaningful information as possible
- Matrix factorization: decomposing the data in order to discover latent features

Clustering

Applications:

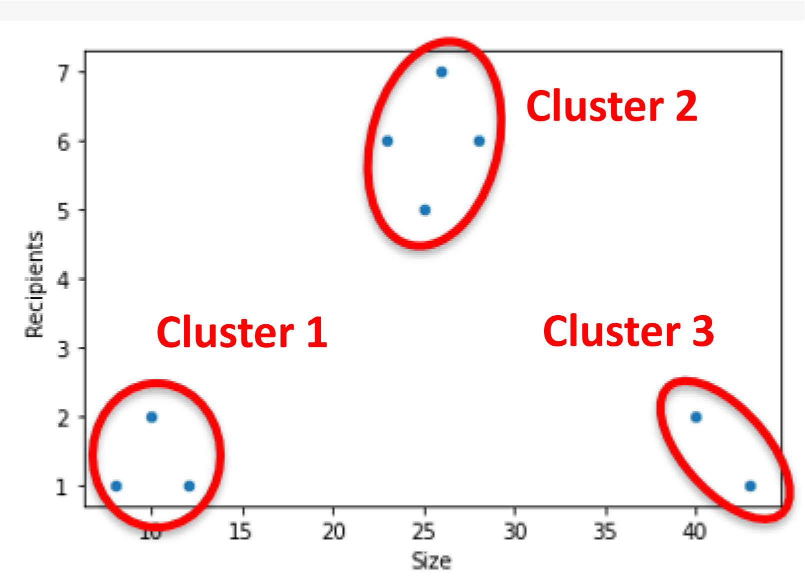
- Genetics: grouping species based on similarities
- Medical imaging: partitioning images based on tissue structures
- Market segmentation: clustering customers based on demographics, income, etc.
- Mails:

No labels!



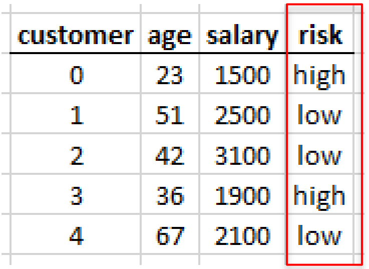
E-mail	Size	Recipients
1	8	1
2	12	1
3	43	1
4	10	2
5	40	2
6	25	5
7	23	6
8	28	6
9	26	7

mails.csv

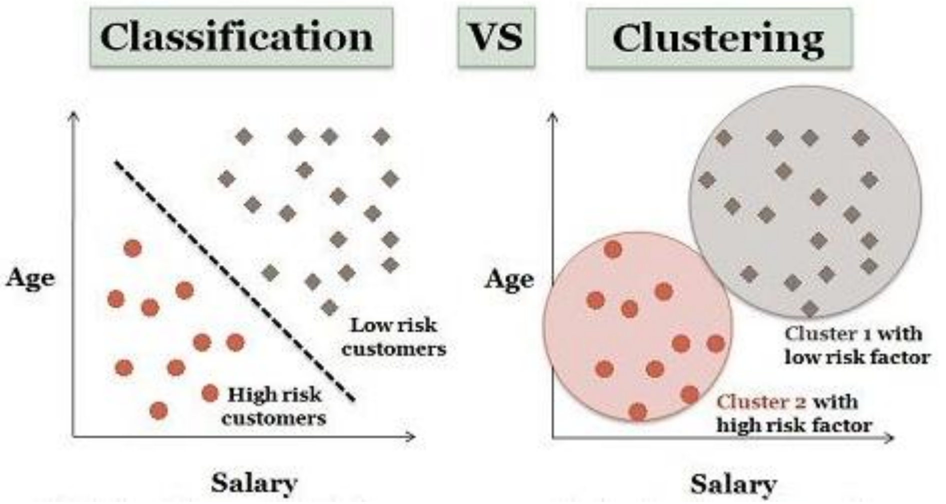


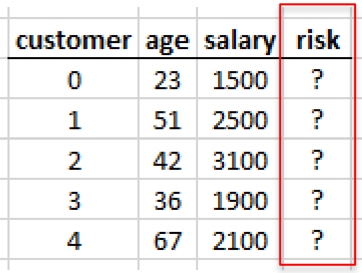
Clustering vs Classification

- **Classification**: labeled data → classes already exist
- **Clustering**: unlabeled data → classes don’t exist yet



customer	age	salary	risk
0	23	1500	high
1	51	2500	low
2	42	3100	low
3	36	1900	high
4	67	2100	low





customer	age	salary	risk
0	23	1500	?
1	51	2500	?
2	42	3100	?
3	36	1900	?
4	67	2100	?

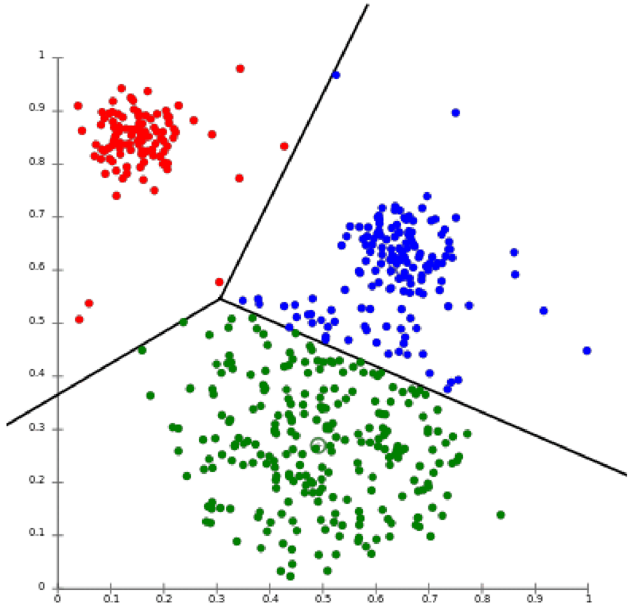
(source: https://techdifferences.com/difference-between-classification-and-clustering.html)

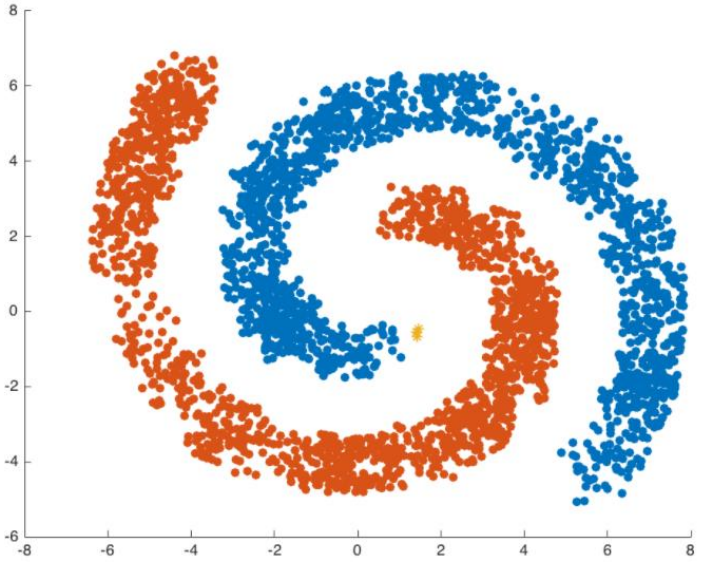
Clustering Algorithms

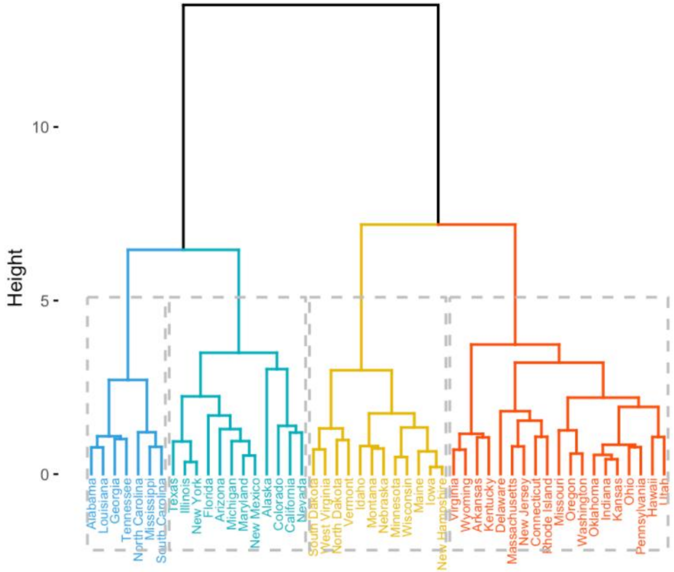
- K-means clustering

  https://youtu.be/nXY6PxAaOk0

- Hierarchical clustering (dendrogram)
- Gaussian mixture models
- DBSCAN
- …





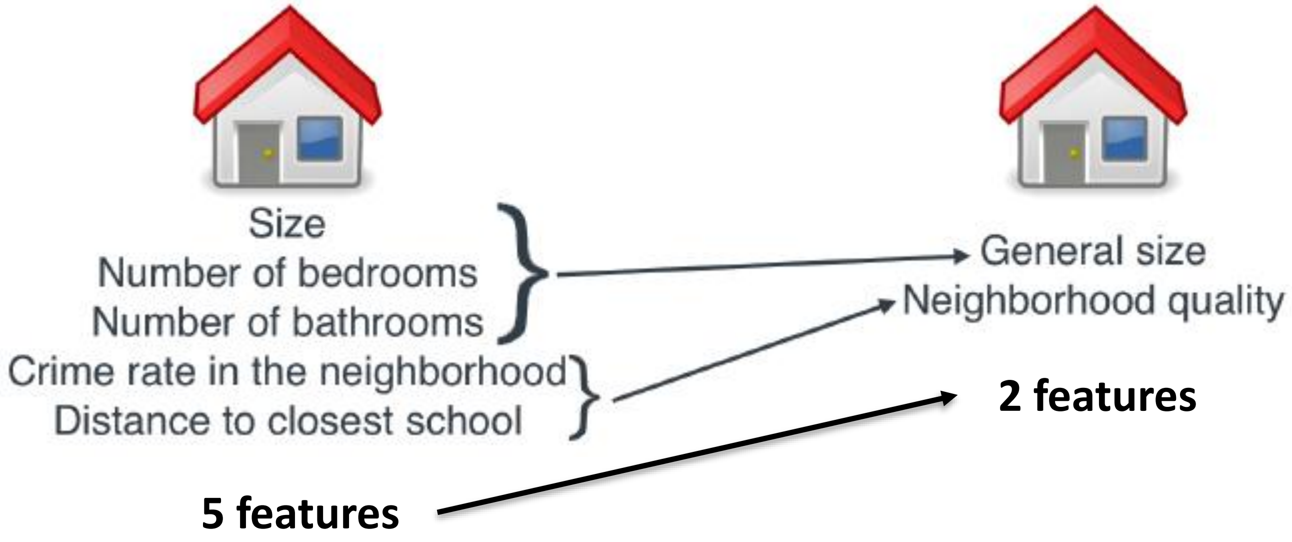


TALLINNA TEHNIKAKÕRGKOOL
TTK UNIVERSITY OF APPLIED SCIENCES

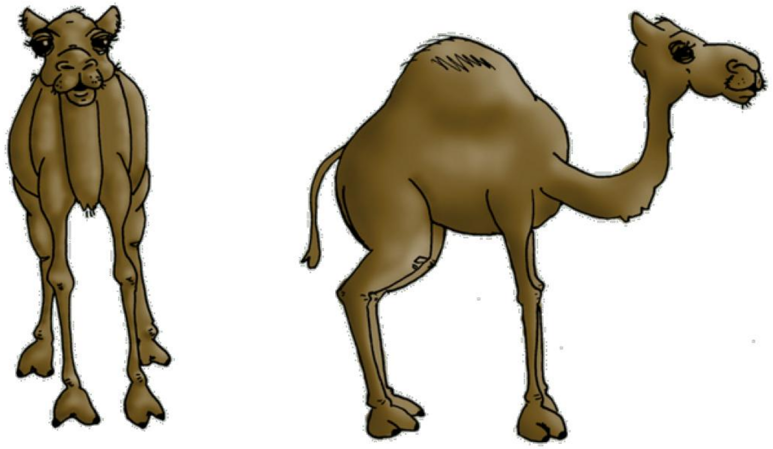
Dimensionality Reduction

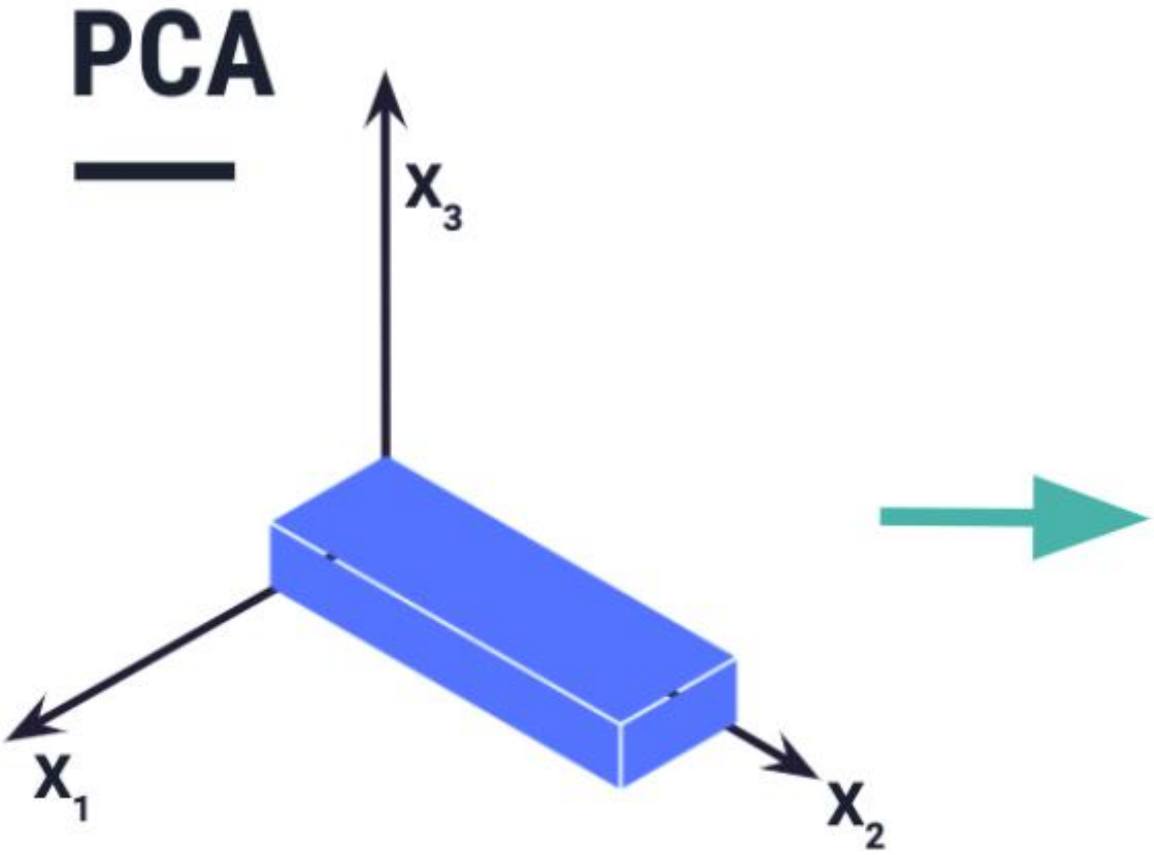
- Number of dimensions = number of features

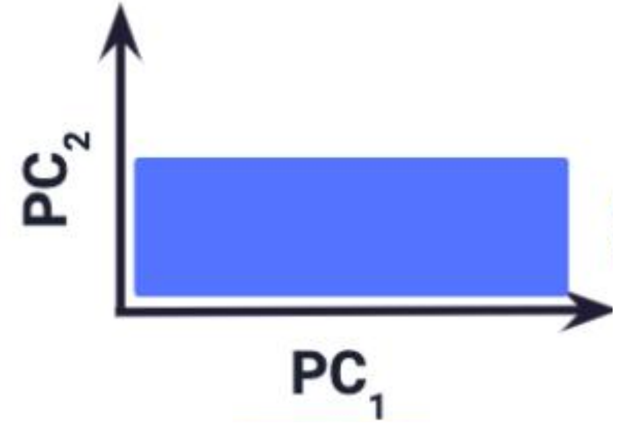
- Reducing the number of features



Principal Component Analysis

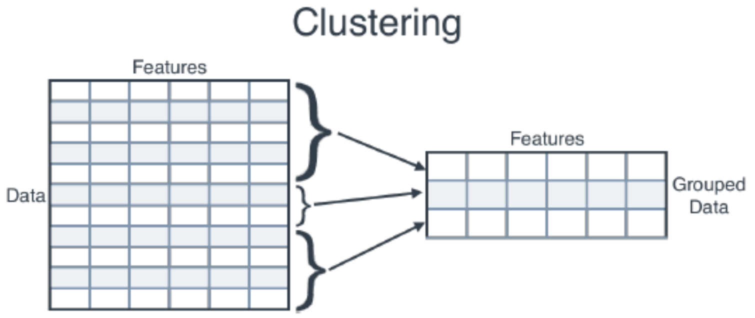




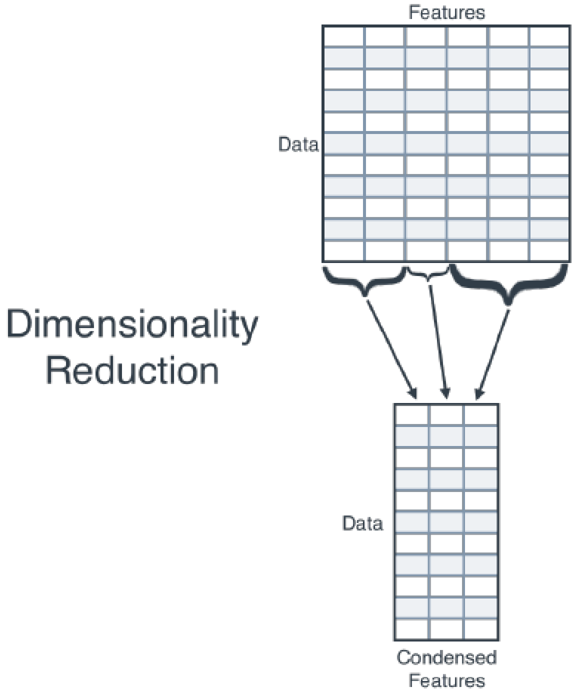


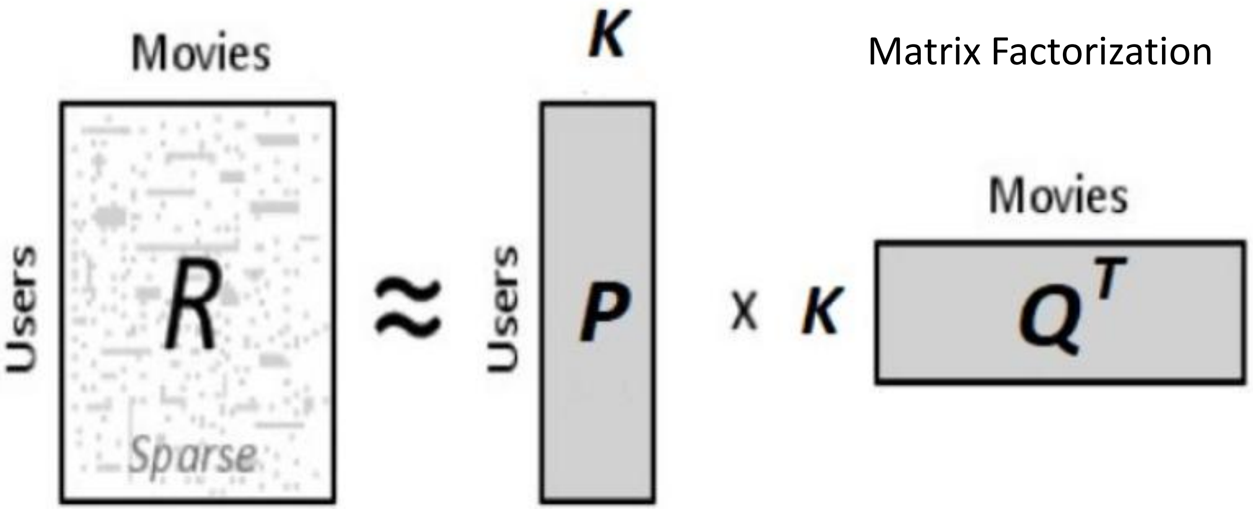
(source: https://knowledge.dataiku.com/latest/ml-analytics/statistics/concept-principal-component-analysis-pca.html)

Matrix Factorization



- **Clustering**: reducing samples (= rows)
- **Dimensionality Reduction**: reducing features (= columns)
- **Matrix Factorization**: reducing both rows and columns

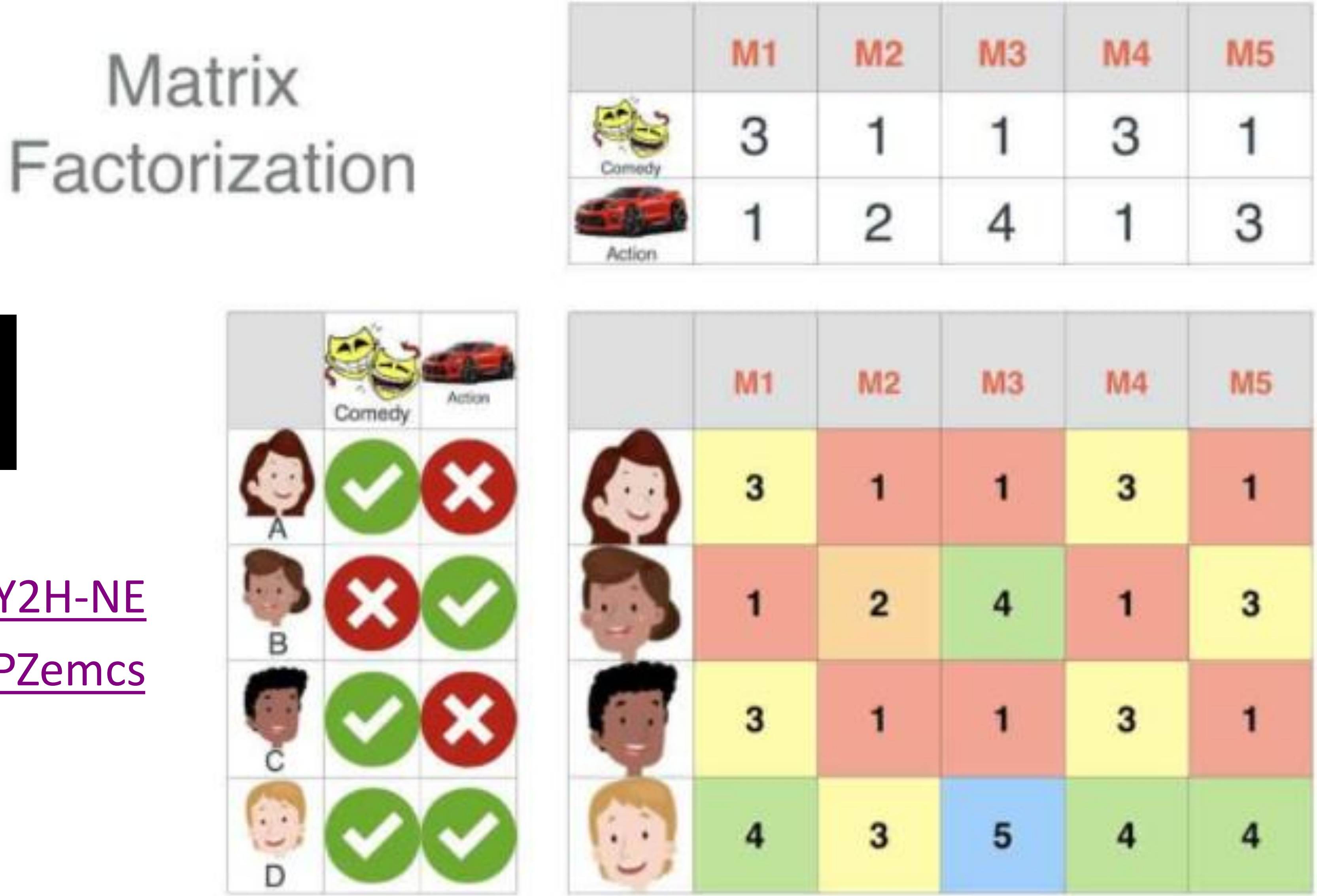




Matrix Factorization

(source: https://www.kaggle.com/code/residentmario/notes-on-matrix-factorization-machines)

Recommender Systems



Matrix Factorization

	M1	M2	M3	M4	M5
Comedy	3	1	1	3	1
Action	1	2	4	1	3

	Comedy	Action
A	✓	✗
B	✗	✓
C	✓	✗
D	✓	✓

	M1	M2	M3	M4	M5
	3	1	1	3	1
	1	2	4	1	3
	3	1	1	3	1
	4	3	5	4	4

NETFLIX

- https://youtu.be/n3RKsY2H-NE
- https://youtu.be/ZspR5PZemcs

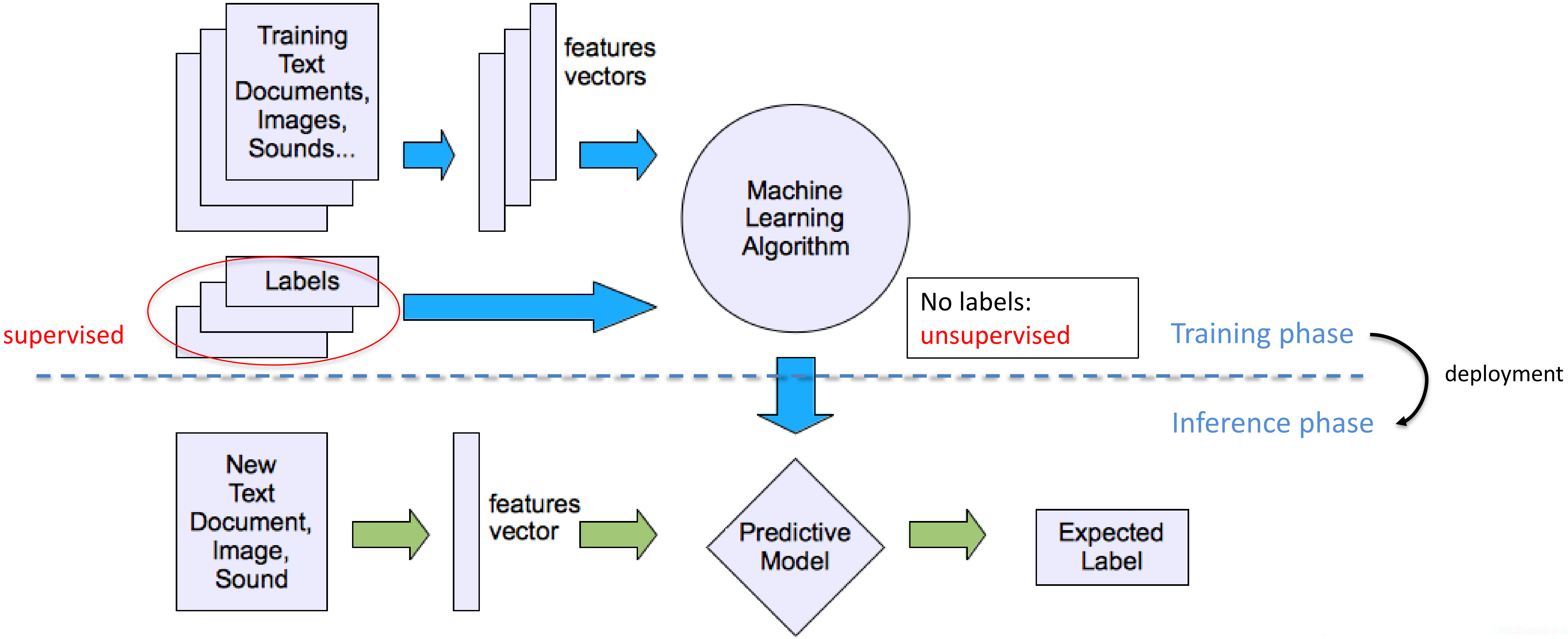
TALLINNA TEHNIKAKÕRGKOOL
TTK UNIVERSITY OF APPLIED SCIENCES

Guest Lectures at TTK University of Applied Sciences, Tallinn, Estonia

Introduction to Machine Learning

TRAINING AND EVALUATION

Training vs Inference

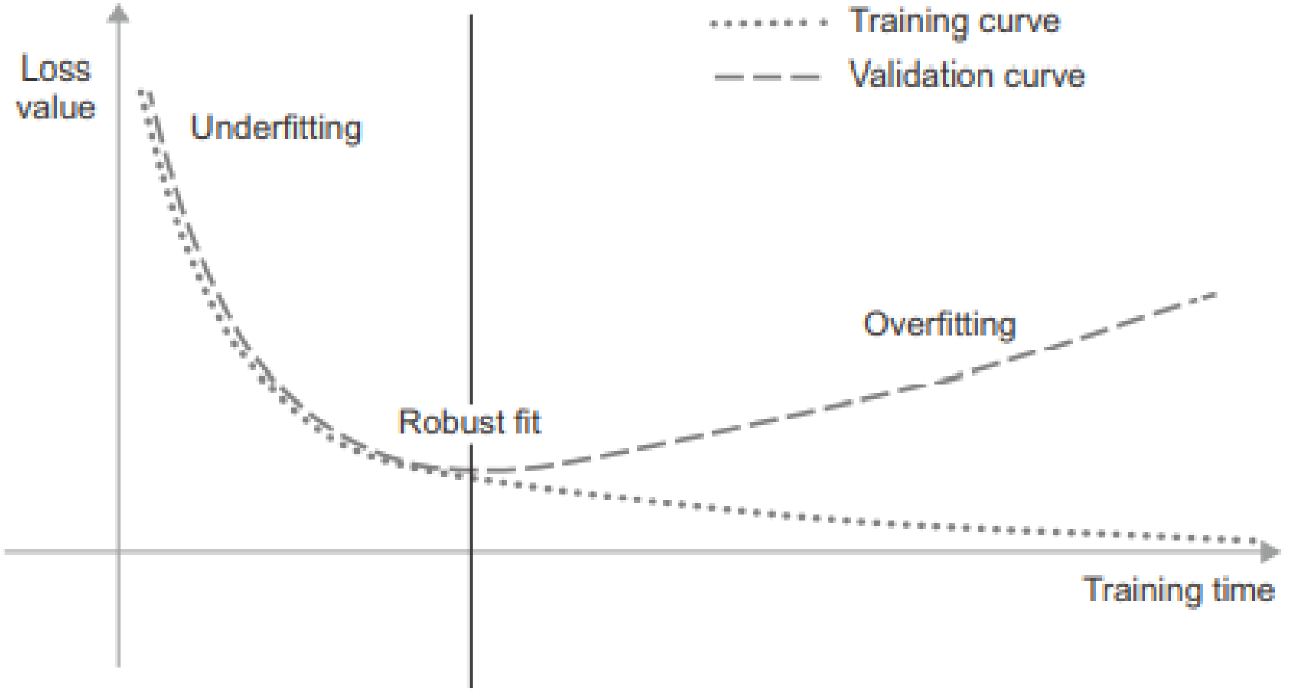


Optimization vs Generalization

- **<u>Optimization</u>:** fitting the data as best as possible during <u>training</u>

↕

- **<u>Generalization</u>:** good model performance on new data during <u>inference</u>



Underfitting vs Overfitting

- **Underfitting:**

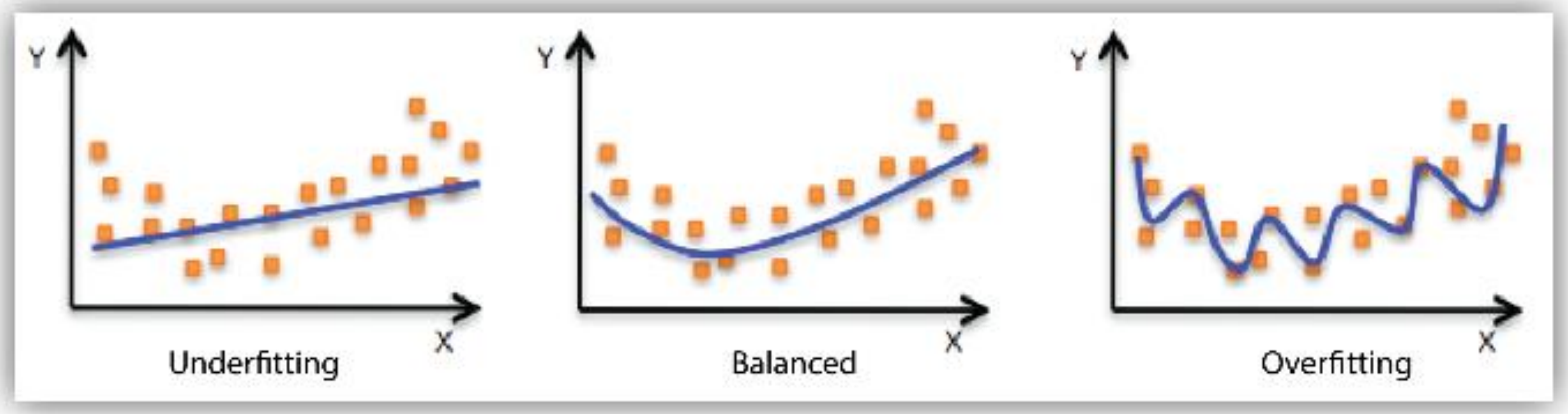
  model is too simple

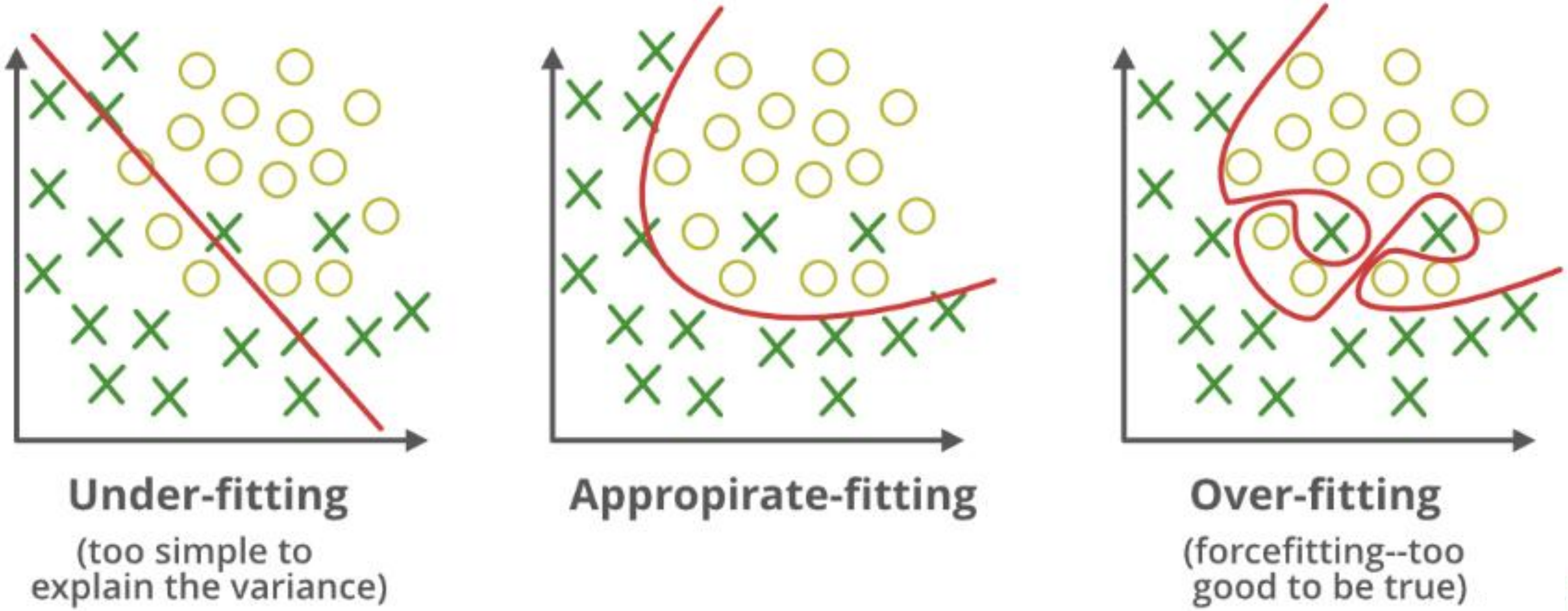
- **Overfitting:**

  model is too specific

  Causes:

  - Noise
  - Uncertainty
  - Rare features
  - …

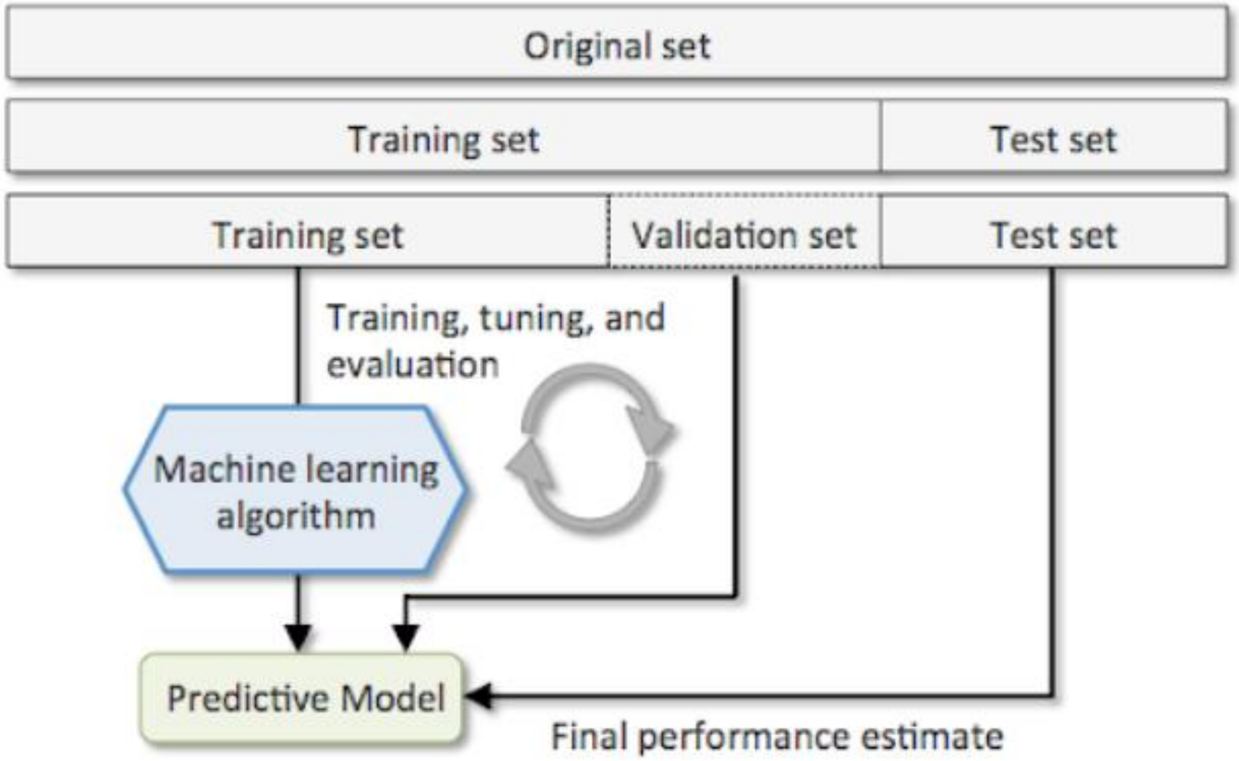




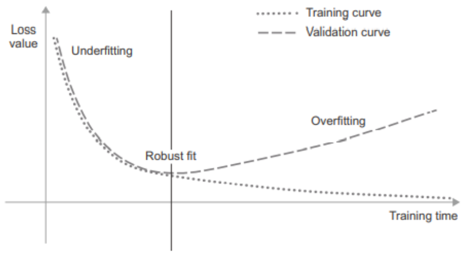
Training, Validating, and Testing

Splitting the dataset:

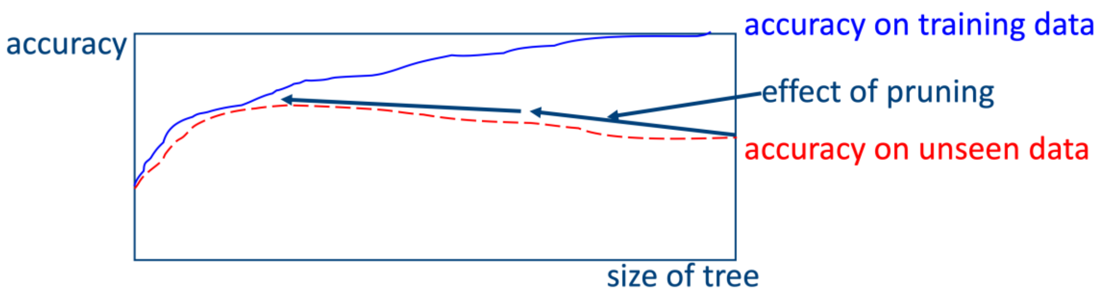
- **training**:
  - deriving the optimal model parameters
  - by the machine learning algorithm
- **validating**:
  - finding the optimal model configuration
  - tuning the hyperparameters
  - to overcome overfitting
  - by the data scientist
- **testing**:
  - final evaluation
  - by the data scientist


Original set
Training set
Test set
Training set
Validation set
Test set
Training, tuning, and evaluation
Machine learning algorithm
Predictive Model
Final performance estimate

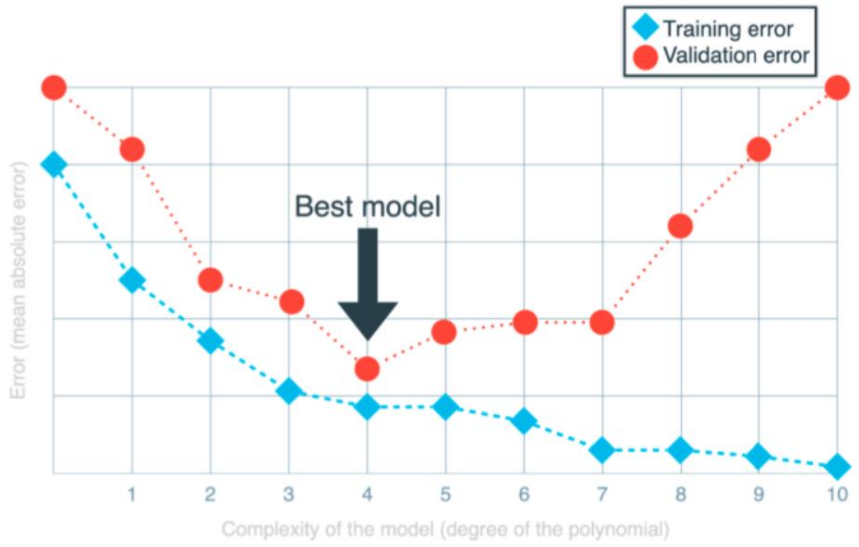
Training vs Validation Performance



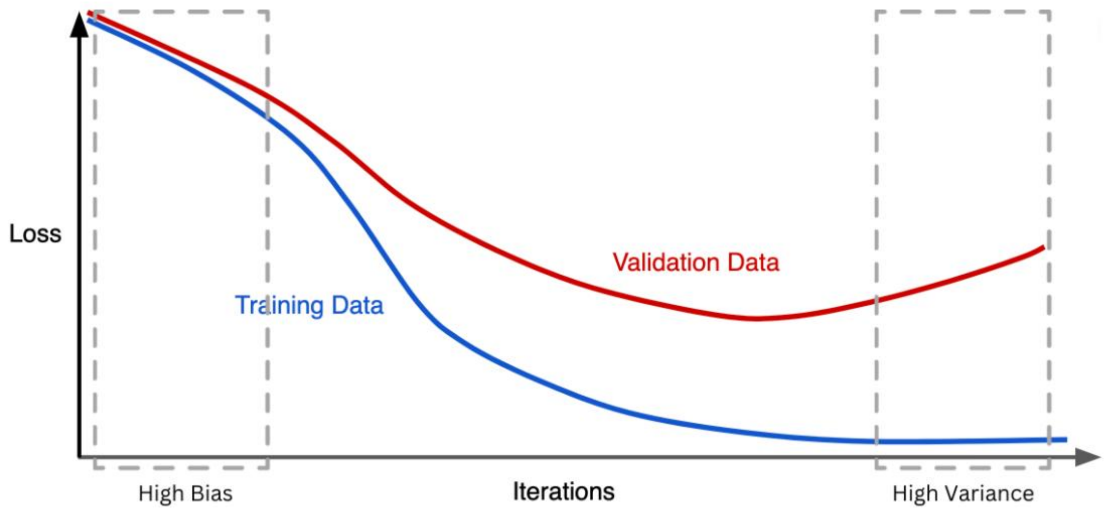
- **Decision Trees**



- **Polynomial Regression**



- **Artificial Neural Networks**



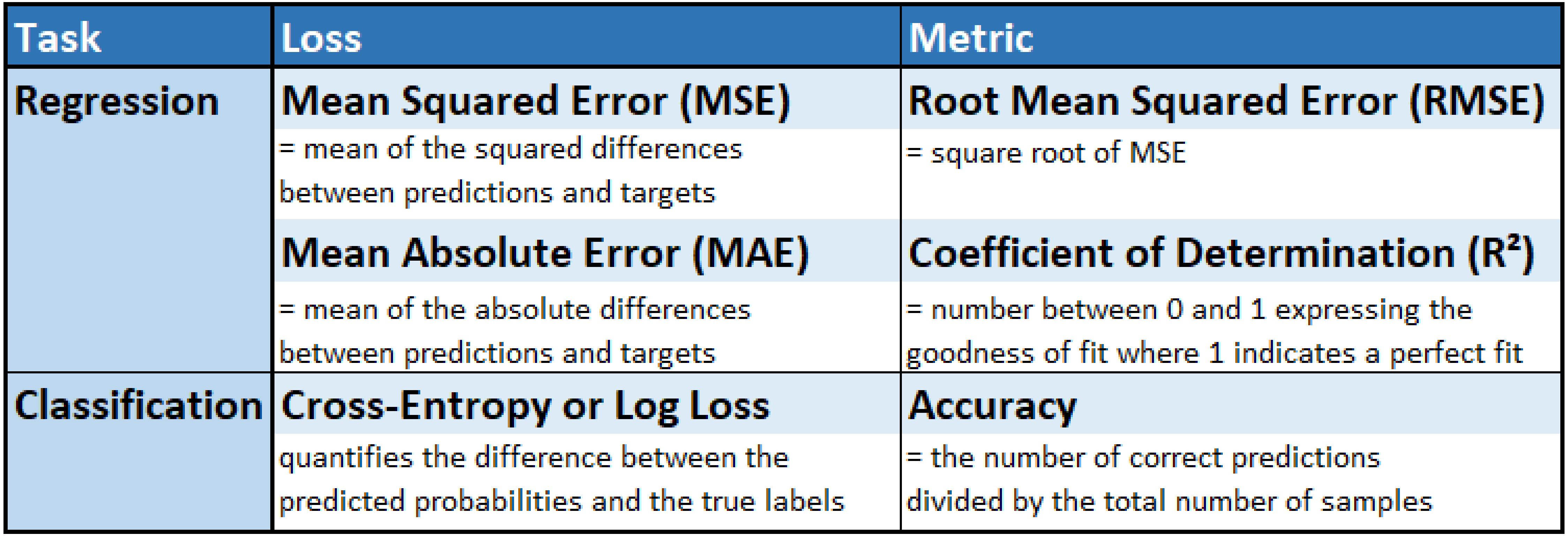
(source: https://www.dataquest.io/blog/regularization-in-machine-learning)

Model Evaluation

- **Loss functions:**
  - Compare predictions and true target values
  - Minimized by machine learning algorithms to obtain the best fit of the data
  - Should be mathematically convenient

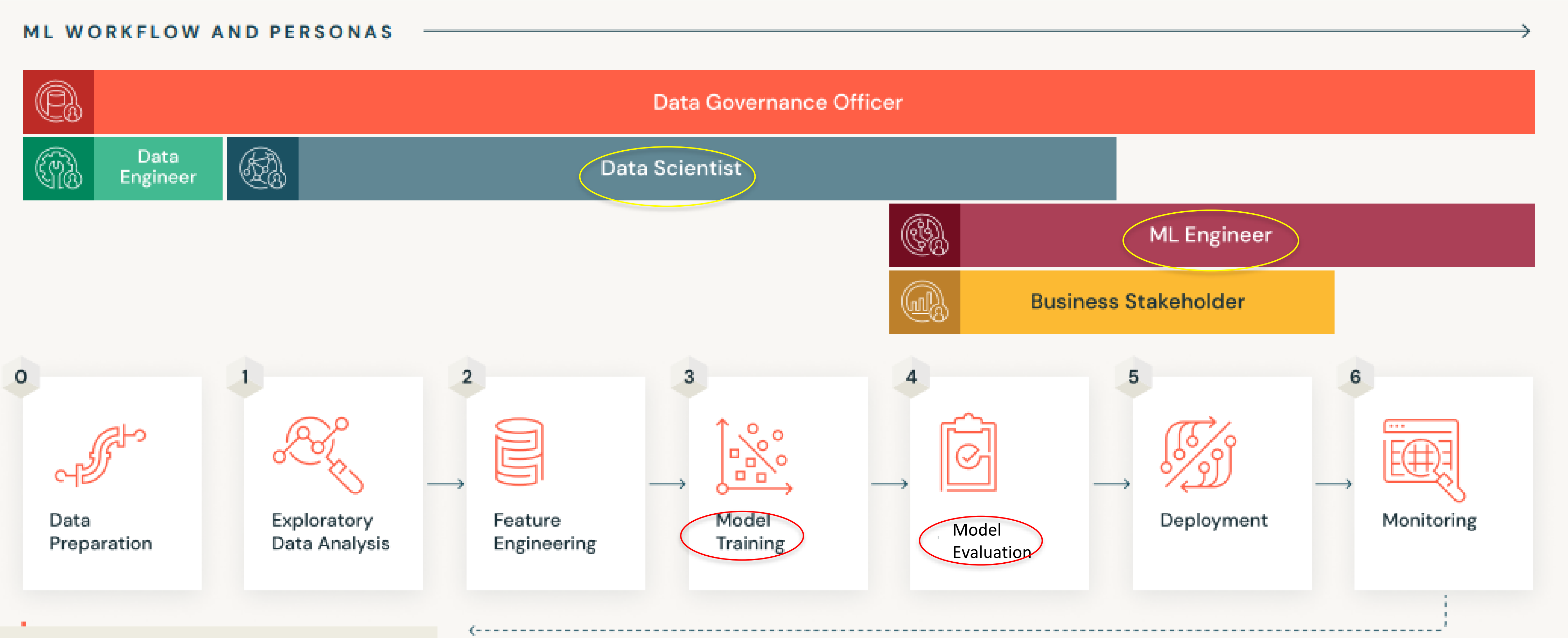
- **Evaluation metrics:**
  - Also compare predictions and true target values
  - Used by machine learning engineers to evaluate the model performance
  - Easier to interpret by humans

Common Loss Functions and Metrics



Task	Loss	Metric
Regression	**Mean Squared Error (MSE)** = mean of the squared differences between predictions and targets	**Root Mean Squared Error (RMSE)** = square root of MSE
	Mean Absolute Error (MAE) = mean of the absolute differences between predictions and targets	**Coefficient of Determination (R^2)** = number between 0 and 1 expressing the goodness of fit where 1 indicates a perfect fit
Classification	**Cross-Entropy or Log Loss** quantifies the difference between the predicted probabilities and the true labels	**Accuracy** = the number of correct predictions divided by the total number of samples

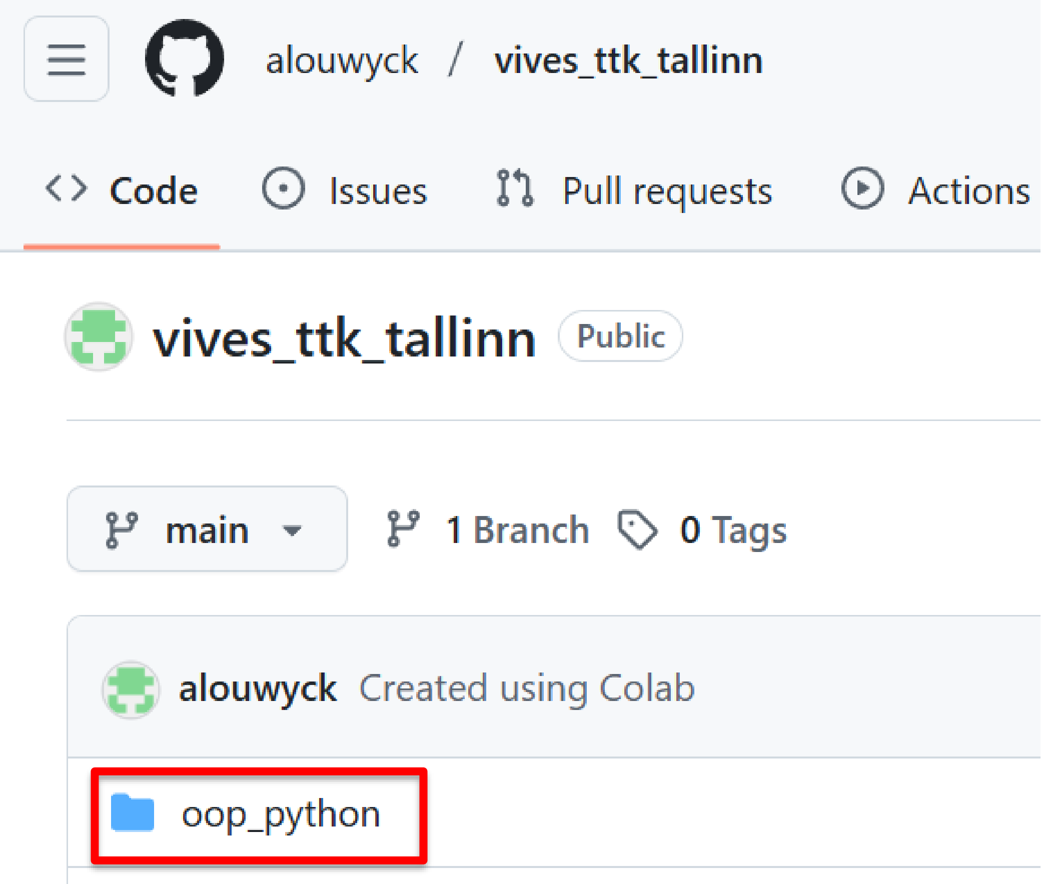
Machine Learning Workflow



The Big Book of MLops (Databricks, 2023)

GitHub Repo

https://github.com/alouwyck/vives_ttk_tallinn



alouwyck / vives_ttk_tallinn

vives_ttk_tallinn Public

alouwyck Created using Colab

oop_python

Sources

- Many slides are based on the book “Grokking Machine Learning” by Luis G Serrano (2021)
- Other slides are inspired by the book “Deep Learning with Python (2nd edition)” by François Chollet (2021)
- Some slides are adopted from the presentation on machine learning that was part of the course “Introduction to Artificial Intelligence” taught by Dr. Stefaan Haspeslagh at the Vives University of Applied Sciences during the academic year 2019-2020
- A few slides are taken from lectures given by Prof. Dr. Celine Vens and Prof. Dr. Hendrik Blockeel (Computer Sciences, KUL)
- Information was also obtained from Andrew Ng’s online course “AI for Everyone”: https://www.deeplearning.ai/courses/ai-for-everyone/
- Scikit-Learn’s User Guide was also consulted: https://scikit-learn.org/stable/user_guide.html
- Other sources are mentioned on the slides