Universidad Zaragoza

Machine Learning (69152)

Machine Learning Fundamentals

Universidad Zaragoza

Rubén Martínez Cantín
rmcantin@unizar.es, office 0.09, Ada Byron bldg
Dpto. Informática e Ingeniería de Sistemas.

Content

DISCLAIMER: SLIDES WILL BE UPDATED DURING THE COURSE, DOWNLOAD THE MOST UPDATED VERSION FROM MOODLE

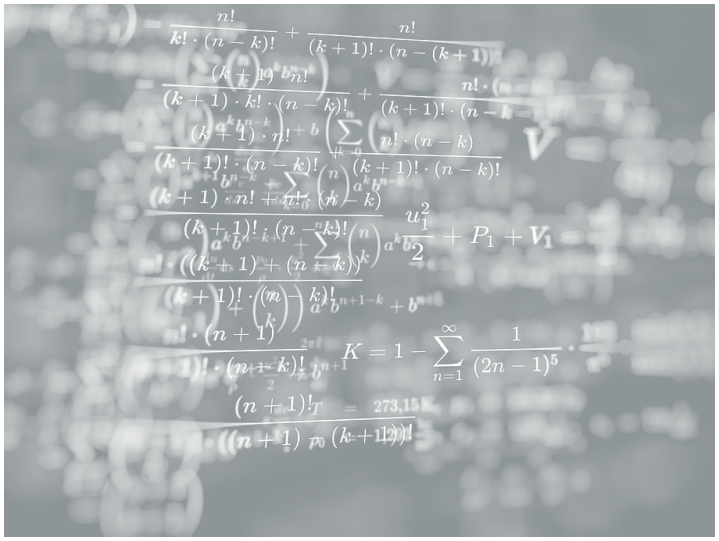
- **Introduction**
- **K-nearest neighbours (as a simple example)**
- **Regression**
    - Linear regression
    - Non-linear regression
    - Overfitting, regularization, model selection
    - Robust regression
    - Ridge regression
- **Classification**
    - Logistic regression
    - SVM
- **Unsupervised methods**
    - PCA
- **Evaluation metrics**

Bibliography

- **Kevin P. Murphy, Probabilistic Machine Learning: An Introduction, The MIT Press, 2022** (available online https://probml.github.io/pml-book/book1.html)
- Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong, Mathematics for Machine Learning, Cambridge University Press, 2020 (available online https://mml-book.com/)
- Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006 (available online https://www.microsoft.com/en-us/research/uploads/prod/2006/01/Bishop-Pattern-Recognition-and-Machine-Learning-2006.pdf)
- Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow, O'Reilly, 2019 (2nd edition)
- Online courses
  - Andrew Ng, Stanford CS229: Machine Learning, https://www.youtube.com/playlist?list=PLoROMvodv4rMiGQp3WXShtMGgzqpfVfbU
  - Ayush Singh and @freecodecamp, ML for beginners, https://www.youtube.com/watch?v=NWONeJKn6kc
- Python resources
  - The Python tutorial: https://docs.python.org/3/tutorial/
  - numpy tutorial: https://cs231n.github.io/python-numpy-tutorial/
  - scikit-learn tutorials: https://scikit-learn.org/stable/tutorial/index.html

Motivation

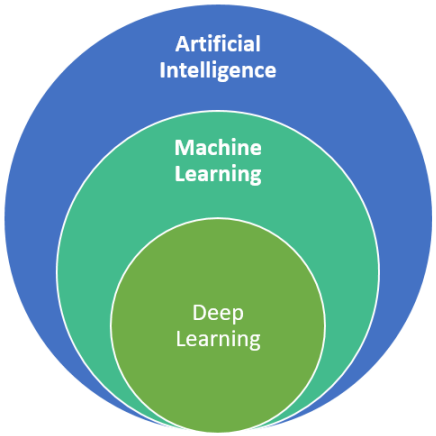
- Machine learning has a huge presence (and growing)…
  - Recommenders, virtual assistants, content generation (intelligent virtual characters, automatic scene creation), spam filtering, search engines, social media, news classification, video surveillance, face recognition, fraud detection, traffic prediction, image retrieval, predictive maintenance, weather forecasting, autonomous driving, speech recognition, automatic translation, medical diagnosis…
- Starting by the fundamentals (and getting deep into math!) is critical for a structured understanding of the field.

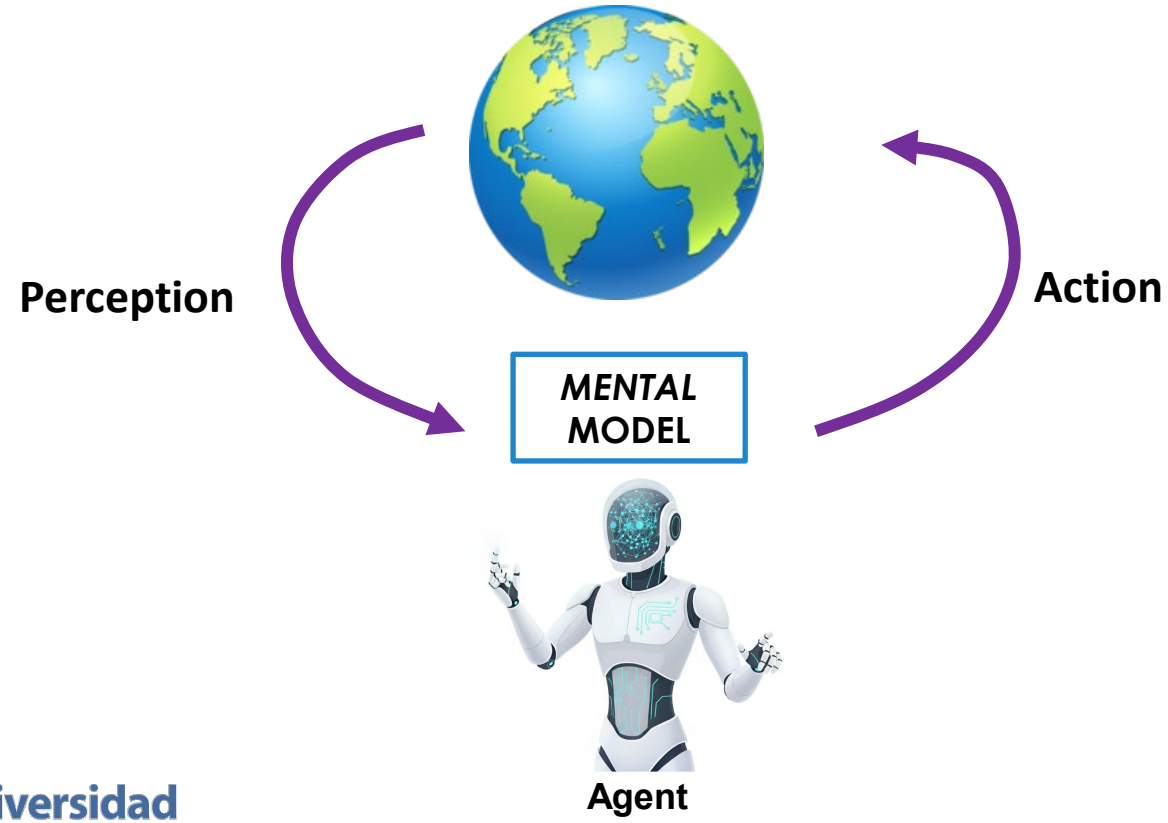


Inteligent Agent

Machine Learning:

Field of study that gives computers the ability to learn without being explicitly programmed (Arthur Samuel 1959).





Machine learning is NOT magic!

- Careful methodology, iterative design and data analysis can be motivated by the No Free Lunch theorem [Wolpert 96].
  - A model is a simplified version of the data
  - Assumptions (~hyperparameters, model structure/architecture, inductive bias) are always there, influencing what should be accounted for (signal, pattern) and what should be ignored (noise) in the data
  - No model is guaranteed a priori to be better than others, it just should "fit" our data (or the structure in it)
  - Usual practice: Evaluate a few reasonable models making reasonable assumptions on your data.

Sometimes it seems magical though…

- **Example:** Neural Radiance Fields
  - Use neural networks (MLPs) for representing scenes and rendering novel views, costly to train and query

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, Ren Ng, NeRF: Representing scenes as neural radiance fields for view synthesis, ECCV 2020
Thomas Müller and Alex Evans and Christoph Schied and Alexander Keller, Instant Neural Graphics Primitives with a Multiresolution Hash Encoding, ACM ToG, 2022

Sometimes it seems magical though…

- **Example:** Neural Radiance Fields
  - Use neural networks (MLPs) for representing scenes and rendering novel views, costly to train and query

Shallow learning also works!

- Scene representation based on Gaussians in the 3D space
- Well implemented but standard point-based rendering
- Enables real-time capabilities at 1080p

Kerbl, B., Kopanas, G., Leimkühler, T., & Drettakis, G., 3D Gaussian Splatting for real-time radiance field rendering, ACM ToG, 2023

Machine learning

- The world is immensely rich in data

$$X = \begin{pmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mn} \end{pmatrix}$$

- Goal: Making sense of data
  - Behavioral function, patterns, etc.

- Different types of learning depending on the problem
  - Supervised
  - Unsupervised
  - Reinforcement

Supervised Learning

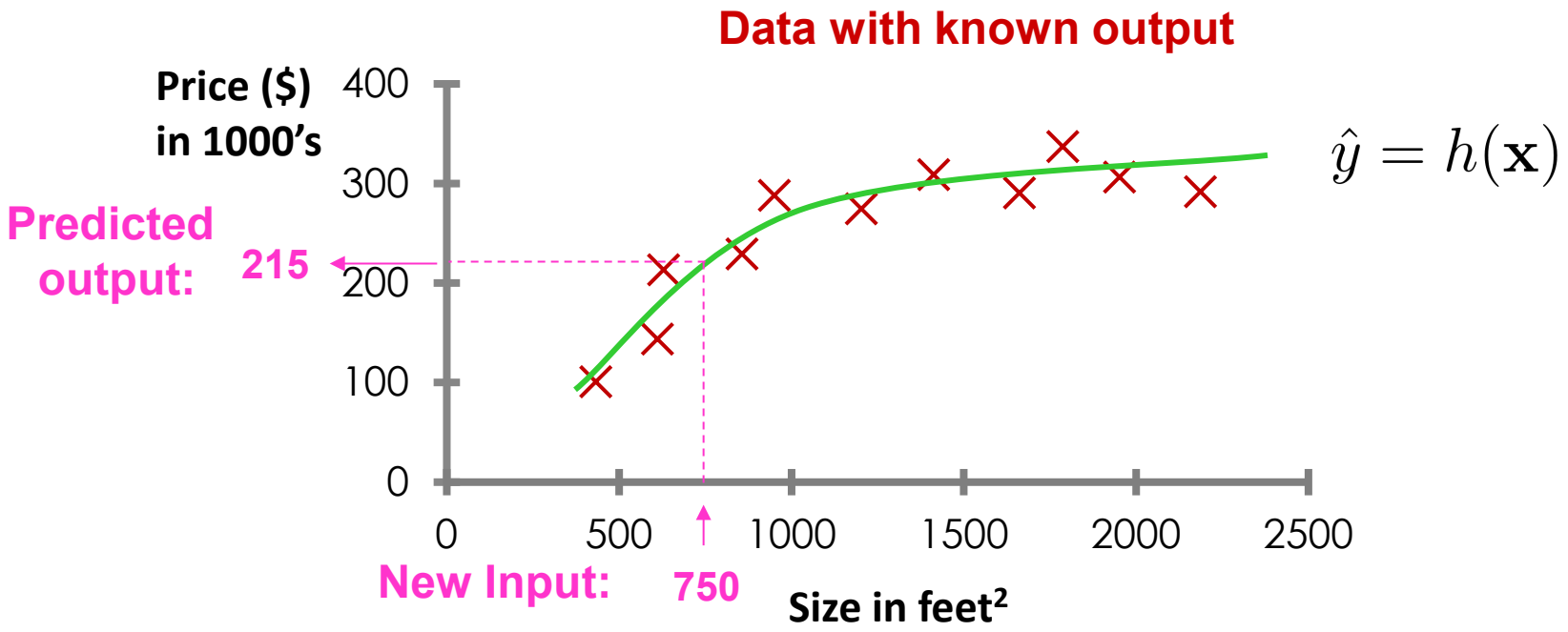
- For each element of the set X an associated output value is known, Y
- The output is generated by an unknown function $y = f(x)$

Features					**Output**	
x_{11}	x_{12}	$\cdots$	x_{1n}	$\rightarrow$	y_1	**Data**
x_{21}	x_{22}	$\cdots$	x_{2n}	$\rightarrow$	y_2	
	$\vdots$				$\vdots$	
x_{m1}	x_{m2}	$\cdots$	x_{mn}	$\rightarrow$	y_m	

- Discover a function $h(\cdot)$ that approximates the real function $f(\cdot)$ and allows predicting the output for future data $y = h(x)$

Supervised Learning: Regression

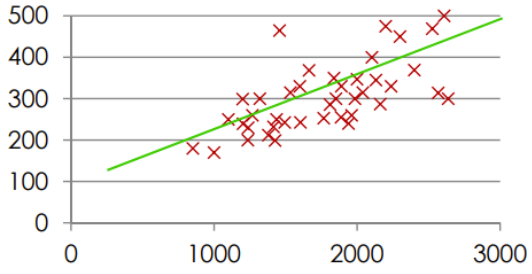
- **Continuous output**

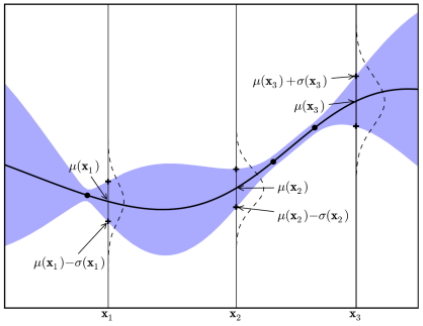


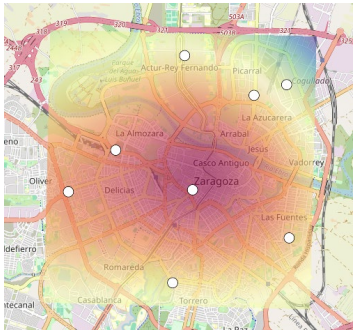
- Features: surface area, number of bedrooms, number of bathrooms, age,....

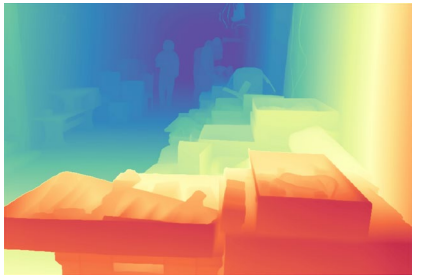
Regression Examples

- Number of units of a product to be sold
- Stock market value tomorrow based on current conditions and other factors
- Number of days each patient will spend in the hospital next year
- Level of degradation of a machine or product





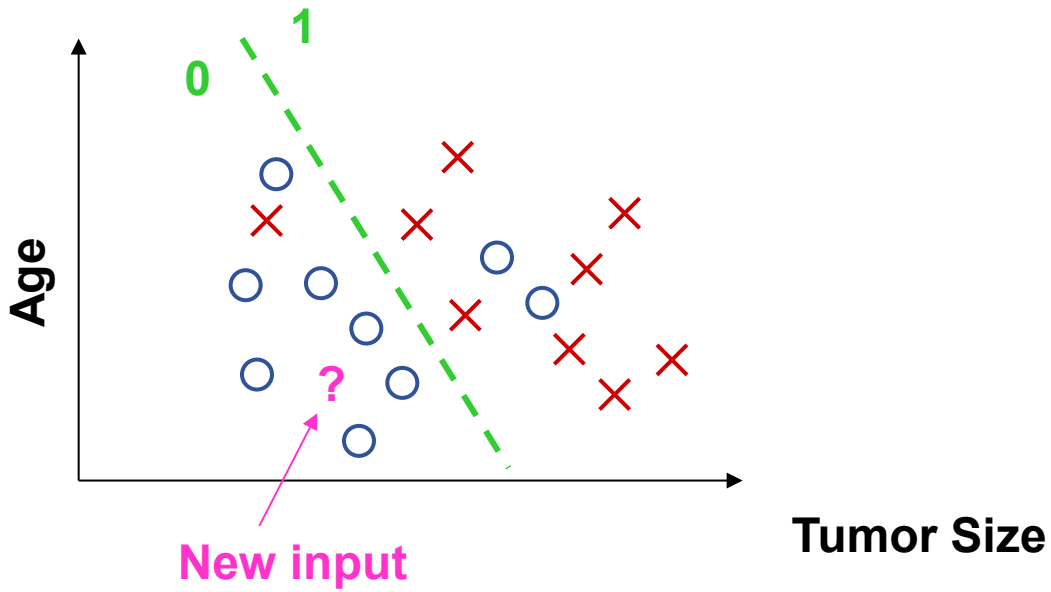




Zaragoza

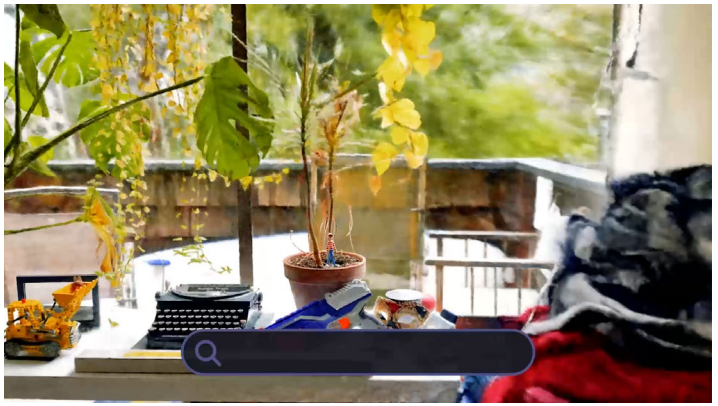
Supervised Learning: Classification

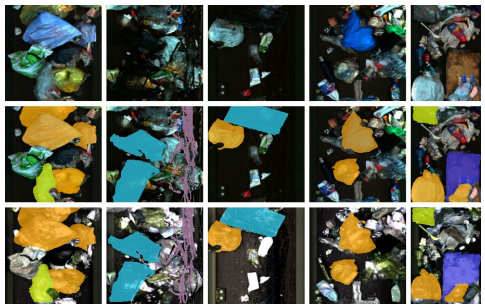
- **Discreet output**
  - **Binary:** $y = \{0,1\}$
  - **Multi-class:** $y = \{dog, cat, person, bed, \dots\}$

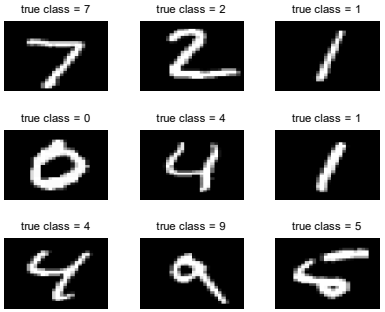


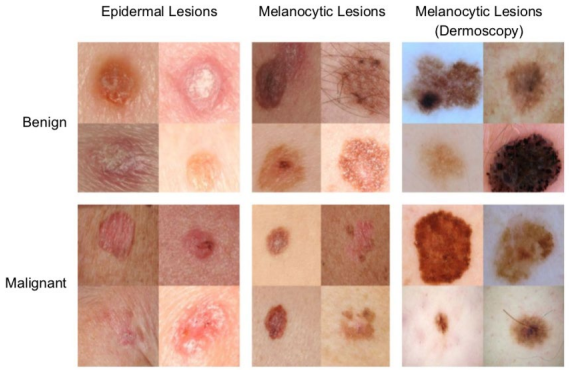
Classification examples

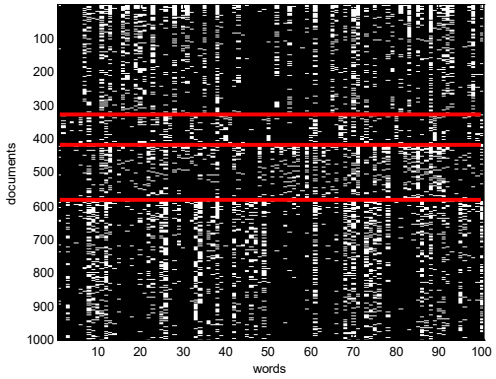
- Spam Filter
- Character Recognition
- Object Detection
- Semantic segmentation

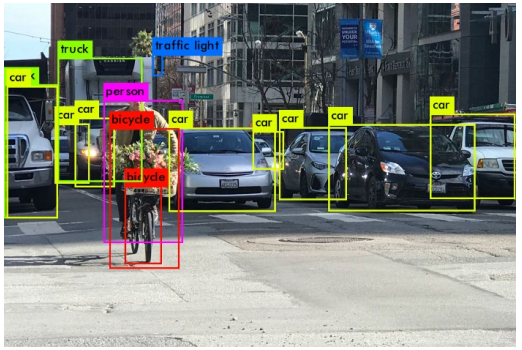








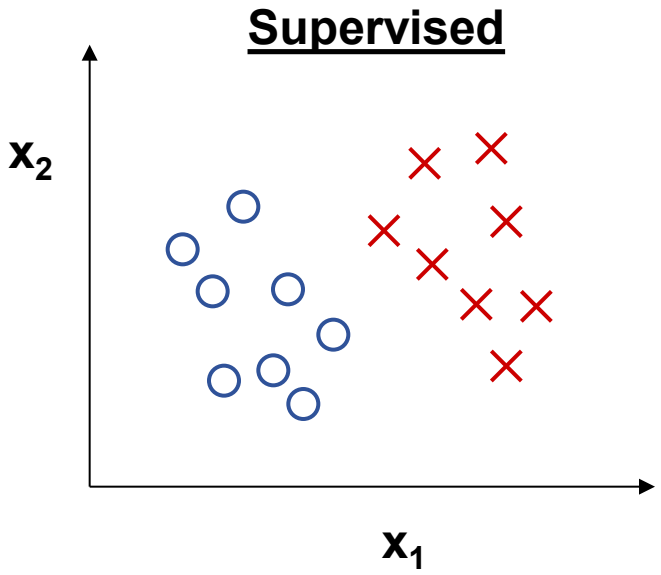


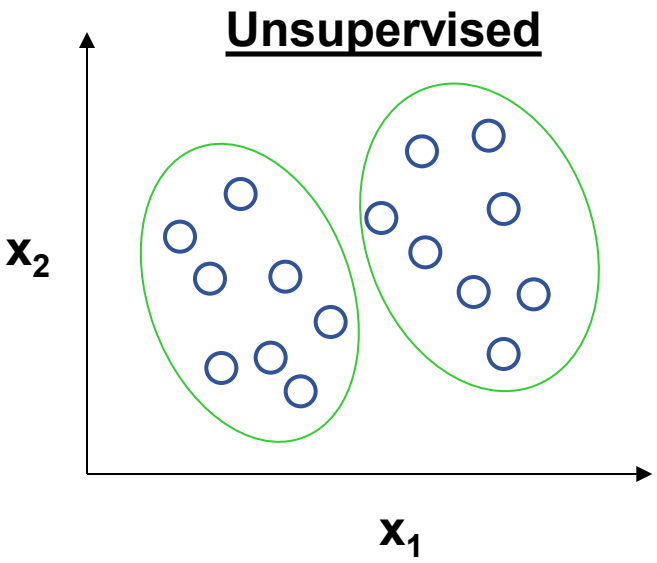


Zaragoza

Unsupervised Learning

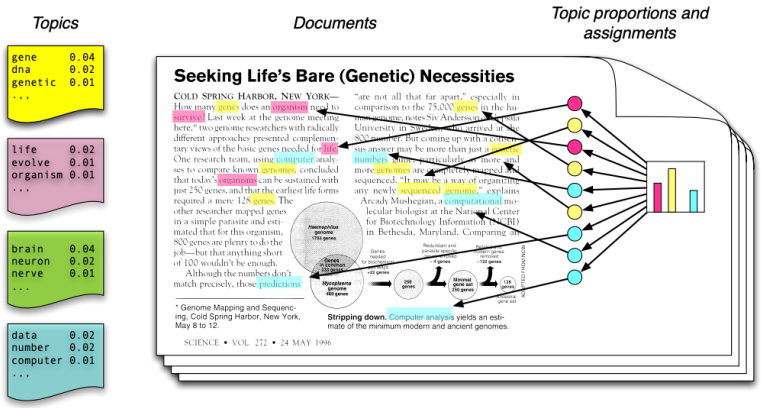
- Also called "knowledge discovery"
- Training data: inputs only, X
- Discover "interesting structure" in data (patterns)





Examples of unsupervised learning

- Astronomy: the *autoclass* system (1988) discovered a new type of star by grouping astrophysical measurements
- E-commerce: grouping users based on their browsing and purchases, to send them propaganda
- **Recommendation systems: series, news, social networks**
- **Text analysis: topic detection, next word prediction…**



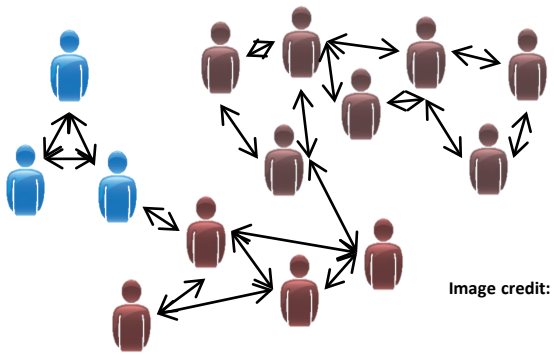
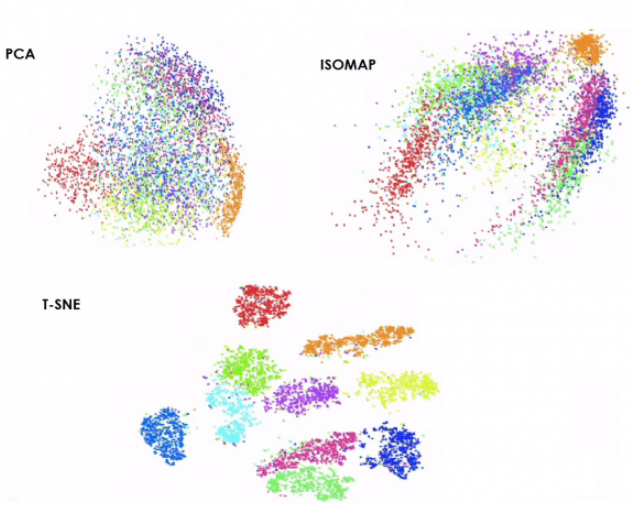


Image credit: NASA/JPL-Caltech/E. Churchwell (Univ. of Wisconsin, Madison)

Universidad Zaragoza

Examples of unsupervised learning

- Dimensionality reduction: visualization of complex systems



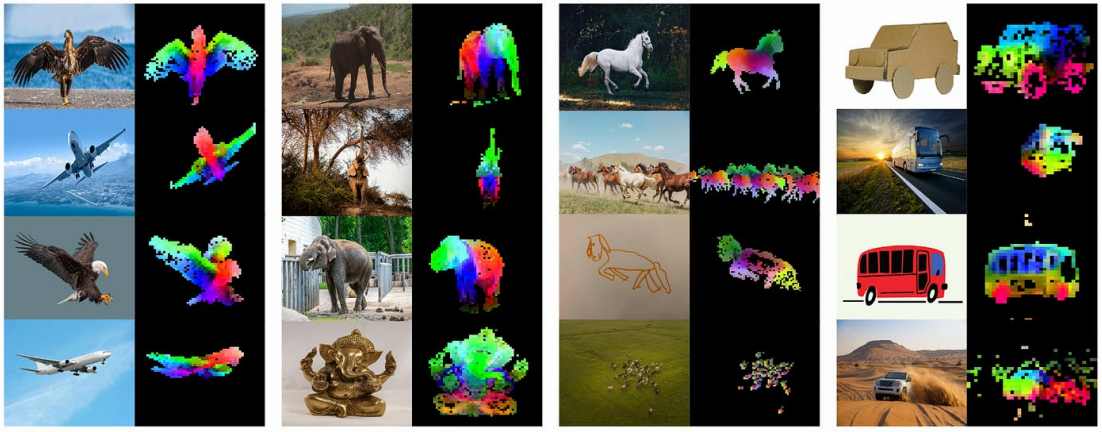


Figure 1: **Visualization of the first PCA components.** We compute a PCA between the patches of the images from the same column (a, b, c and d) and show their first 3 components. Each component is matched to a different color channel. Same parts are matched between related images despite changes of pose, style or even objects. Background is removed by thresholding the first PCA component.

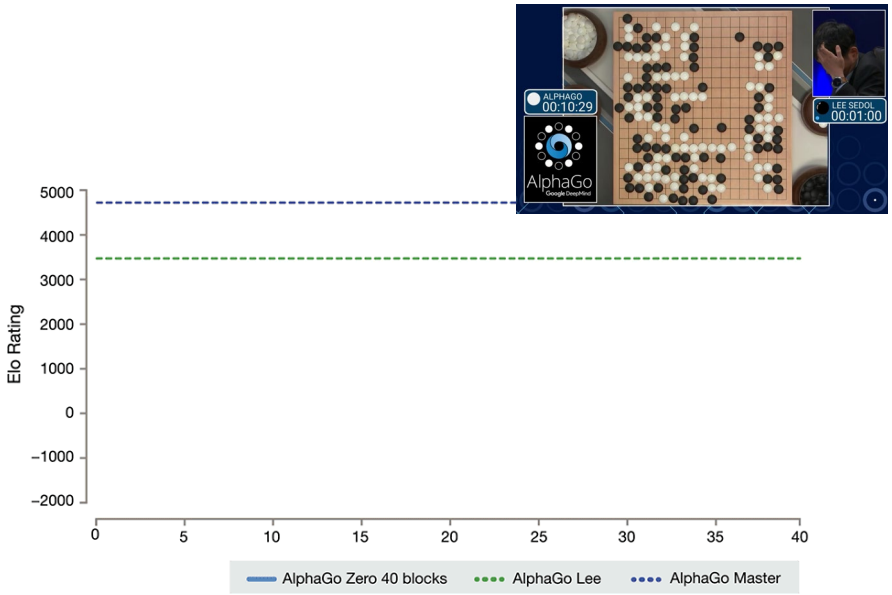
Reinforcement Learning

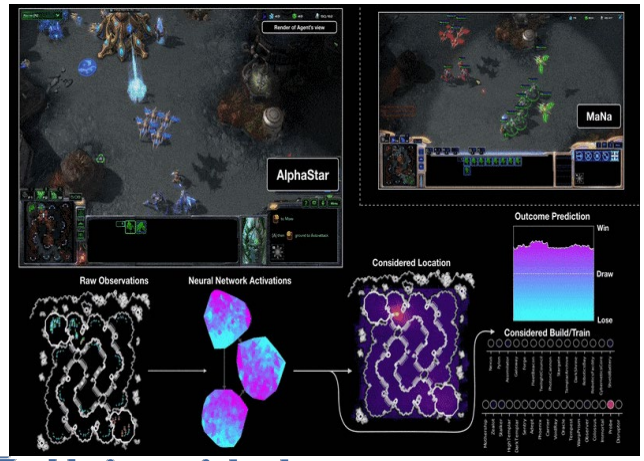
- *Reinforcement Learning* (RL)
- Learn behavioral policy
  - Robot movement commands
  - Strategy games
- The algorithm learns based on experiences (interactive)
- Rewards associated with good actions.
- Important time component
  - Present actions affect future outcomes

19

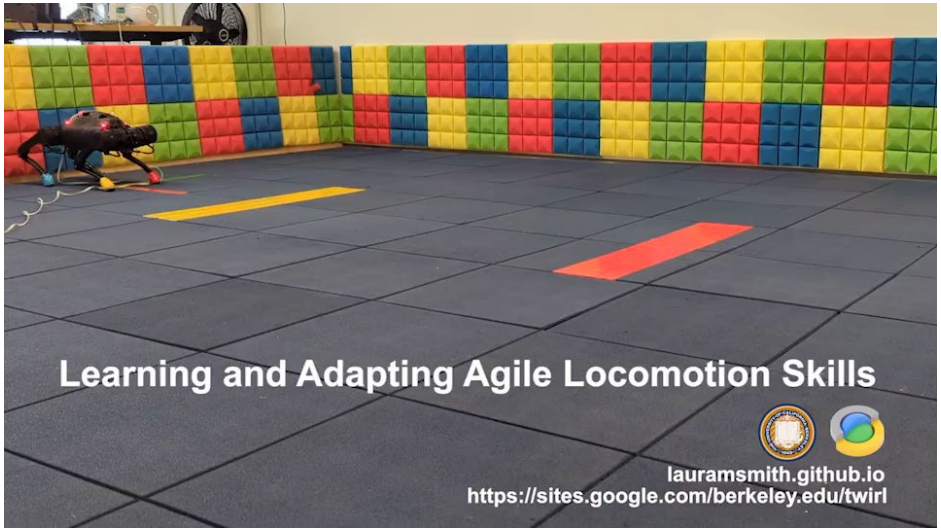
Examples of reinforcement learning

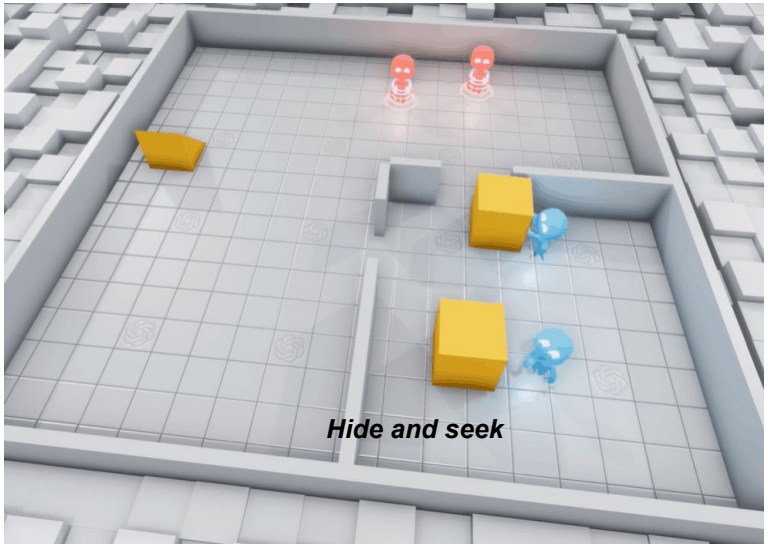
- AlphaGo
- Robotics
- Video games
- Autonomous driving

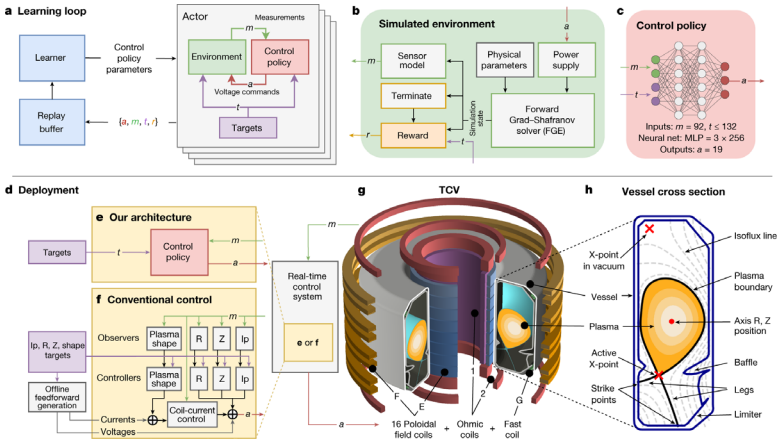


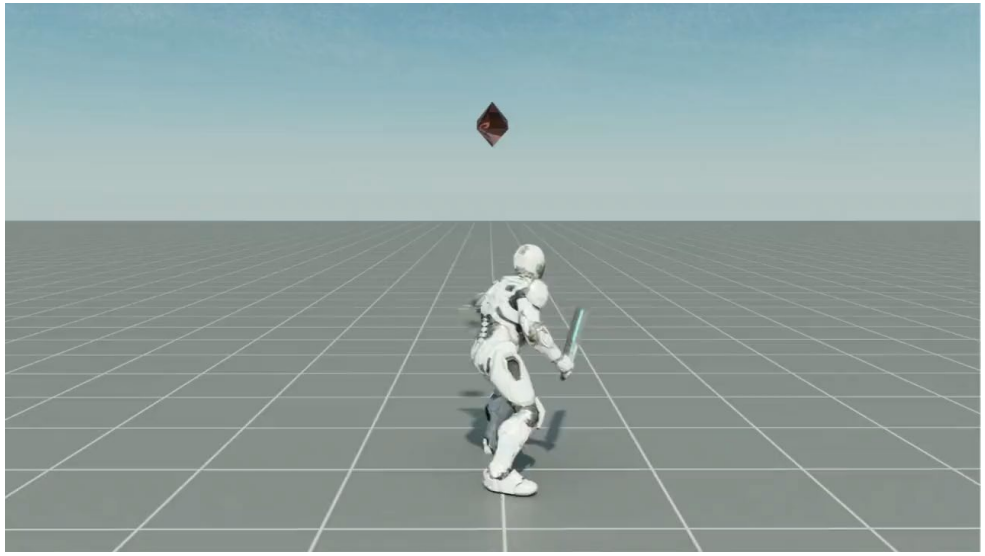


Examples of reinforcement learning





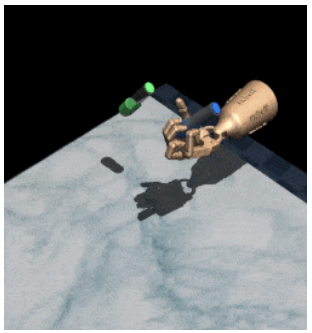


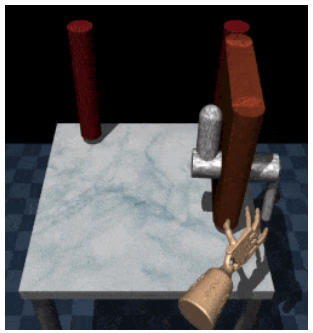


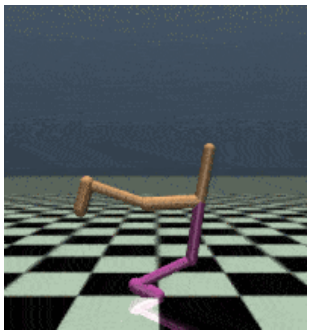
Character design for games and movies

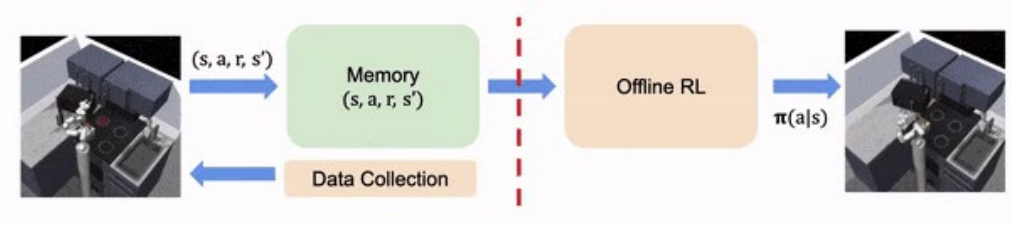
Universidad Zaragoza
1542

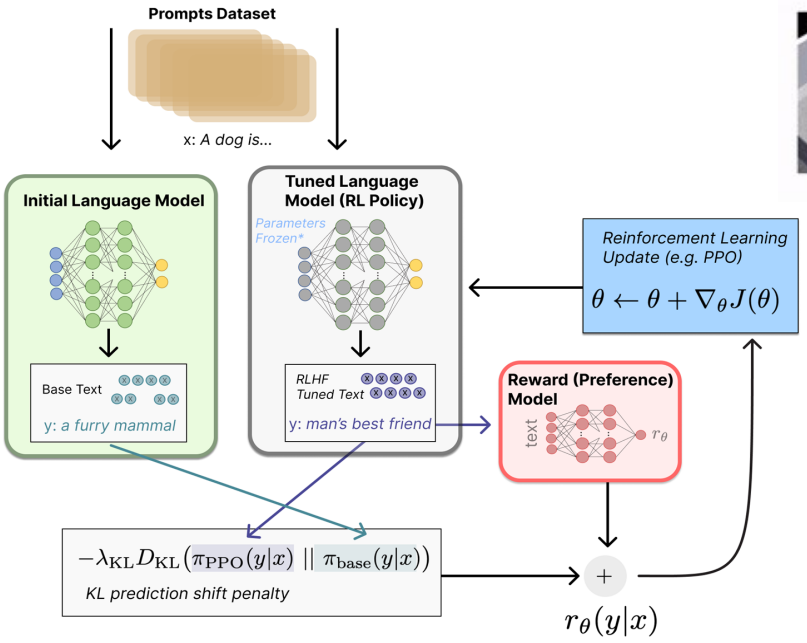
Examples of reinforcement learning

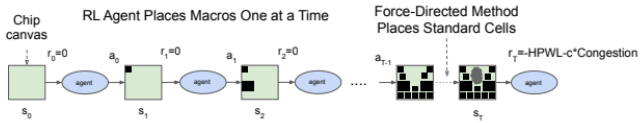


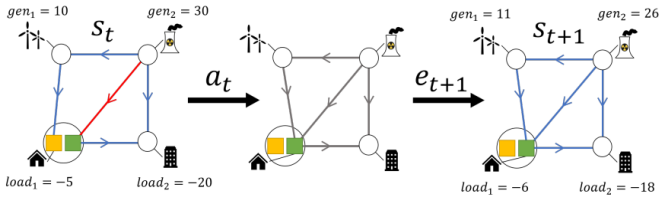












Combining ideas

Reinforcement Learning (cherry)

- The machine predicts a scalar reward given once in a while.
- **A few bits for some samples**

Supervised Learning (icing)

- The machine predicts a category or a few numbers for each input
- **10→10,000 bits per sample**

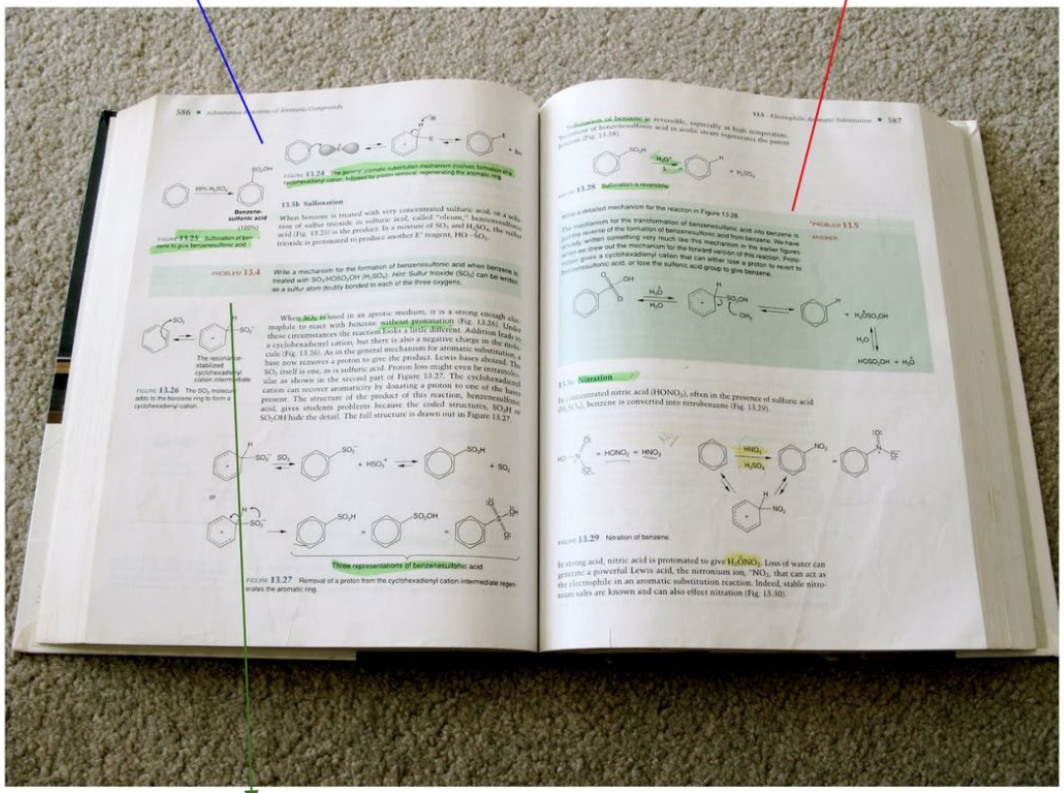
Unsupervised Learning (cake)

- The machine predicts any part of its input for any observed part.
- Predicts future frames in videos
- **Millions of bits per sample**

ChatGPT, DeepSeek, Gemini…

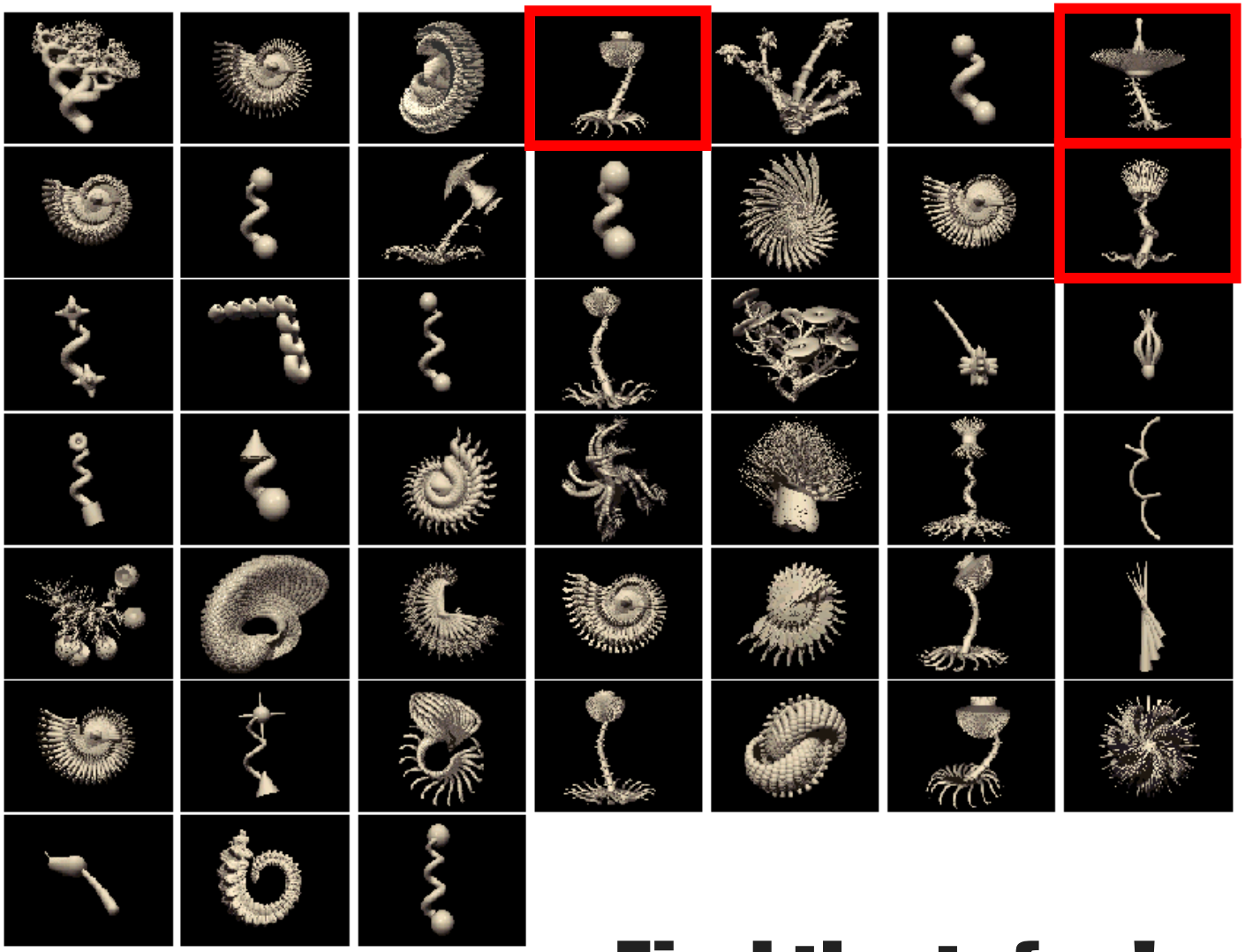
exposition ⇔ pretraining
(background knowledge)

worked problems ⇔ supervised finetuning
(problem + demonstrated solution, for imitation)



practice problems ⇔ reinforcement learning
(prompts to practice, trial & error until you reach the correct answer)

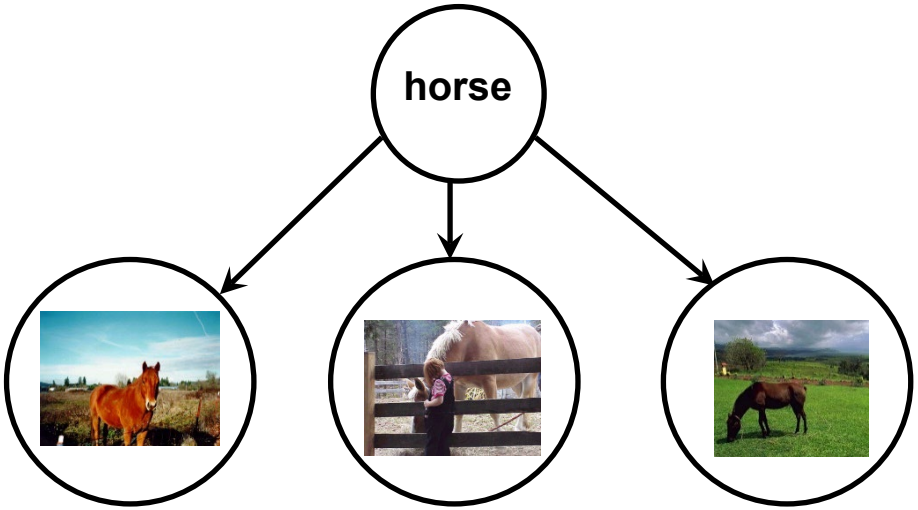
24



Find the tufas!

Why do we need math and probabilities?

- Abstraction

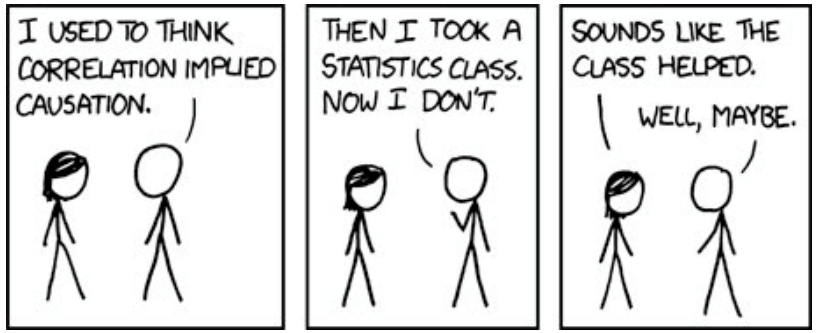


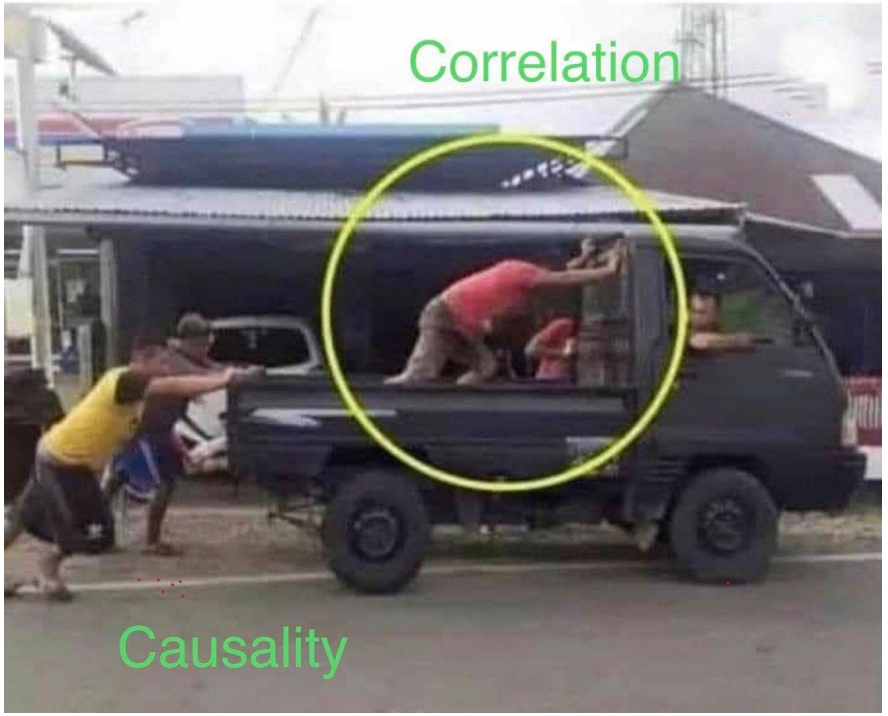
- Partial information

- Noise/distortion

Learning correlation

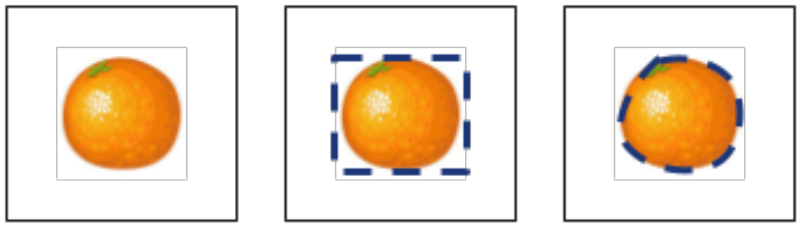
- In this course, we focus on **correlation-based learning**
- Causal learning is much harder and still under research.





Data

- Critical in any ML problem
- Have a good training dataset
  - Representative
  - Diverse
  - Well labeled (depends on the type of supervision)
- Beware of bias and discrimination
- In many cases it is the bottleneck



Orange Orange Orange

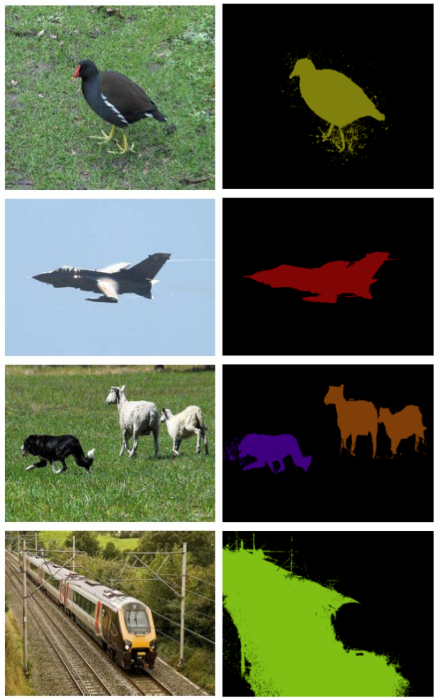
28

The fallacy of Big Data

- Tesla has 12,000 million kilometers of data (in 2016)

Data bias

Three black/white teenagers



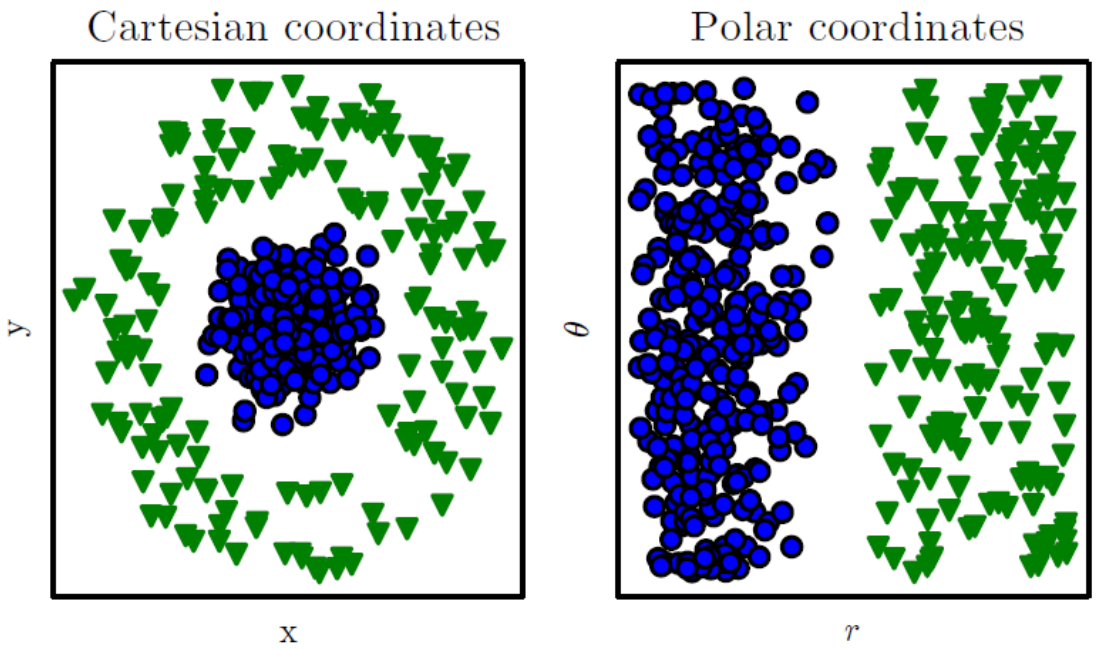
CamFind

blue and black woman's dress

men's polo shirt with blue and black stripes

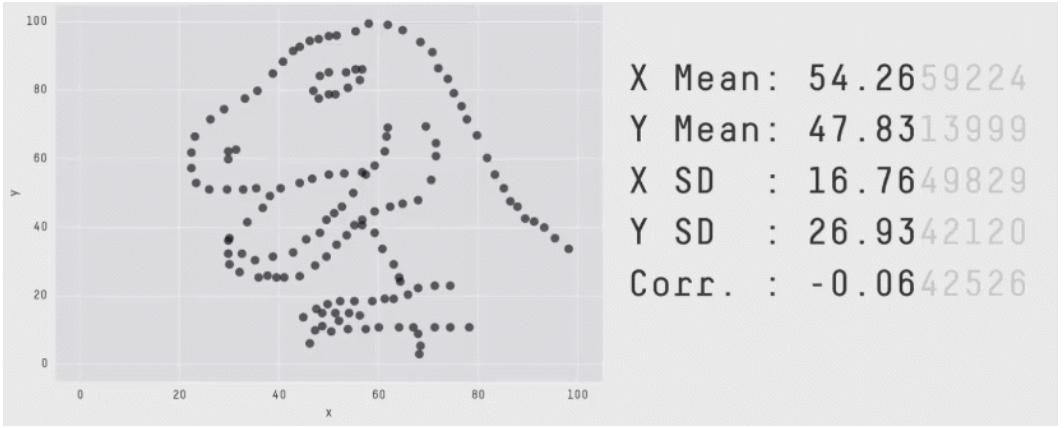
Universidad Zaragoza 1542

The problem of choosing features



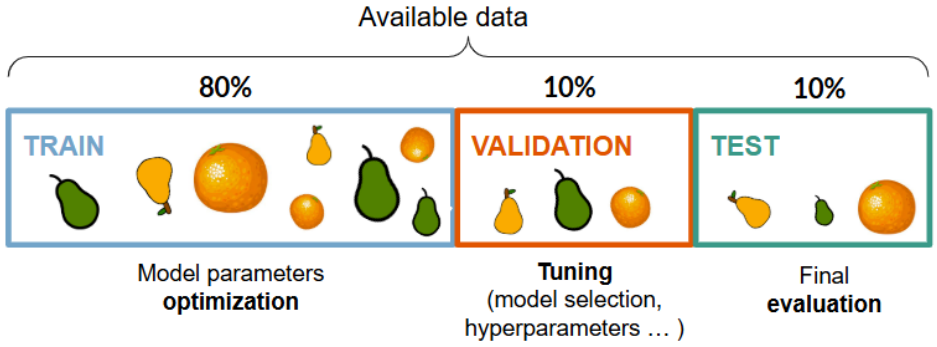
(Goodfellow et al, 2016)

(Alberto Cairo, 2016)



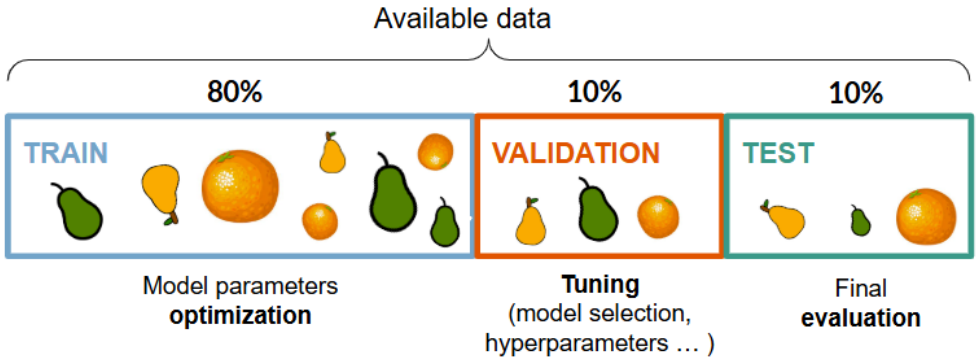
Data organization

- They are divided into three subsets
  - Training data (example ~80%)
    - Compute model parameters
  - Validation Data (example ~10%)
    - Evaluate different variations of the model
    - Tuning hyperparameters
  - Test data (example ~10%)
    - Final evaluation of the model with **unknown data**

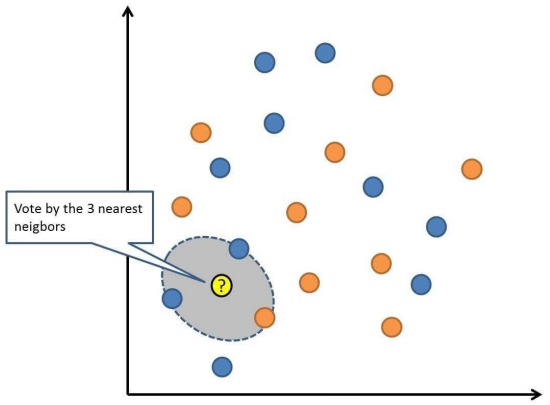


Learning process

- Training phase
  - Using the starting data X learn the model
  - Calculating Model-Defining Parameters
  - Different algorithms and hyperparameters
- Validation phase
  - Using a (different) subset within X to "measure" model quality
  - Each issue has different metrics
- Test phase
  - Evaluate the model on new unknown data



Introductory example k-NN



- k-Nearest-Neighbor (birds of a feather…)
- Supervised, classification
  - Can be extended for regression using smoothing
- Non-parametric
  - "Lazy": keeps the data instead of summarizing it into a parametric model.
  - The model complexity increase with the data
- Memory-based learning. Very simple and intuitive.
- It works reasonably well if the feature space is reasonably well sampled by the data.
- Computationally demanding at test time!

k-Nearest-Neighbor

- We have N sample pairs in our training dataset $\mathcal{D}$

$$\mathcal{D} = \left\{\left(x^{(1)}, y^{(1)}\right), \cdots, \left(x^{(i)}, y^{(i)}\right), \cdots, \left(x^{(N)}, y^{(N)}\right)\right\}$$

- 1-NN: for a query sample $x^{(q)}$, we predict the $y^{(q)}$of the closest element in our dataset $\mathcal{D}$

$$y^{(q)} = y^{(\mathrm{j})}, \qquad \mathrm{j} = \arg min_i(d(x^{(q)}, x^{(i)}))$$

  - $d(\cdot)$ distance metric (usually Euclidean, can be learnt)

- In k-NN (usually k odd), the k-nearest neighbours vote for the class. More formally:

$$p\left(y^{(q)} = c \middle| x^{(q)}, \mathcal{D}, k\right) = \frac{1}{k} \sum_{i \in N_k(x^{(q)}, \mathcal{D})} \mathbb{I}(y_i = c)$$

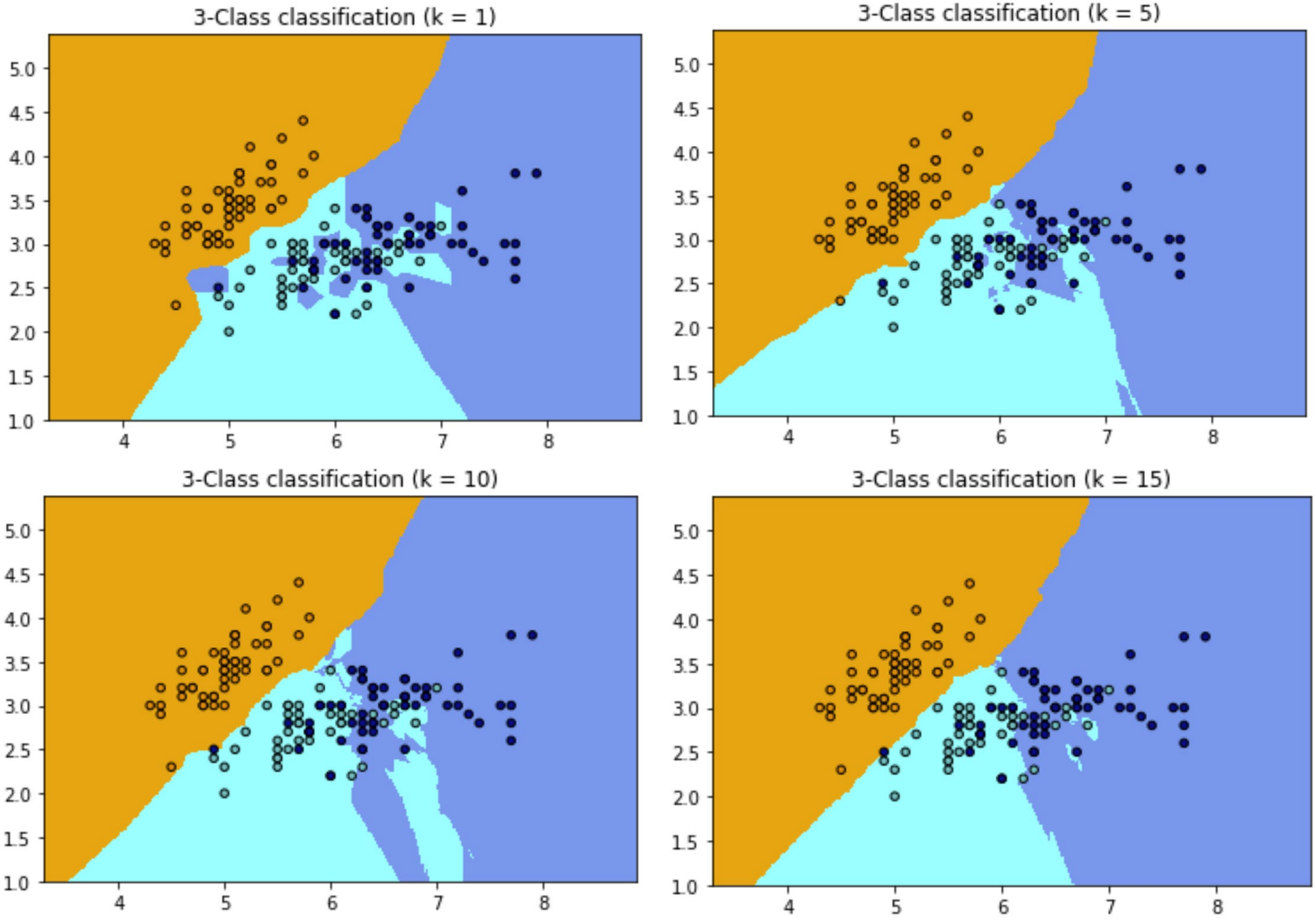
with $N_k(x^{(q)}, \mathcal{D})$ being the índices of the k nearest points and $\mathbb{I}(\cdot)$ the indicator function.

Let's start coding!

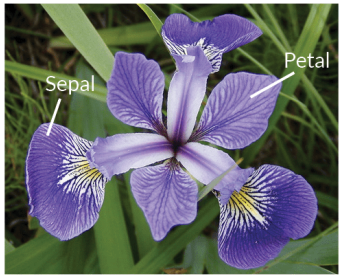
- If you are not familiar with Python, **do the tutorial** https://docs.python.org/3/tutorial/
- To make things easier at the beginning, use Google Colab https://colab.research.google.com/
- Use sklearn documentation https://scikit-learn.org/ and google for help
- The ML community shares a lot of resources, enjoy!

K-nn examples – Iris

- Iris, sepal length and width, k={1, 5, 10, 15}



Iris Setosa



Iris Versicolor

Iris Virginica

Colab link:
https://colab.research.google.com/drive/1Dyov9NO9ySNi3vSphk1JZJzlyqJoF3EG

Universidad Zaragoza 1542

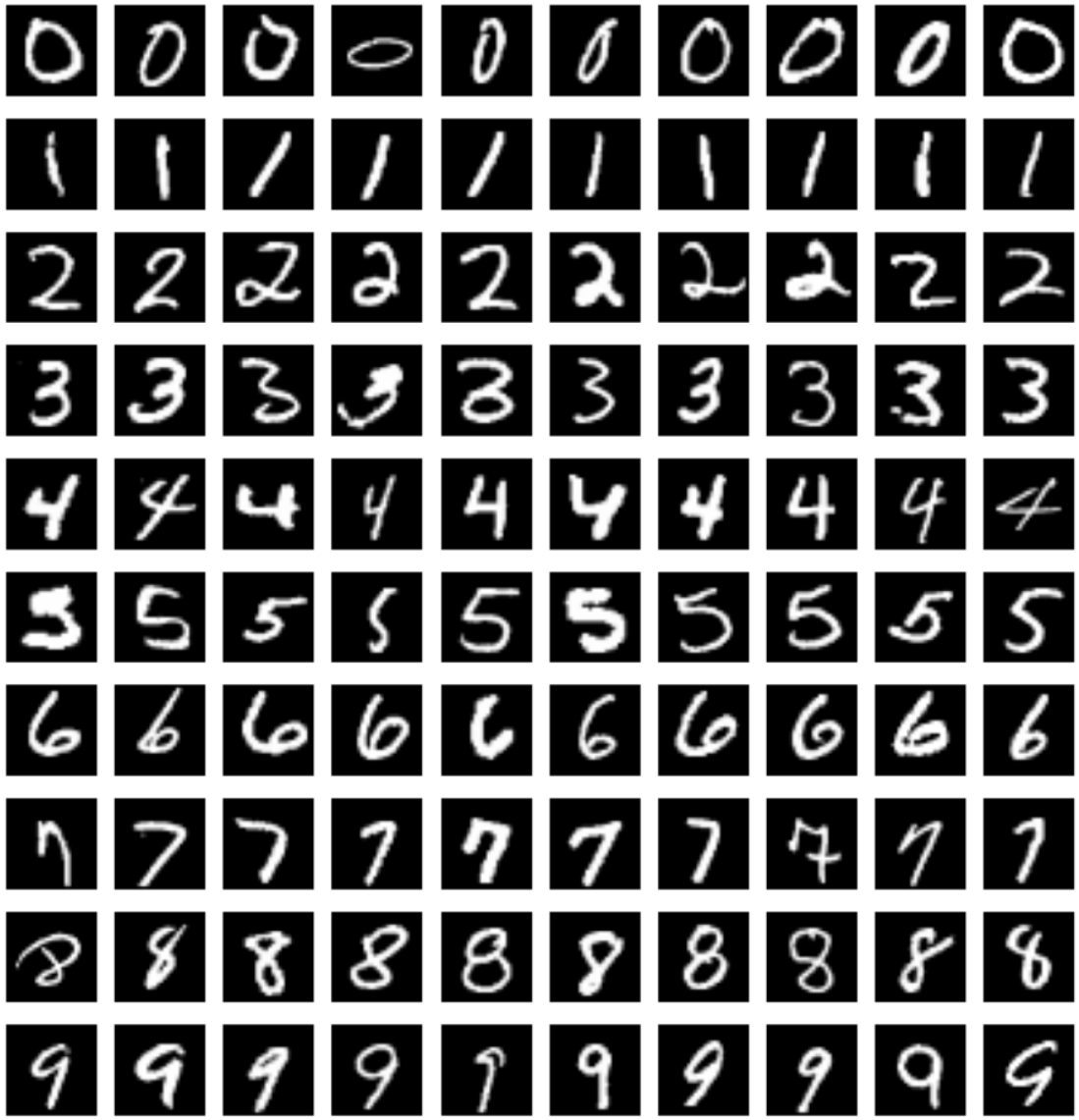
K-nn examples – MNIST

- Written digits
  - Training size: 60K

```
k=1, accuracy=97.22%
k=3, accuracy=97.11%
k=5, accuracy=96.96%
k=10, accuracy=96.55%
k=15, accuracy=96.48%
k=20, accuracy=96.18%
k=25, accuracy=95.92%
k=30, accuracy=95.75%
```

  - Training size: 5K

```
k=1, accuracy=93.61%
k=3, accuracy=93.12%
k=5, accuracy=93.08%
k=10, accuracy=92.37%
k=15, accuracy=91.89%
k=20, accuracy=91.45%
k=25, accuracy=90.93%
k=30, accuracy=90.46%
```



Colab link:
https://colab.research.google.com/drive/1EdDg6rk23BW7rO9uwhogJf5YSMsZ-zK4

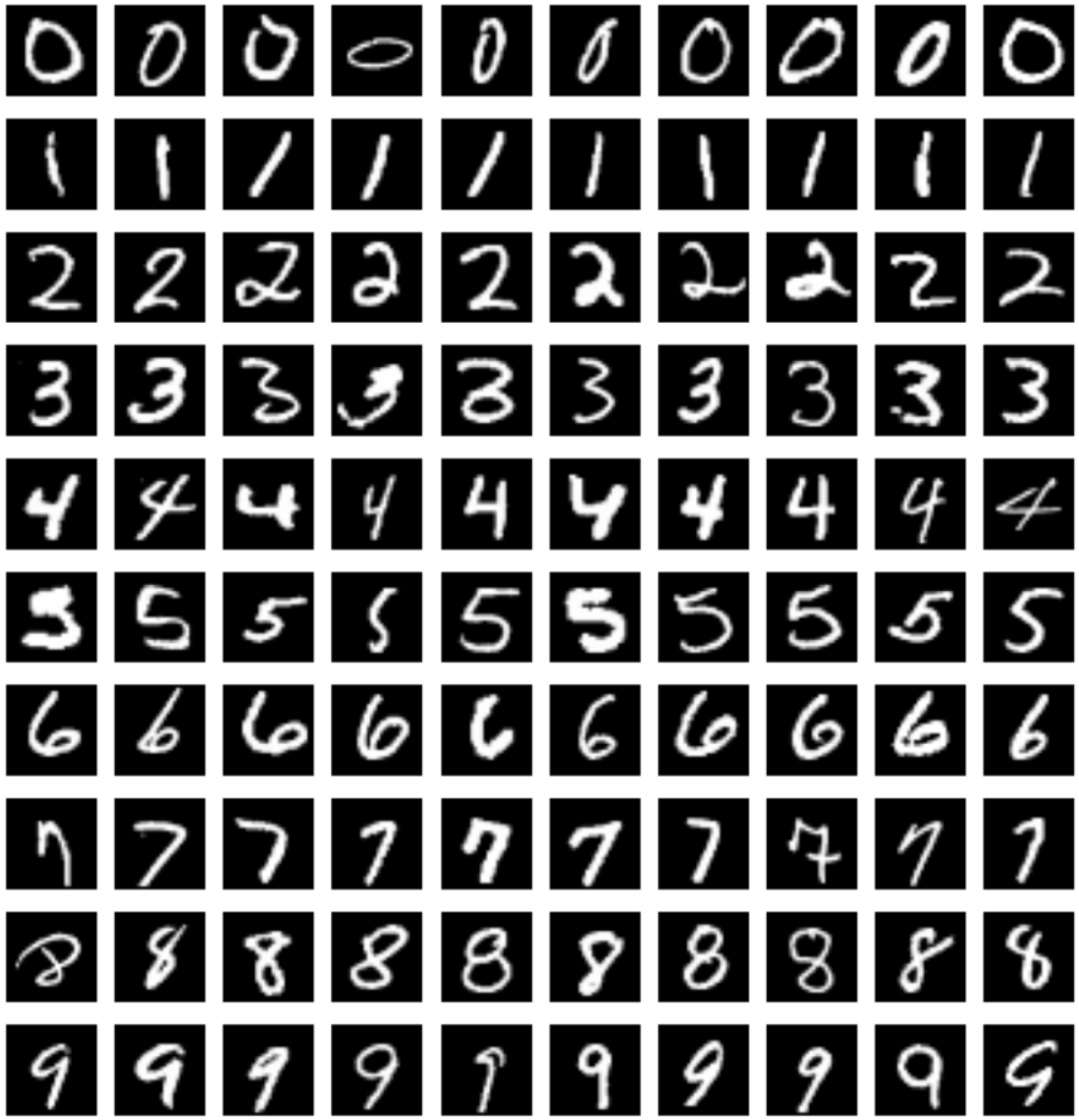
K-nn examples – MNIST

- Changing the split:
  - Training size: 60K

```
k=1, accuracy=97.16%
k=3, accuracy=97.31%
k=5, accuracy=97.22%
k=10, accuracy=96.69%
k=15, accuracy=96.60%
k=20, accuracy=96.36%
k=25, accuracy=96.04%
k=30, accuracy=95.78%
```

  - Training size: 5K

```
k=1, accuracy=93.10%
k=3, accuracy=92.98%
k=5, accuracy=92.68%
k=10, accuracy=92.14%
k=15, accuracy=91.60%
k=20, accuracy=90.90%
k=25, accuracy=90.37%
k=30, accuracy=89.97%
```

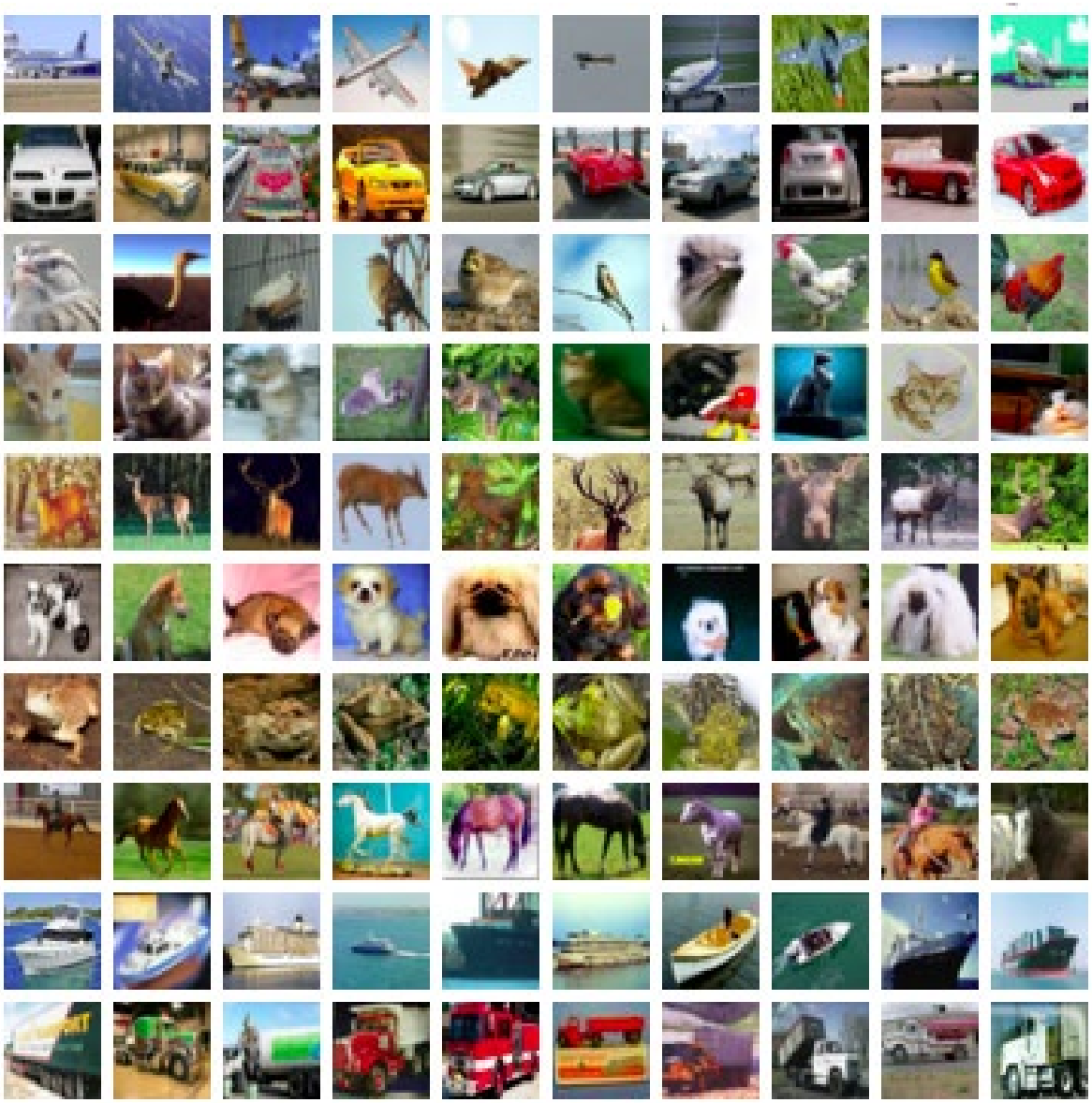


Colab link:
https://colab.research.google.com/drive/1EdDg6rk23BW7rO9uwhogJf5YSMsZ-zK4

K-nn examples – CIFAR

- For CIFAR images, things get much worse… why?
  - Training size: 50K

```
k=1, accuracy=35.39%,
k=3, accuracy=33.03%,
k=5, accuracy=33.98%,
k=7, accuracy=33.58%,
k=10, accuracy=33.86%,
k=15, accuracy=34.05%,
k=20, accuracy=33.75%,
k=25, accuracy=33.47%,
```

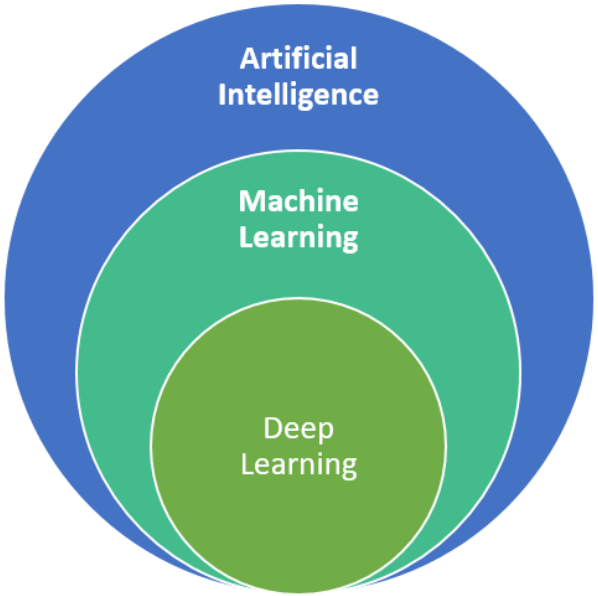


- We need other models here…

Colab link:
https://colab.research.google.com/drive/15vcuCEHcPex5Xbm_VhgcHttHqPWJzY7U

Summary

- **Machine Learning:**
  - Field of study that gives computers the ability to learn without being explicitly programmed (Arthur Samuel 1959).
  - A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E. (Tom Mitchell, 1997)
  - Success stories: Spam filtering, OCR, face detection, search engines…
- **Related terms:**
  - **Data mining:** The field of study that aims to extract information and structure from raw data.
  - **Artificial Intelligence:** the theory and development of computer systems able to perform tasks normally requiring human intelligence, such as visual perception, speech recognition, decision-making, and translation between languages.
  - **Knowledge Discovery**, **Pattern Recognition**, …



Summary (II)

- **Training set:** Data samples that machine learning algorithms use to find the model parameters.
- **Test set:** Data used to assess generalization over samples that are unseen during training.
- **Supervised Learning:** The training set is composed of **labeled data** (data + desired output)
- **Unsupervised Learning:** The training data is **unlabeled**.
- **Regression:** Estimates a function that relates an input vector with a **continuous output**.
- **Classification**: Estimates a function that relates an input vector with a **discrete output**.
- **Parametric:** The model has a fixed number of parameters (by making assumptions on the data distribution)
- **Non-parametric:** The number of parameters grow with the data

Methodology (I of III)

1. Frame the Problem (Look at the Big Picture)

- Define the general high-level objective
- Frame the problem (supervised/unsupervised, regression/classification, etc.)
- Define metrics to evaluate performance (e.g., error, accuracy, false positives rate, etc.)
- What is a minimum/reasonable/maximum performance? Check scientific literature and existing similar systems.

2. Get the data

- What data is needed? How much? Which format?
- How/where do you get the data? Check legal aspects and sensitive information!
- Divide into training/validation/test sets. Never look at the test set!
- The unreasonable effectiveness of data (size and quality!) → https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/35179.pdf

Methodology (II of III)

3. Explore the data

- Visualize the data
- Study correlations between atributes

4. Prepare the data

- Work in a copy of the data, keep the original data intact
- Data cleaning: Fix/remove outliers, fill missing values / remove rows
- Feature selection: Drop uninformative attributes (garbage in – garbage out)
- Feature engineering: Transform the data into more descriptive/separable features (e.g., #murders vs. #murders/population)
- Feature scaling: Standardize or normalize features/data
- Shuffle the data

Methodology (III of III)

5. Shortlist promising models

- Sample subsets of the training sets for quick exploration of many techniques/models
- Compare models using cross-validation (mean and std over folds)
- Analyze errors

6. Fine-tune the system

- Fine-tune hyperparameters using cross-validation
- Try ensemble methods
- Once finished, evaluate the performance on the test set

7. Present the solution

- Report relevant aspects of the process, not only performance
- Failure cases might be relevant!
- Report assumptions, limitations, how the model could be improved
- Summarize (graphs, key findings)

Universidad Zaragoza

Machine Learning (69152)

Máster in Robotics, Graphics and Computer Vision