Data-intensive space engineering

Lecture 1

Carlos Sanmiguel Vila

Instructor

Lecturer: INTA – Dr. Carlos Sanmiguel Vila

Short Bio:

- **Education**: Aerospace Eng. MSc'01 & BSc'14 (UPM, SP),
  (10yrs) Fluid Mechanics. PhD'19 (UC3M, USA)
- **Academic Experience**: 1yrs Postdoc (UC3M, SP),
  (3yrs) 2yrs Adjunct Professor (UC3M, SP)
- **Industrial Experience**: 2yrs Data Scientist/Analyst (Santander, UK),
  (5yrs) 3yrs Research Scientist (INTA, SP)
- **Expertise & projects**: **Artificial intelligence in the aerospace industry,** applications in Surrogate Model for Aerodynamics (aircraft), flow control, reduced-order modelling, multifidelity and super resolution algorithms.

Author of over 20 publications (peer-reviewed journals and conferences). **Research Team and/or Principal Investigator of 20 projects,** in aeronautics (TIFON as PI).

Mail: csanmigu@ing.uc3m.es

Course Information

- **Course Aim:** Explore statistical and artificial intelligence techniques for the analysis of space-engineering data via:
- Theoretical sessions: Lectures will introduce key concepts, theories, and the latest developments in data-intensive techniques, laying a solid foundation for students
- Practical sessions: hands-on laboratory sessions where students will apply the discussed machine learning techniques to real-world space engineering problems using computer-based tools

Course Information

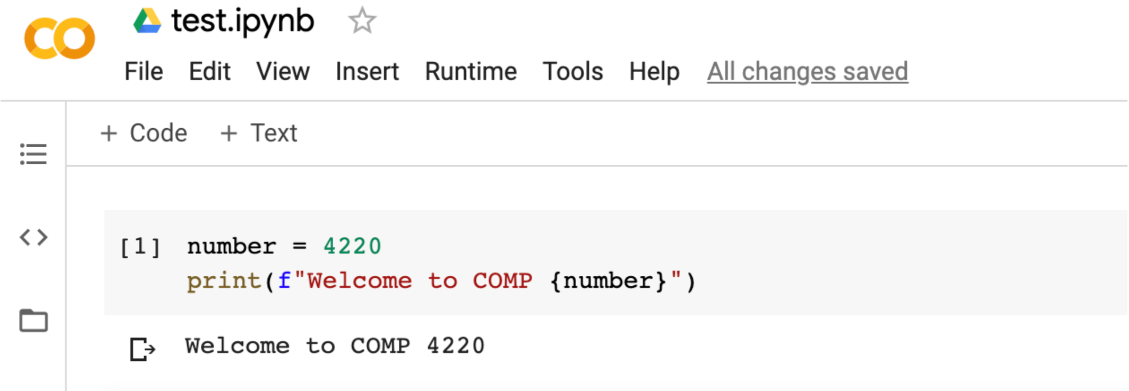
Practical sessions:

- Python will be the programming language during the course.
- Labs will be Google Colab notebooks, which will count as 75% of the final grade.
- End-of-term-examination will be 25% of the final grade.

Google Colab

Google Colaboratory, or “Colab” for short, allows you to write and execute Python in your browser

- Zero configuration required
- Free access to GPUs and other computing resources
- Use terminal commands on Google Colab
- https://drive.google.com (for the first time, select “Connect more apps”)



```
number = 4220
print(f"Welcome to COMP {number}")
```

```
Welcome to COMP 4220
```

https://www.youtube.com/watch?v=RLYoEyIHL6A

Course Information

To pass the course, the following two requirements must be met:

- Obtain a MINIMUM of 4.0/10 in the final exam
- Obtain a MINIMUM of 5.0/10 in the overall grade (obtained weighting 25% of the final exam and 75% of the continuous evaluation)

The continuous evaluation includes 5 laboratory sessions with corresponding reports (each corresponding to 15% of the final grade)

Continuous evaluation

Continuous evaluation will consist of:

- Complete a Google Colab Python notebook proposed during laboratory class.
- Prepare a presentation of 5 minutes about a scientific paper related to the class topics.
- 2 students will form groups.

Continuous evaluation

The presentations will consist of a few slides, where the students will look for a practical application of machine learning in the aerospace industry.

Students must ask the following questions:

- What is the problem that is investigated?
- Which is the dataset employed? Describe it briefly.
- What are the main results obtained?

Continuous evaluation

To look for scientific articles, students can use:

- https://scholar.google.es/ (Use UC3M VPN)
- https://www.perplexity.ai/
- https://elicit.com/

Take care using ChatGPT since it uses fake references and made-up studies.

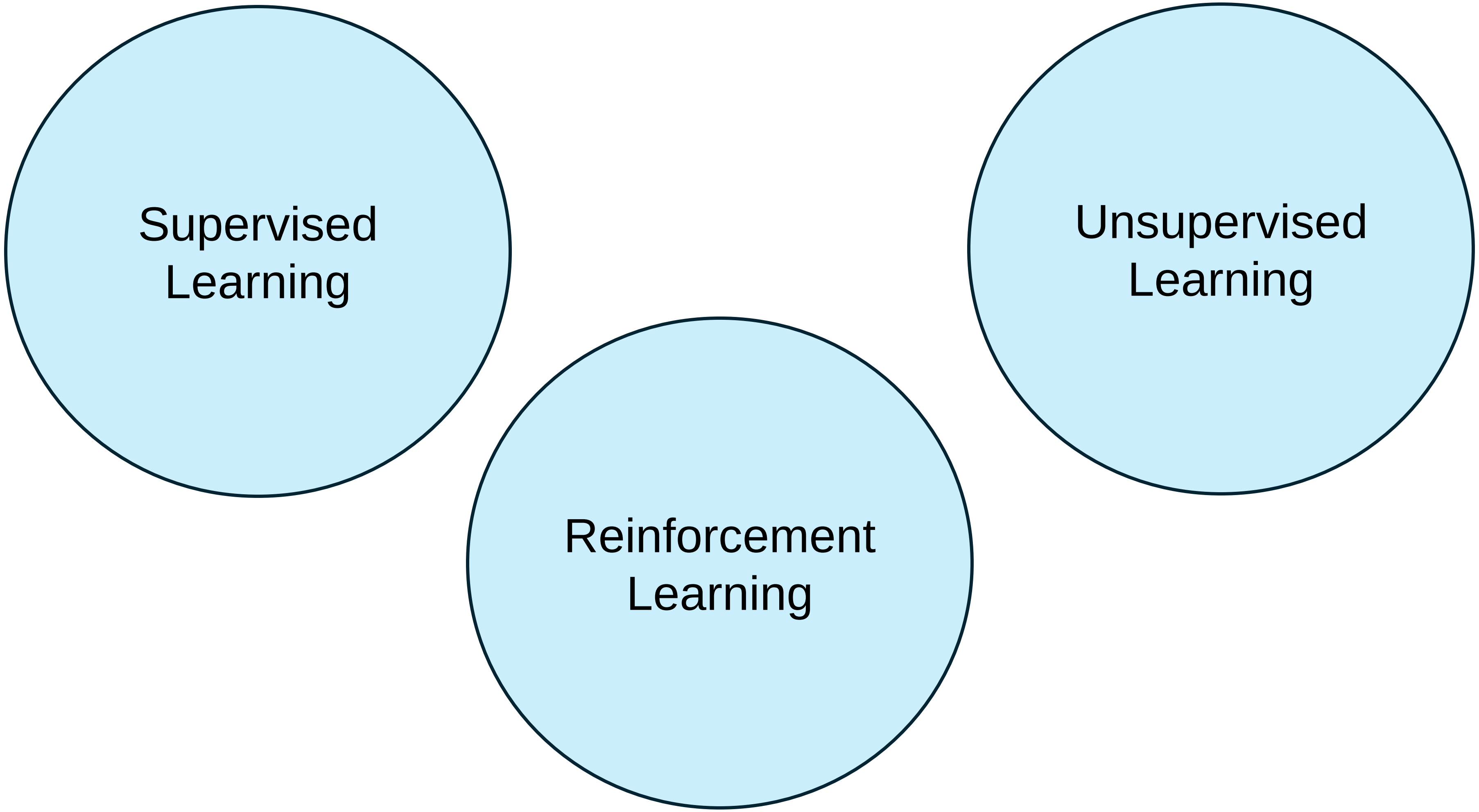
Students must send me the selected paper in advance to avoid coincidences between groups. The articles will be selected on a first-come, first-served basis; if you have troubles, you can ask for help.

Definition of Machine Learning

- Arthur Samuel (1959): Machine Learning is the field of study that gives the computer the ability to learn without being explicitly programmed.
- Tom Mitchell (1998): a computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.

Experience (data): games played by the program (with itself) Performance measure: winning rate

Taxonomy of Machine Learning
(A Simplistic View Based on Tasks)

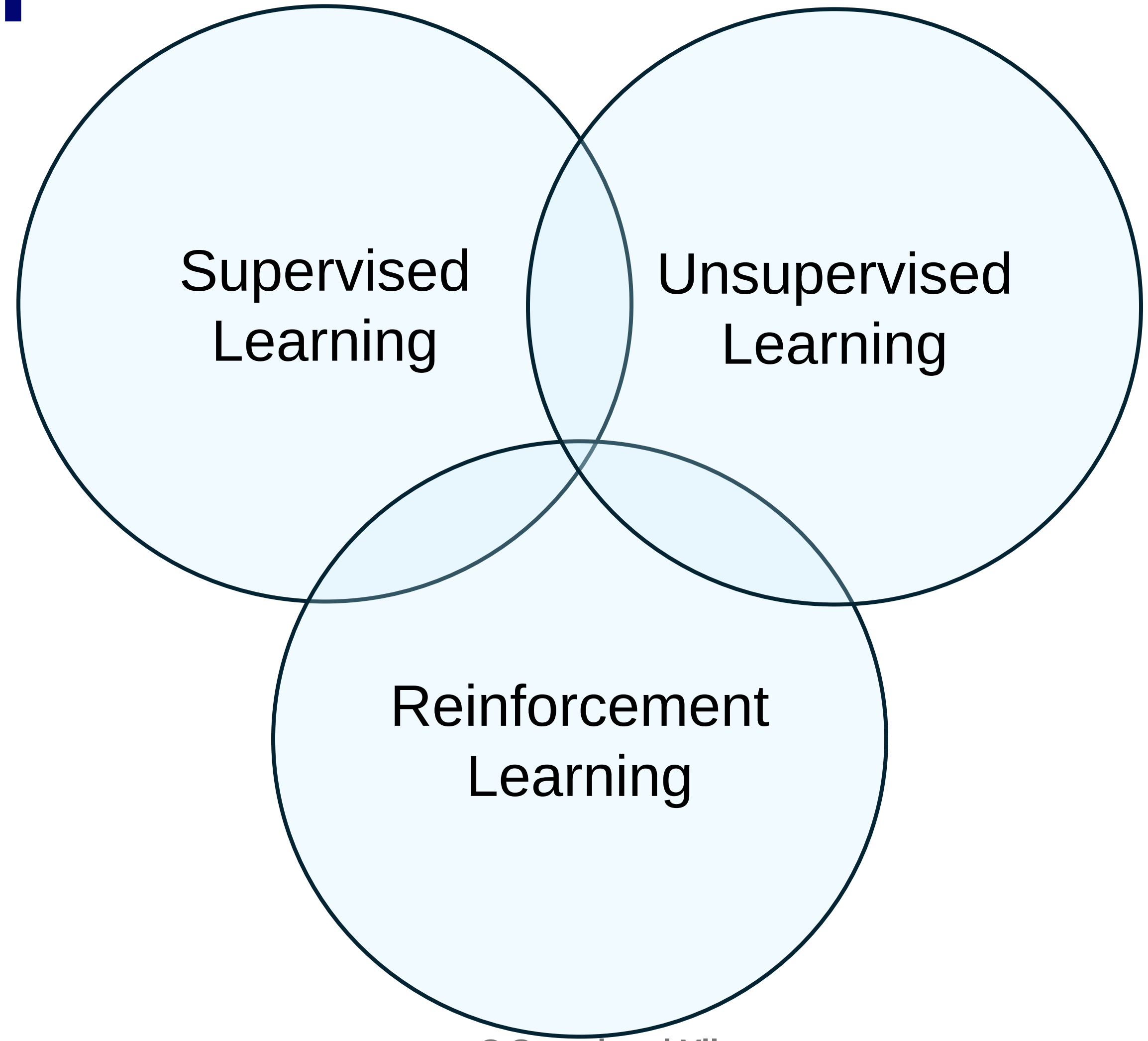


Supervised Learning

Unsupervised Learning

Reinforcement Learning

Taxonomy of Machine Learning
(A Simplistic View Based on Tasks)



Supervised Learning

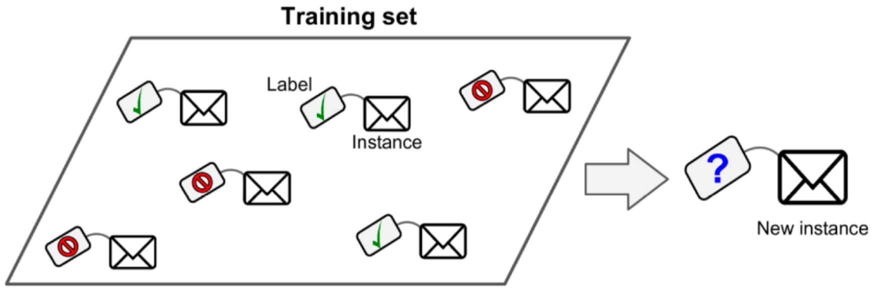
Unsupervised Learning

Reinforcement Learning

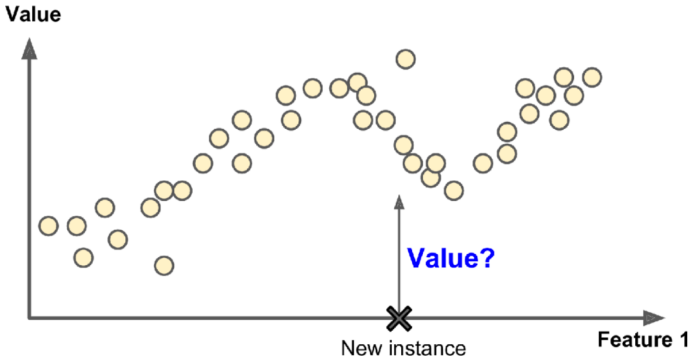
Supervised Learning

Supervised learning: training set includes the desired solutions or labels

- **Classification**: labels are classes (e.g., spam or ham)



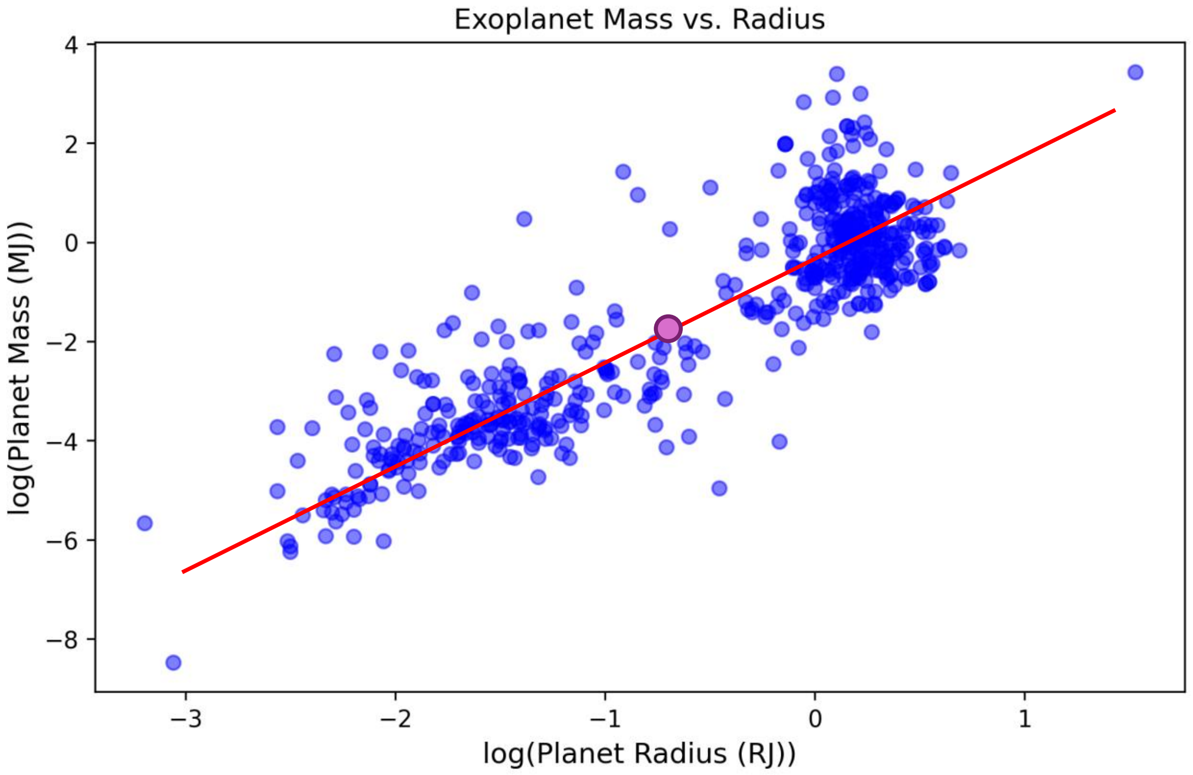
- **Regression**: labels are numeric values (e.g., price of car)



Supervised Learning - Regression

1. **Exoplanet Detection and Characterization:**
   Regression models are used to analyze light curves from stars to detect and characterize exoplanets.
2. **Solar Flare Prediction:**
   Regression models are employed to predict the intensity and duration of solar flares based on solar activity data.
3. **Space Weather Forecasting:**
   Regression techniques are used to predict various space weather phenomena, such as geomagnetic storms.
4. **Satellite Orbit Prediction:**
   Regression models are used to predict satellite orbits and potential collisions with space debris.
5. **Spectral Analysis of Astronomical Objects:**
   Regression models are applied to analyze spectral data from stars, galaxies, and other celestial objects.

Supervised Learning - Regression



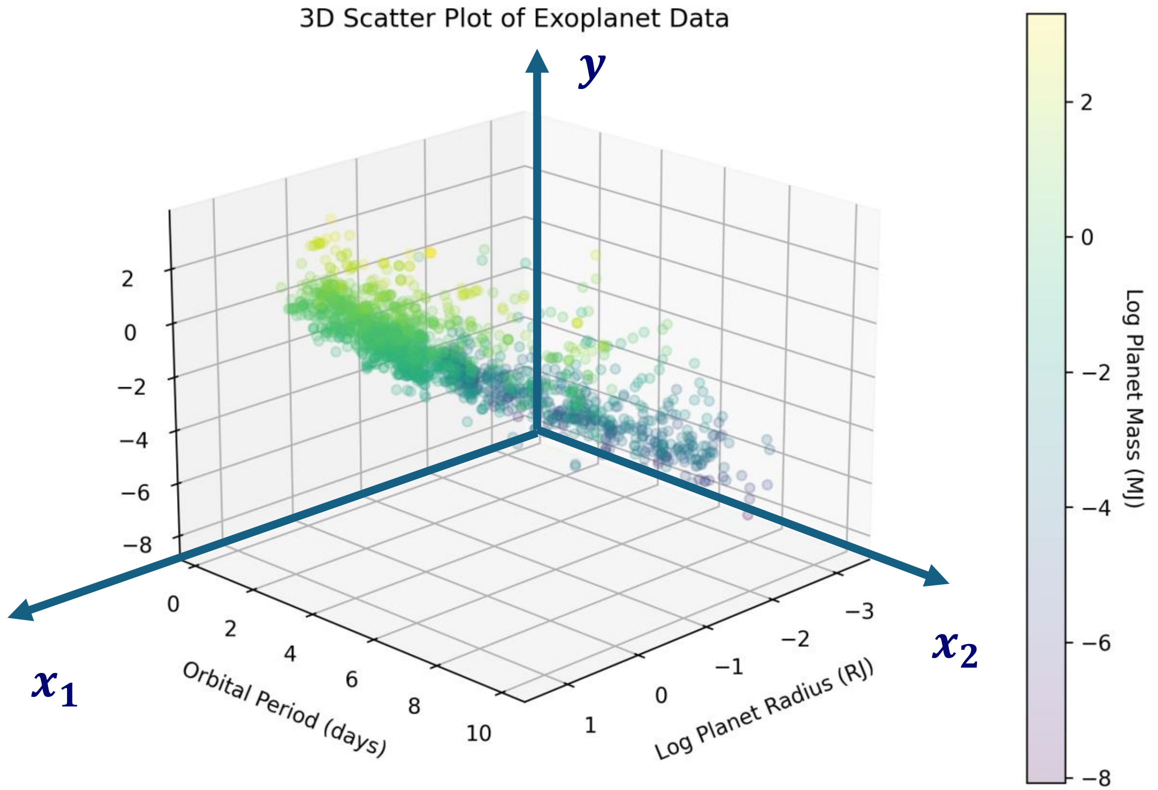
Data from NASA Exoplanet Archive

- Given: a dataset that contains n samples

$$\left(x^{(1)}, y^{(1)}\right), \dots \left(x^{(n)}, y^{(n)}\right)$$

Task: if a planet has x radius, predict its mass?

Supervised Learning - Regression



Data from NASA Exoplanet Archive

- Suppose we also know other information
- **Task**: find a function that maps

(orbital period, Planet radius) → (Planet Mass)

features/input $x \in \mathbb{R}^2$

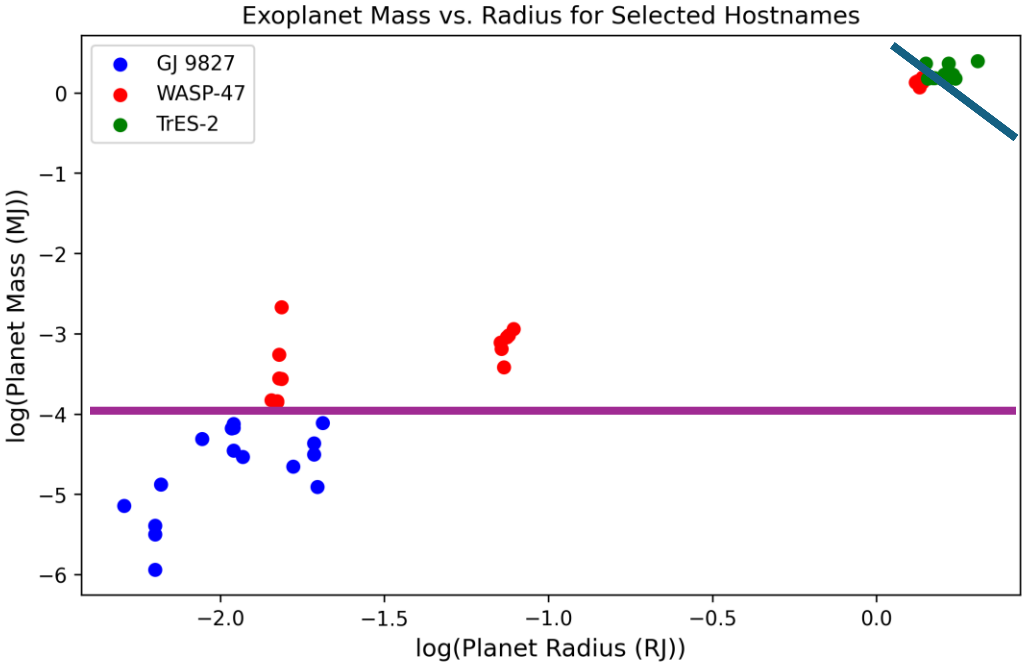
label/output $y \in \mathbb{R}$

- Dataset: $\left(x^{(1)}, y^{(1)}\right), \ldots \left(x^{(n)}, y^{(n)}\right)$ where in general we can have a very large number of features $x^{(i)} = \left(x_1^{(i)}, x_2^{(i)}, \ldots, x_d^{(i)}\right)$ with $x \in \mathbb{R}^d$
- Supervision refers to $y^1, \ldots, y^{(n)}$

Supervised Learning - Classification

1. **Space Object Identification and Classification:** Extract information from the hyperspectral signatures of unknown space objects. It involves using machine learning techniques to decompose spectra and identify materials, followed by probabilistic classification of space objects based on material identification. Neural Networks(NNs) can be used to decompose and classify satellites into categories such as communication satellites, rocket bodies, and Earth observation satellites.
2. **Exoplanet Detection:** Classification models are used to identify potential exoplanets from light curve data collected by space telescopes like Kepler. These models classify whether a dip in light intensity is due to an exoplanet transit or other phenomena.
3. **Astronomical Object Classification:** Classifying celestial objects such as stars, galaxies, and quasars based on their spectral data. This helps in organizing large astronomical datasets and identifying new types of objects.

Regression vs Classification



Data from NASA Exoplanet Archive

- **Regression**: if $y \in \mathbb{R}$ is a continuous variable, e.g., Planet mass

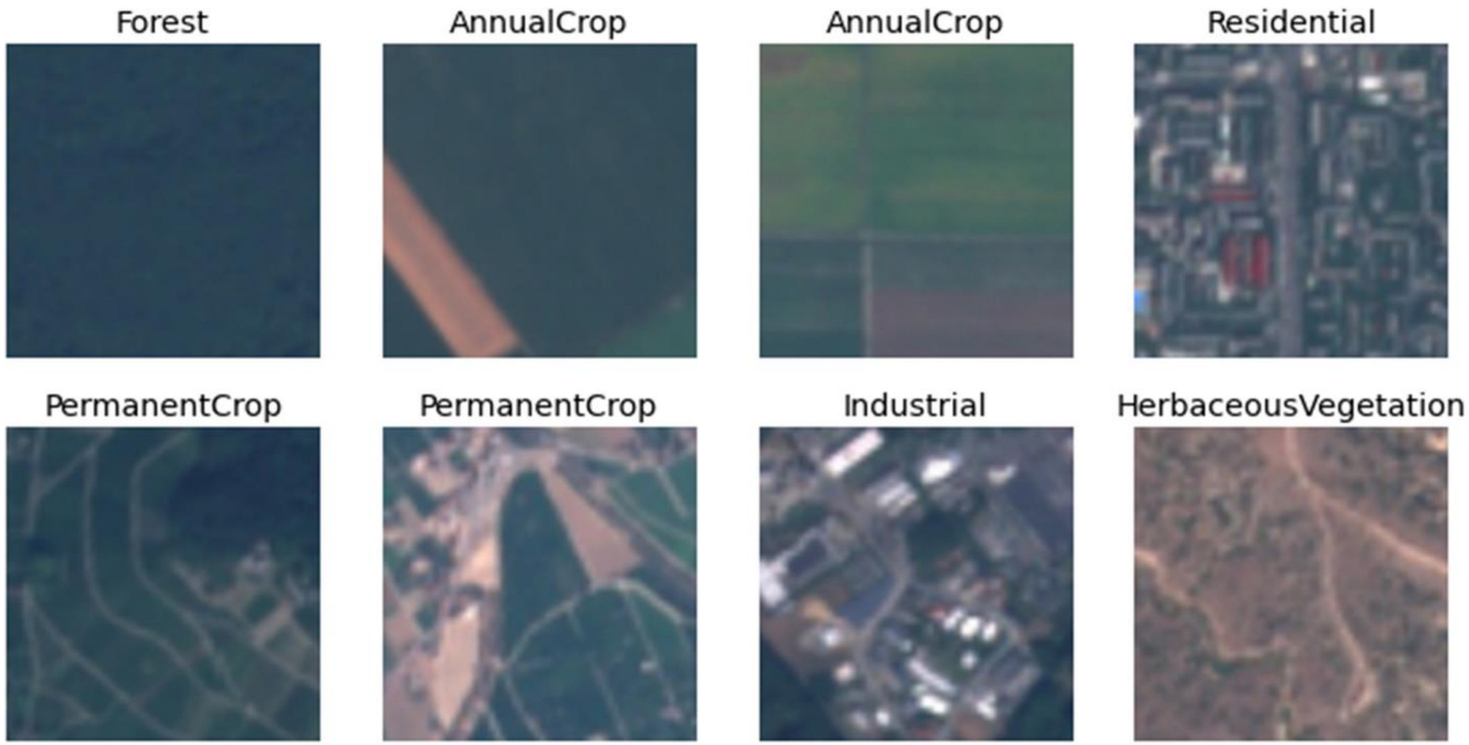
- **Classification**: the label is a discrete variable

e.g., the task of predicting which is their host star

(Planet Mass, Planet radius) → (Host star)

Supervised Learning in Computer Vision

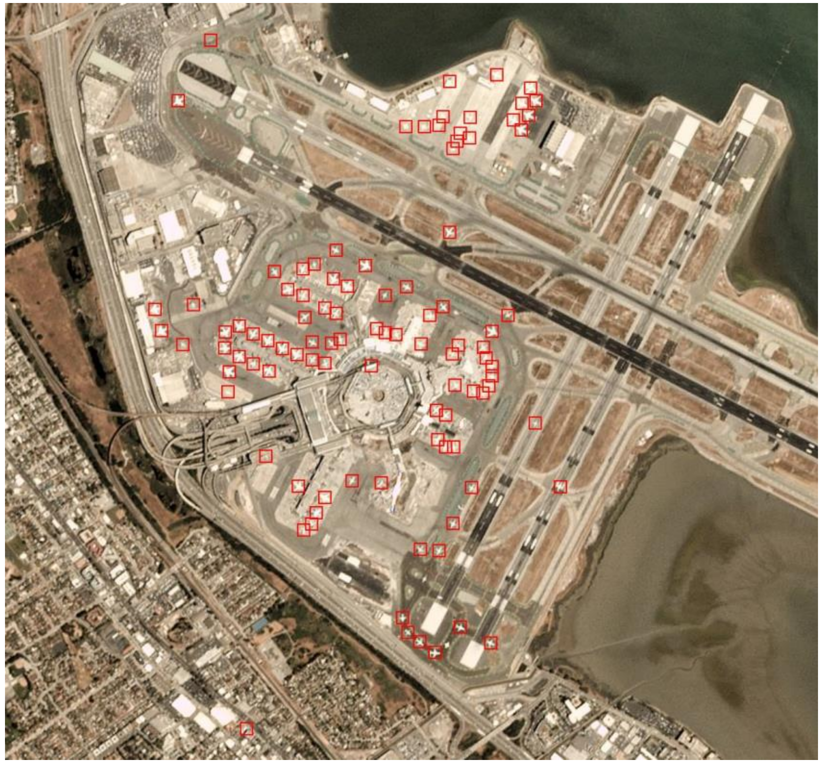
- Image Classification
- $x =$ raw pixels of the image, $y =$ the main object



Sample images from EuroSAT dataset

Supervised Learning in Computer Vision

- Object localization and detection
- x = raw pixels of the image, y = the bounding boxes

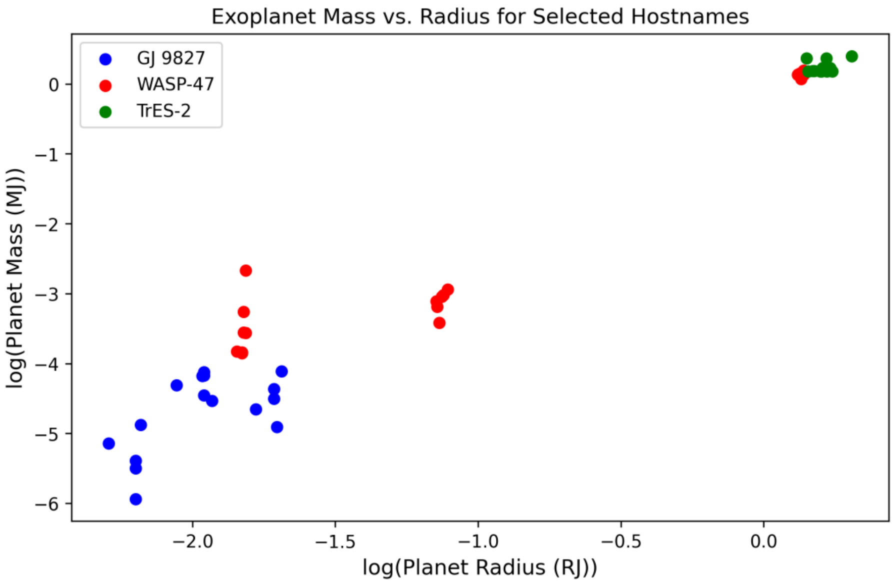


Sample images from the PlanesNet dataset

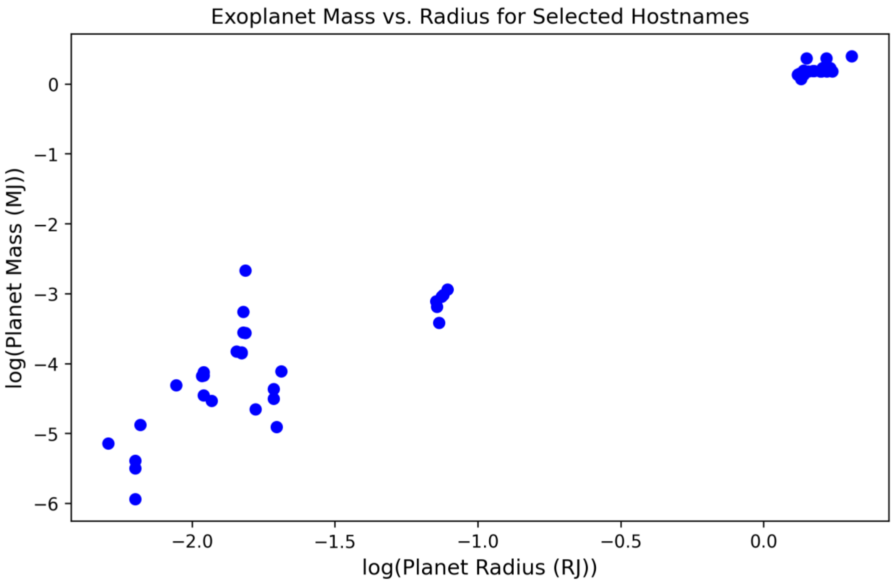
Unsupervised Learning

- Dataset contains no labels: $\left(x^{(1)}, \ldots, x^{(n)}\right)$
- Goal (vaguely-posed): to find interesting structures in the data

Supervised



Unsupervised

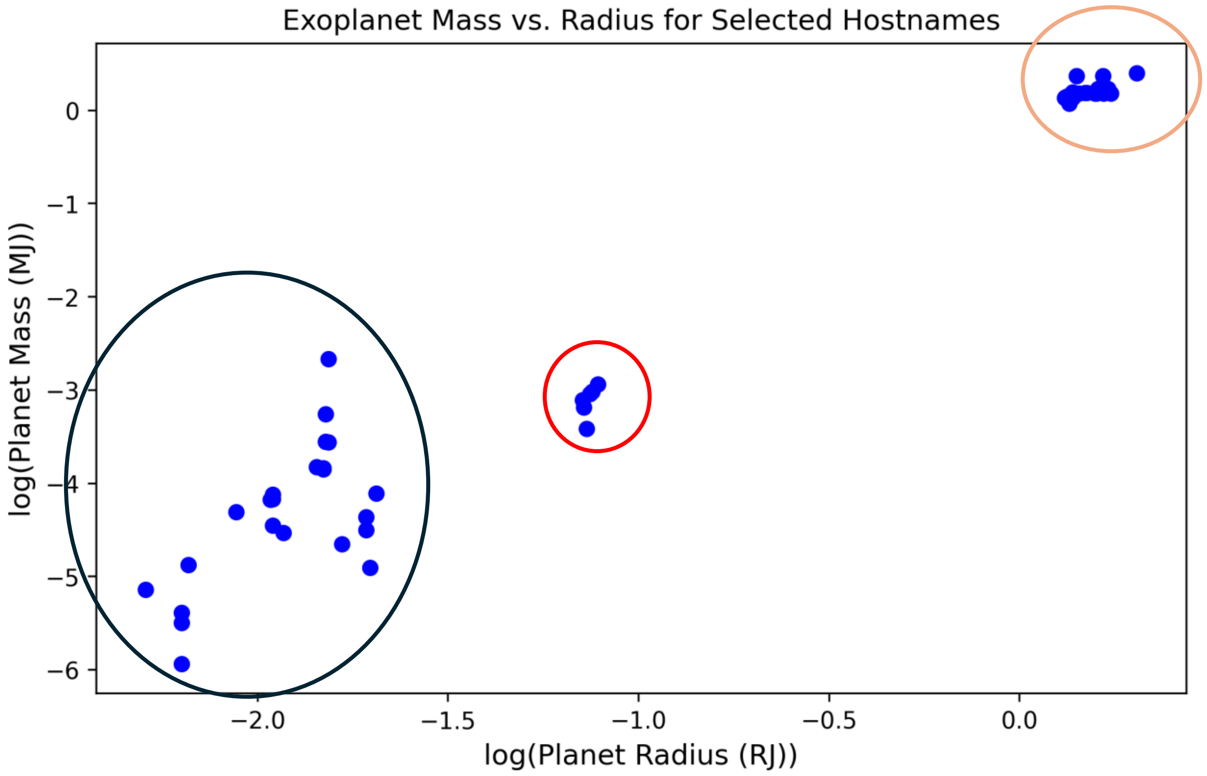


Unsupervised Learning

- **Astronomical Object Classification:** Unsupervised learning techniques are used to classify celestial objects like stars, galaxies, and quasars based on their spectral data
- **Galaxy Morphology Classification:** Unsupervised methods are applied to classify galaxies based on their morphology without relying on predefined categories
- **Exoplanet Detection:** Unsupervised learning is used to analyze light curve data from space telescopes to detect potential exoplanets
- **Space Object Identification:** Clustering techniques are used to identify and classify space objects based on their spectral signatures
- **Solar Wind Classification:** Unsupervised learning is applied to classify different regions of the Earth's magnetosphere based on solar wind parameters

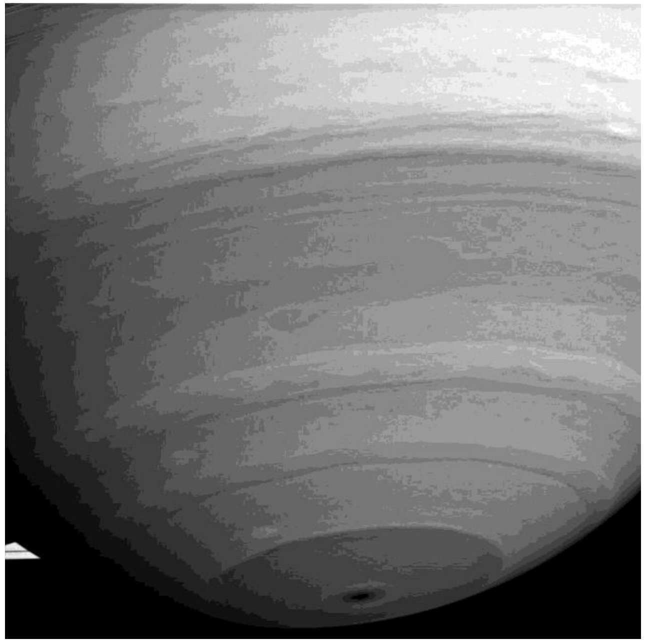
Unsupervised Learning - Clustering

Clustering: Detect groups of similar data



Unsupervised Learning – Dimensionality Reduction

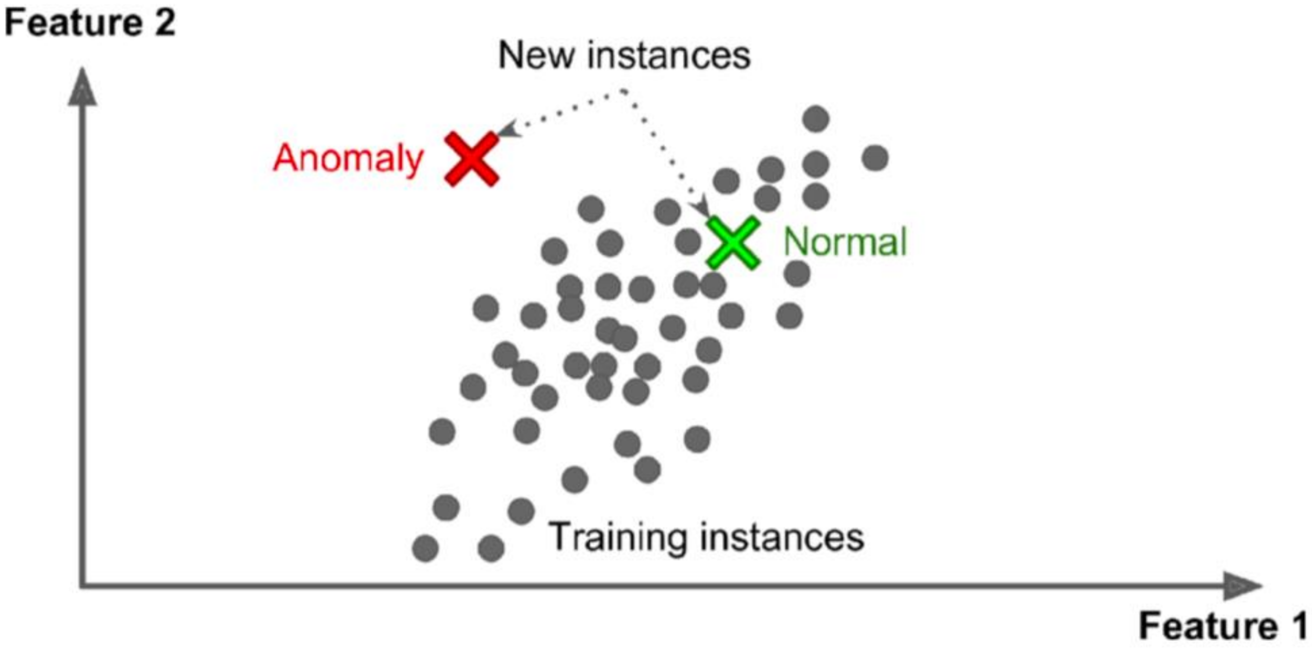
Dimensionality Reduction: Reduce the dimension of the data, i.e., image compression



https://science.nasa.gov/resource/image-compression/

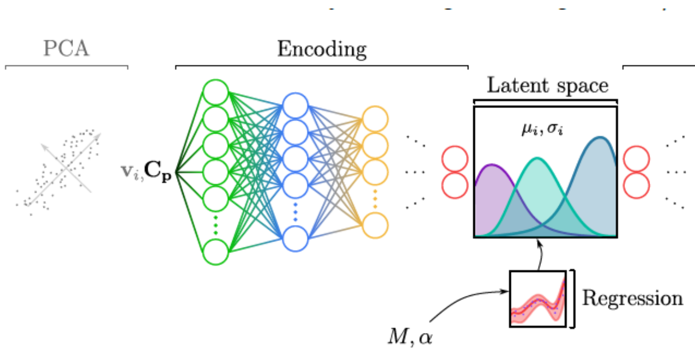
Unsupervised Learning – Dimensionality Reduction

Anomaly detection: find new instances that look different



Supervised & Unsupervised Learning

Reduce dimensionality to perform a more efficient regression in a low-dimensional space


PCA
Encoding
Latent space
Decoding
PCA^{-1}
μ_i, σ_i
$\mathbf{v}_i, \mathbf{C_p}$
$\tilde{\mathbf{C}}_\mathbf{p}, \tilde{\mathbf{v}}_i$
Regression
M, α

FRANCÉS-BELDA, Víctor, et al. Towards aerodynamic surrogate modeling based on β-variational autoencoders. arXiv preprint arXiv:2408.04969, 2024.

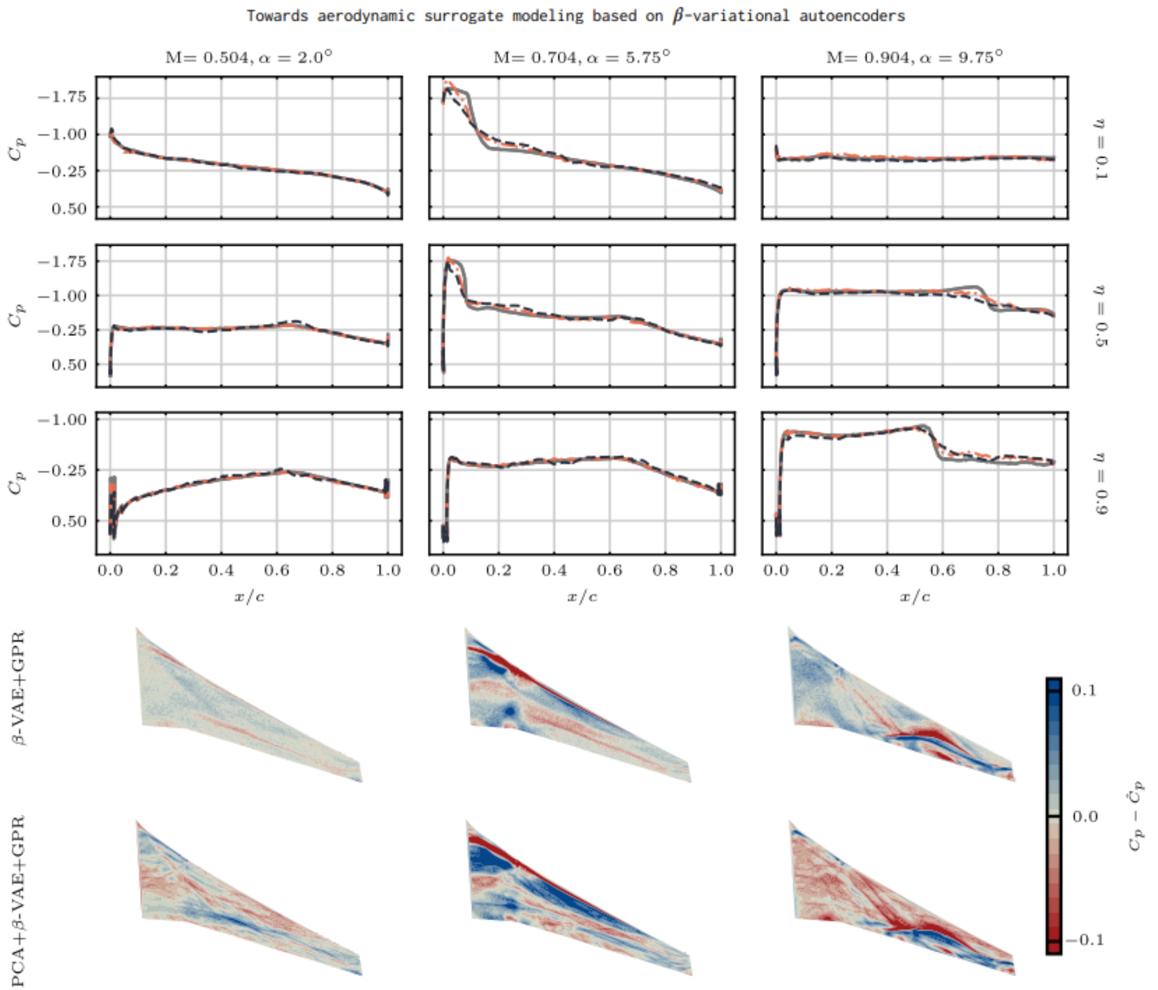
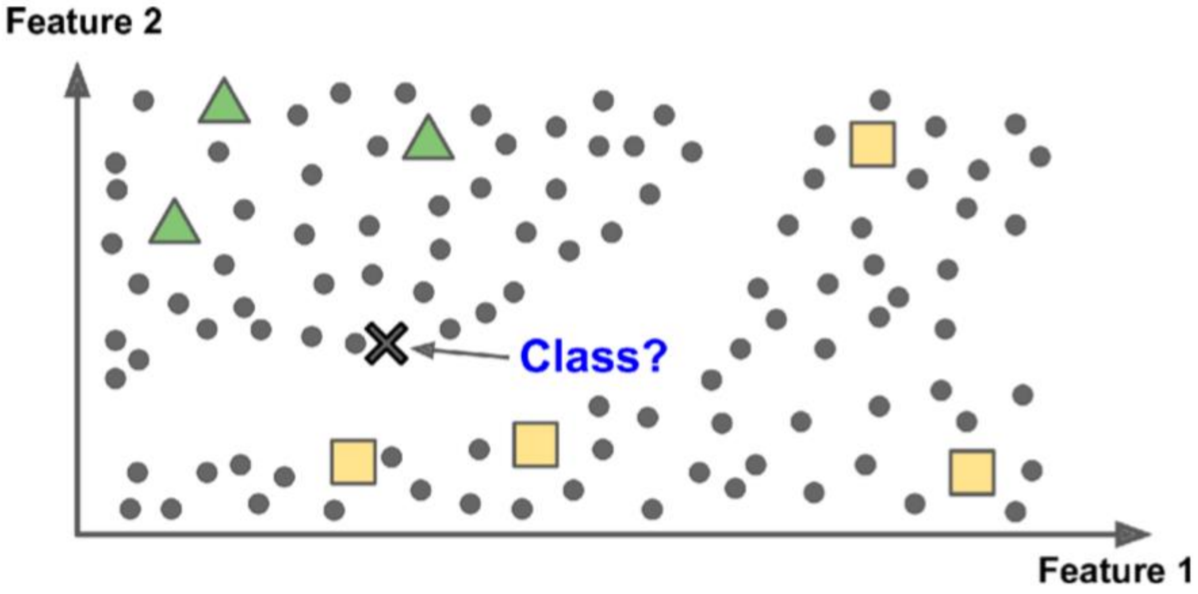


Figure 12: Difference between ground truth and predicted coefficients for the visualization test cases with $\beta = 0.008$. Chordwise pressure distributions from fine-tuned β-VAE+GPR (−·−) and PCA+β-VAE+GPR (−−) models at span percentages $\eta = 0.1$, 0.5, 0.9 are displayed and contrasted with ground truth (—).

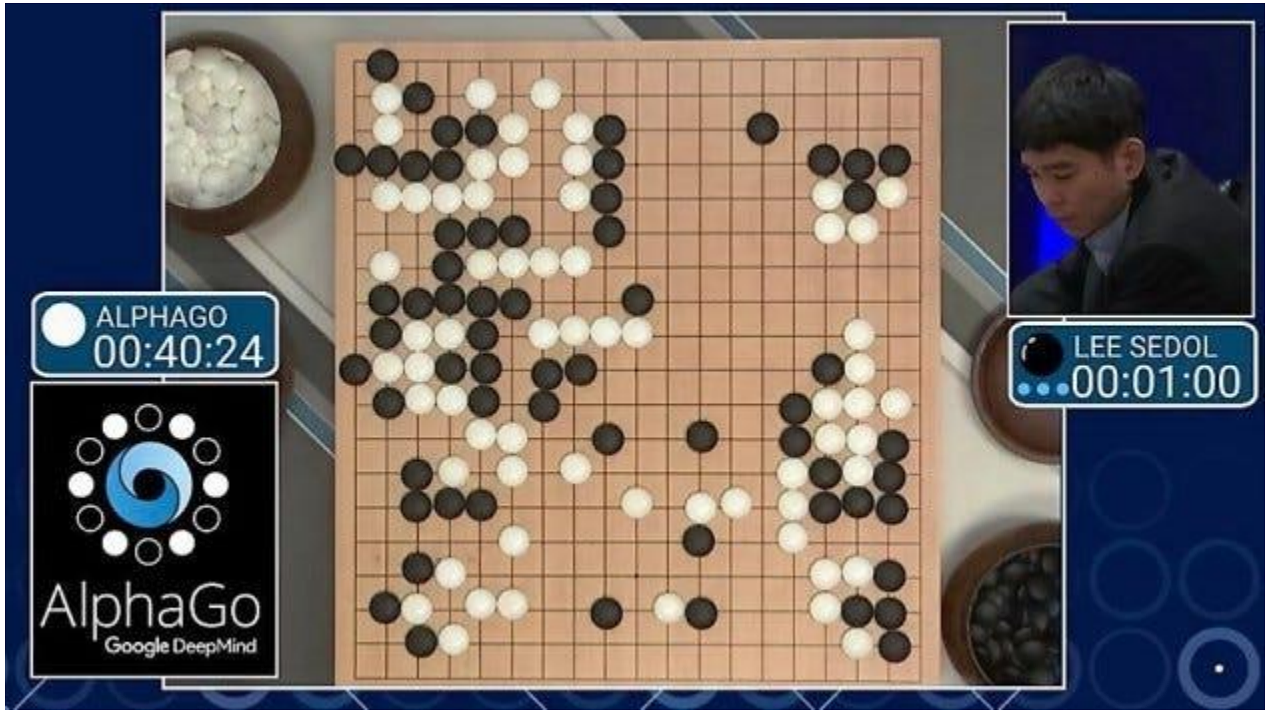
Semisupervised Learning

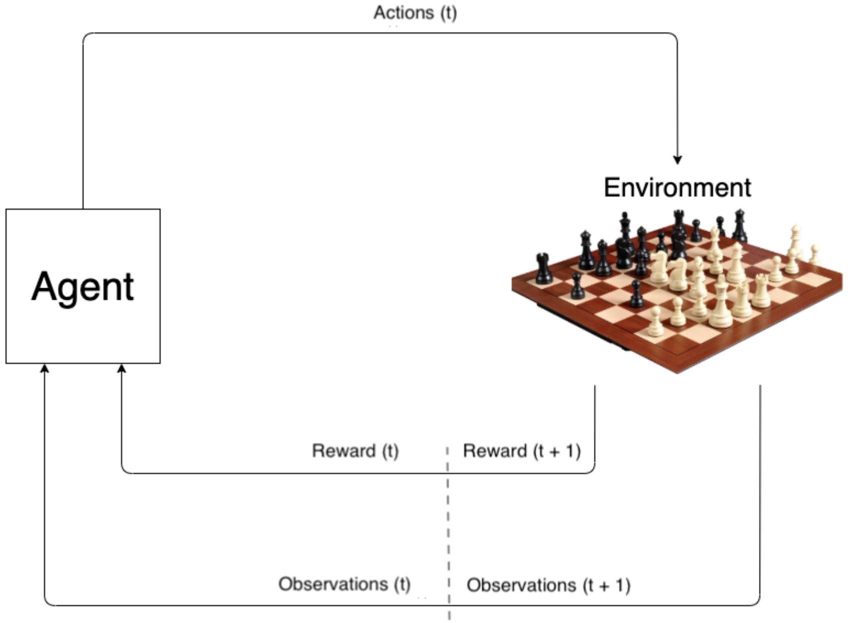
Semisupervised learning: plenty of unlabeled instances and few labeled



Reinforcement Learning

Learning to make sequential decisions.

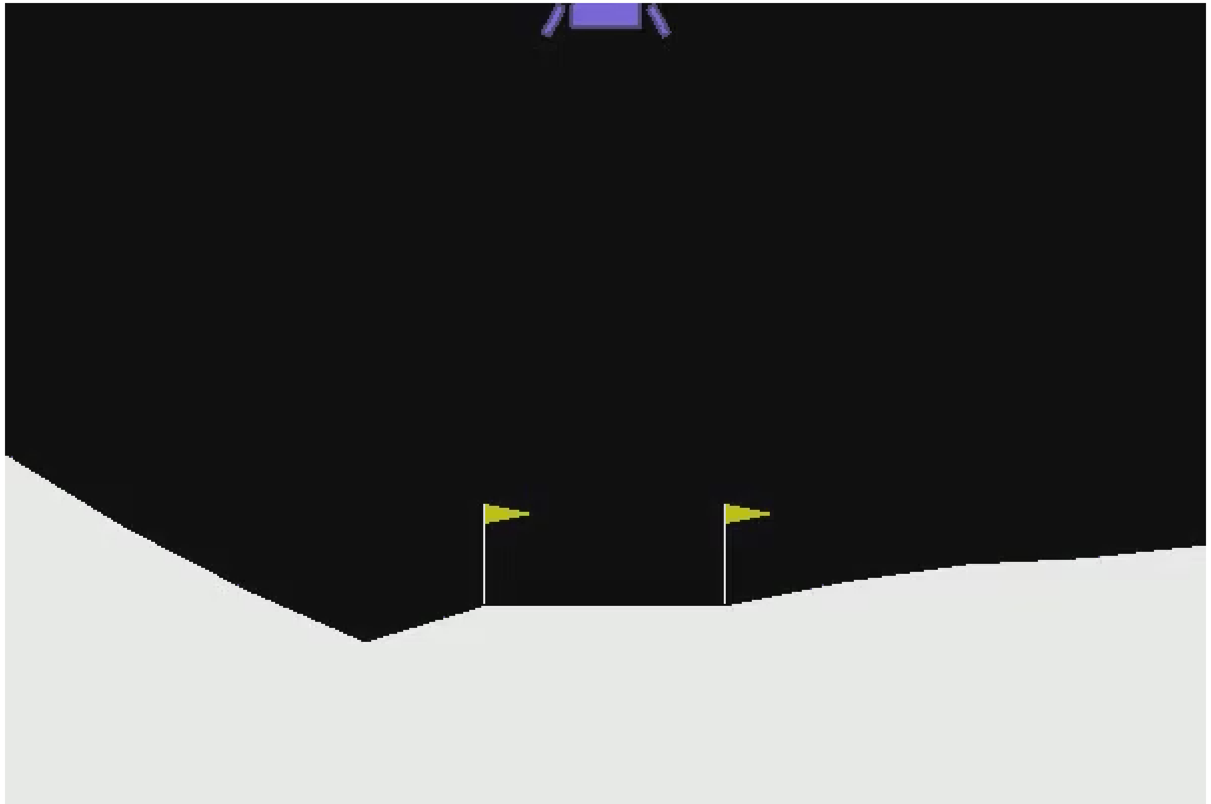




https://youtu.be/3jDoPobFgwA

Reinforcement Learning

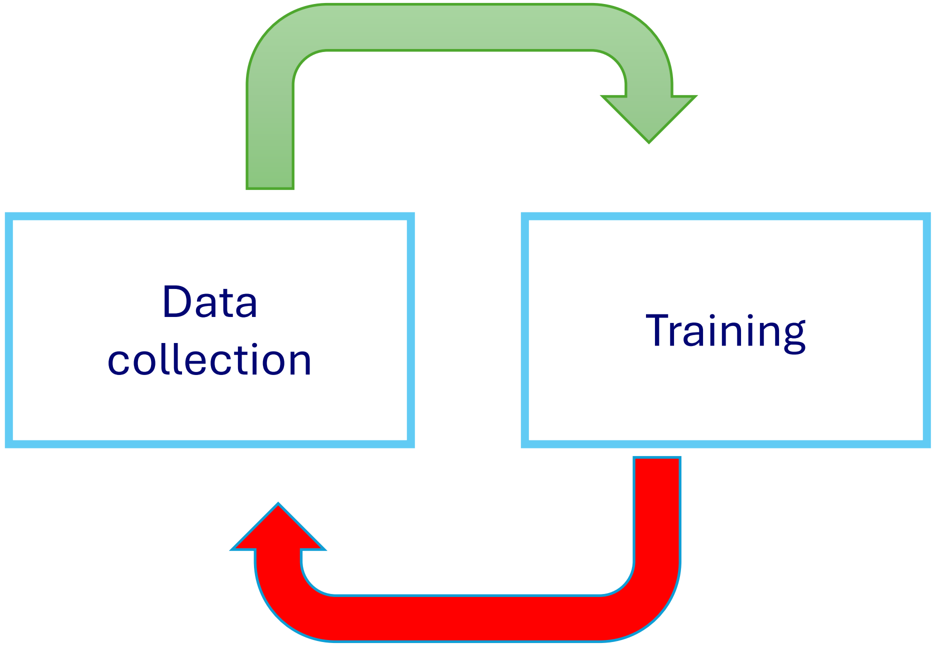
Learning to make sequential decisions.



Reinforcement Learning

The algorithm can collect data interactively

Try the strategy and collect feedbacks



Improve the strategy based on the feedbacks

Natural Language Processing - Large Language Models

Can NLP improve the efficiency of space mission planning?

1.Yes, Natural Language Processing (NLP) can significantly improve the efficiency of space mission planning in several ways: Requirements Management: NLP can extract semantic information from textual requirements in space engineering documents.

2.It can help find related requirements, enhancing search capabilities in requirement databases.

3.NLP can check consistency of traceability between requirements and identify suspicious or overlapping requirements.

4.Knowledge Reuse: ESA has developed a method using NLP to automatically extract critical information from space mission data to populate a Knowledge Graph.

5.This unified view of the space ecosystem enables mission experts to find unstructured information more efficiently.

6.It allows users to query all knowledge from a single interface instead of reading through thousands of web pages.

7.Information Extraction: NLP can parse data from unstructured datasets related to space missions.

8.It can extract key information about mission requirements, objectives, payloads, propulsion systems, and target environments from various documents.

9.Communication Enhancement: NLP enables spacecraft to comprehend and respond to complex voice commands, aiding astronauts in controlling and monitoring various systems.

10.It helps translate scientific data into human-readable formats, facilitating seamless collaboration between scientists and engineers.

11.Data Analysis: NLP techniques can analyze large volumes of space-related text data, such as scientific papers, mission reports, and news articles.

12.This analysis can uncover trends, relationships, and insights that may not be immediately apparent to human planners.

13.Predictive Analytics: By analyzing historical mission data and documentation, NLP can help predict potential issues or challenges in future missions

14.Automated Documentation: NLP can assist in generating and organizing mission documentation, saving time and ensuring consistency.

These applications of NLP can significantly streamline the mission planning process, reduce human error, and enable more efficient use of existing knowledge and resources in space mission design and execution.

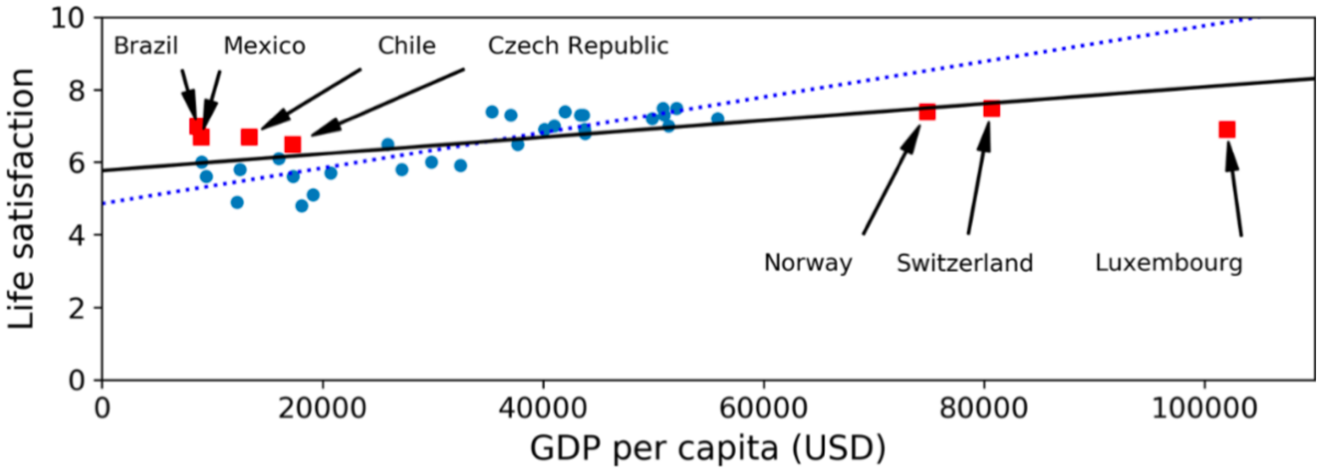
Machine Learning Pipeline

- Study the data (data preparation)
- Select a model or learning algorithm
- Train on the available data and assess performance
- Apply the model to make predictions on new cases

Challenges of machine learning

- Insufficient quantity of training data

- Non-representative and poor-quality training data

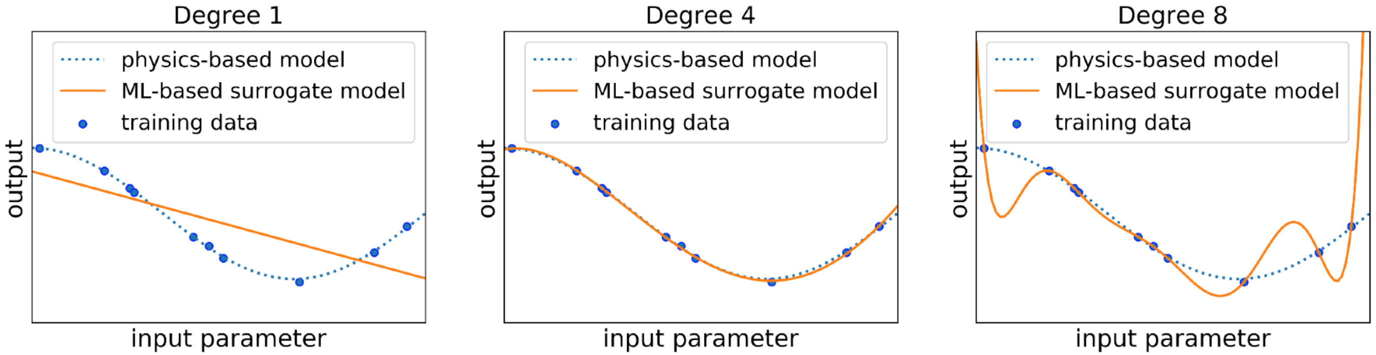


- Irrelevant features: coming up with a good set of features

Challenges of machine learning

Overfitting the training data

- When the model is too complex relative to the amount and noisiness of data



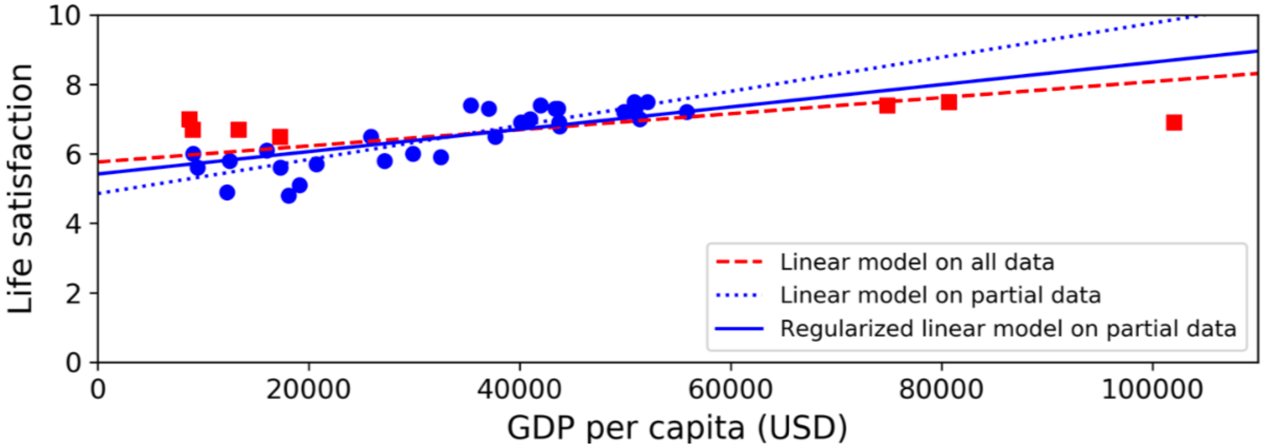
How to alleviate this problem?

- Simplify the model
- Gather more training data
- Reduce noise and outliers (i.e., data preparation)

Challenges of machine learning

Use a subset of data (train, test and validation splits) and regularization

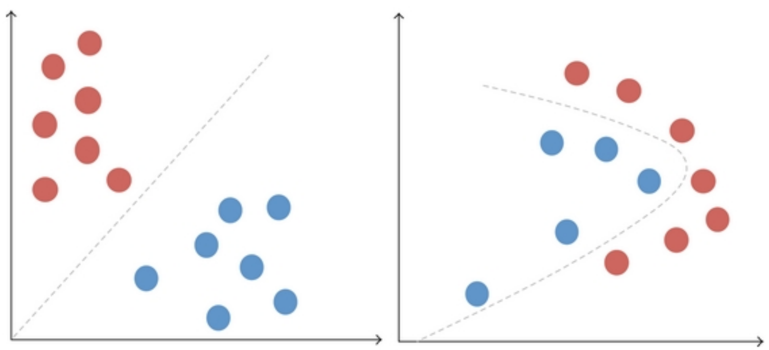
- Regularization: balance between fitting the data perfectly and keeping the
- model simple enough
- A hyperparameter controls the amount of regularization (must be set prior
- to learning and remains constant during training)
- Tuning hyperparameters is extremely important



Challenges of machine learning

Underfitting the training data

- When the model is too simple

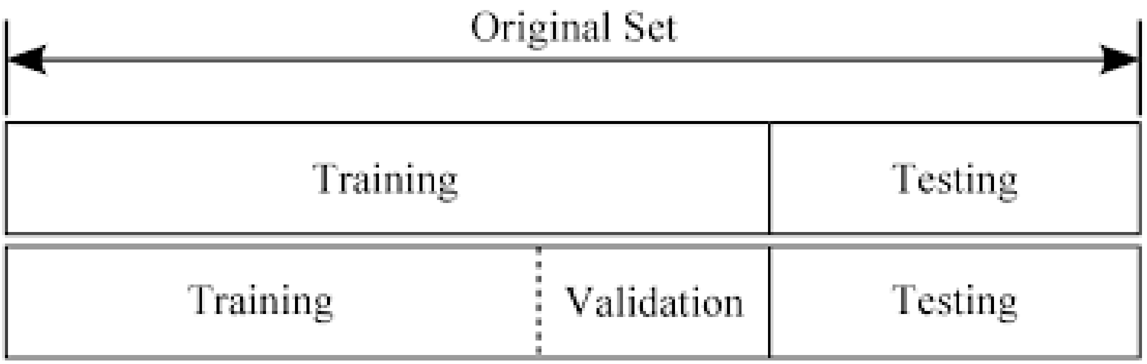


How do we solve this problem?

- Select a more powerful model
- Reduce constraints on the model, such as regularization

Challenges of machine learning

- The only way to know how a model will generalize to new cases is to try it out on new cases
- Split the available data into two main sets: training set and testing set
- The error on the test set is called the generalization error or out-of-sample error
- If the training error is low and the generalization error is high, then overfitting occurs
- Choose suitable models with an adequate set of hyperparameters


Original Set
Training
Testing
Training
Validation
Testing