Unsupervised learning for materials informatics

GDS Short Course on Data Science
for Physicists I

Trevor David Rhone

Department of Physics, Applied Physics and Astronomy,
Rensselaer Polytechnic Institute

Google colaboratory excercise preparation

- Navigate to the GDS github page
  - https://github.com/quantum-intelligence/materials-informatics-tutorial
- Open Google colab
  - Load the google colab file associated with this tutorial
- Register with the materials project
  - https://next-gen.materialsproject.org/
  - Record your API key

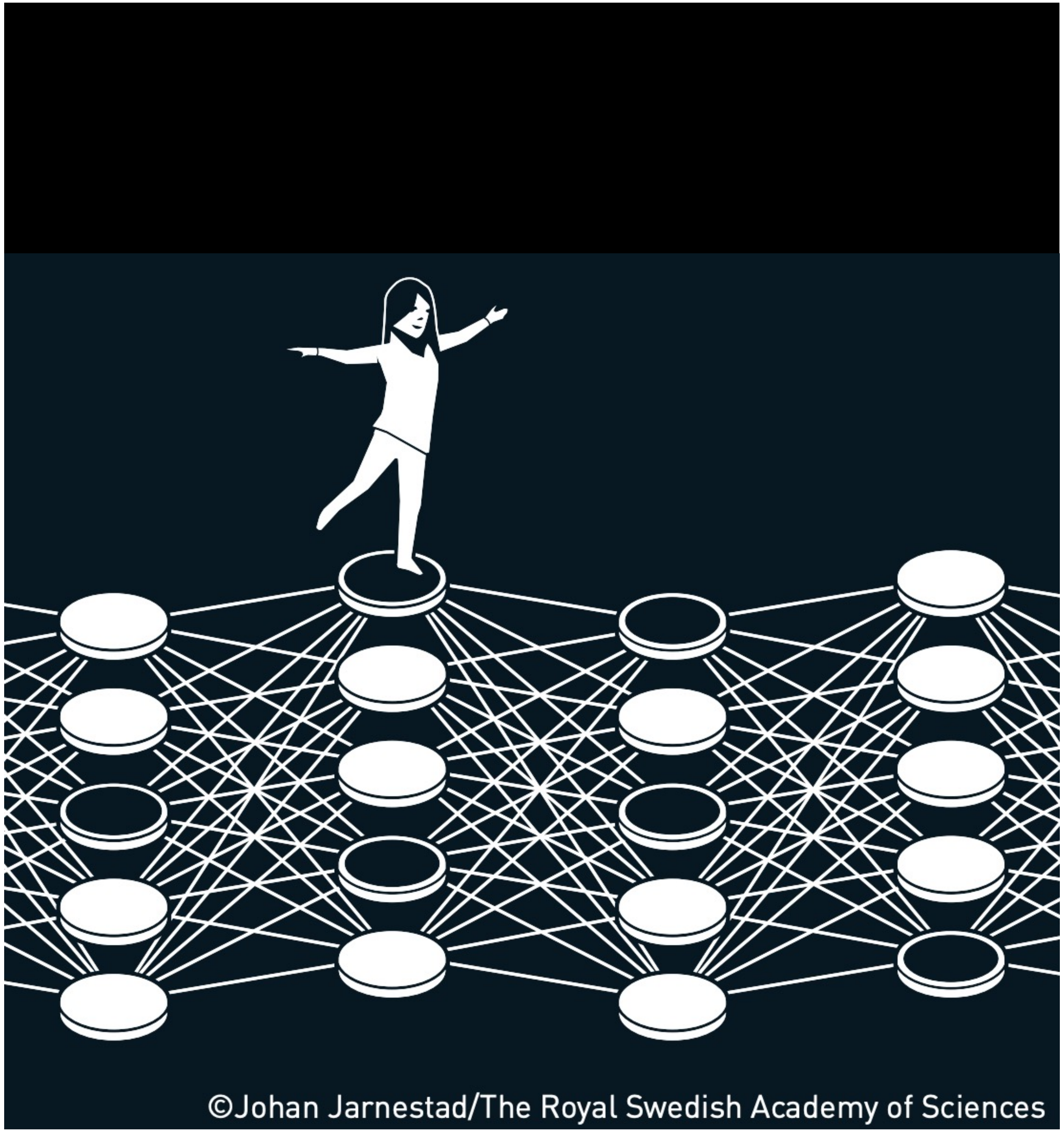
Introduction

- Challenges for materials discovery
  - Predicting properties of materials with experiments (ab initio)
  - Materials design
  - Knowledge discovery
- Materials informatics
  - Materials science + machine learning
- Challenges for materials informatics
  - Lack of labelled data

Nobel prize in physics goes to AI

John J. Hopfield
Princeton University, NJ, USA
Geoffrey Hinton
University of Toronto, Canada

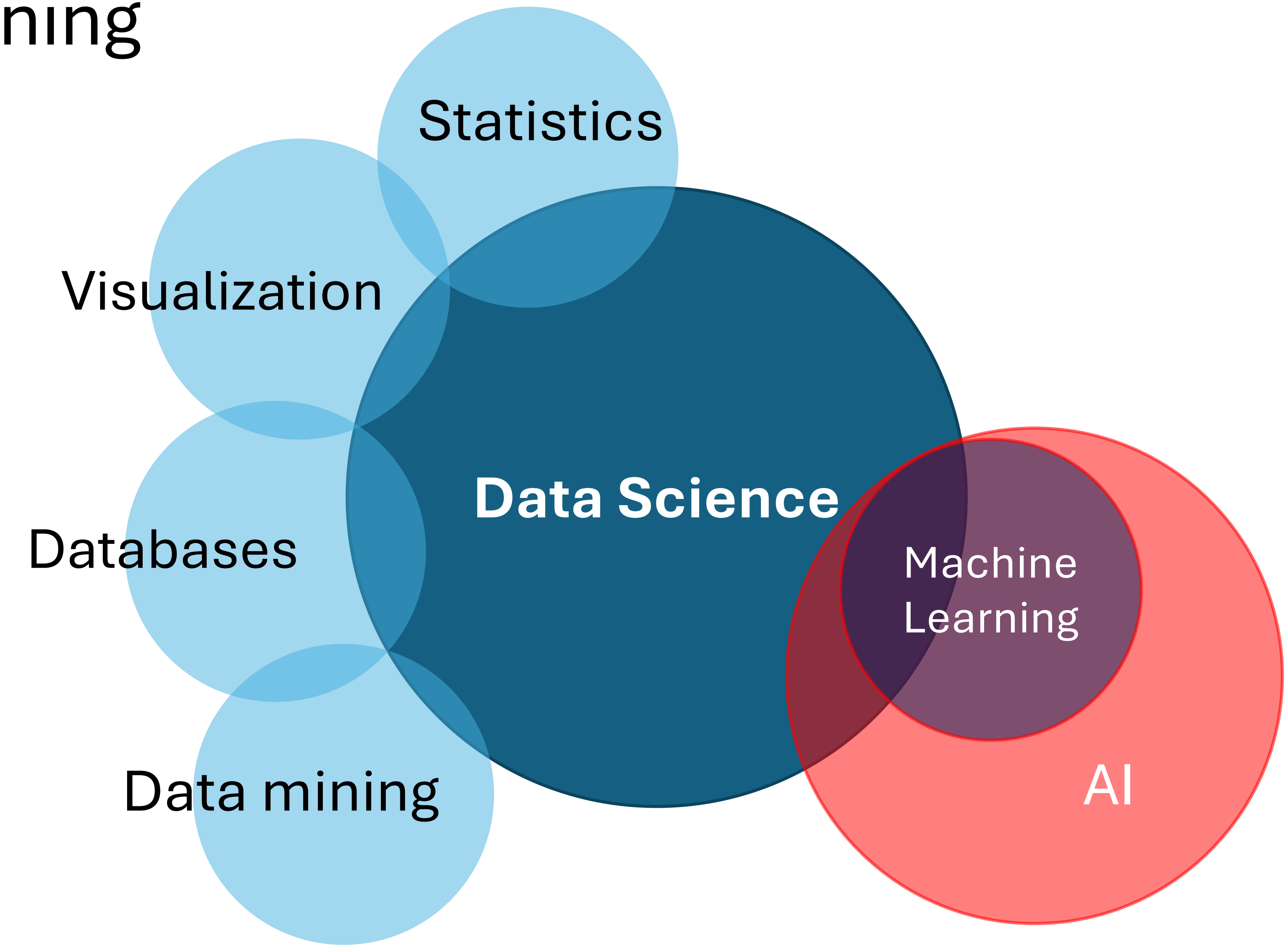
“for foundational discoveries and inventions that enable machine learning with artificial neural networks”



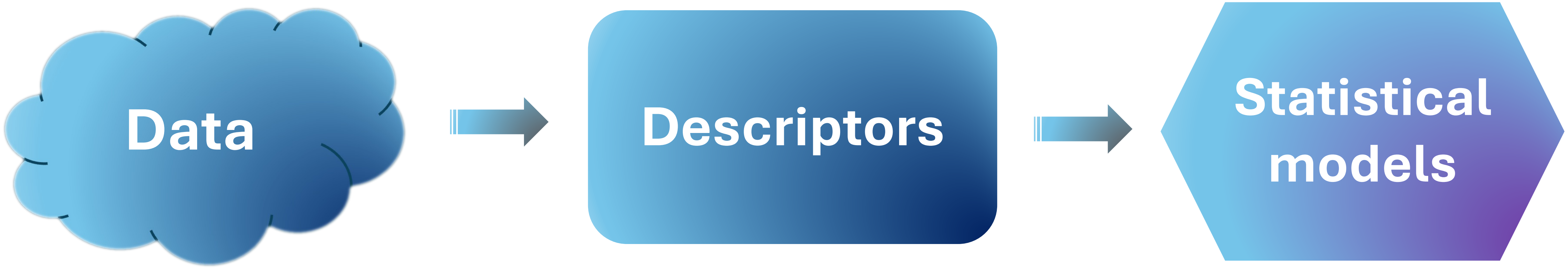
©Johan Jarnestad/The Royal Swedish Academy of Sciences

Machine learning overview

Machine learning



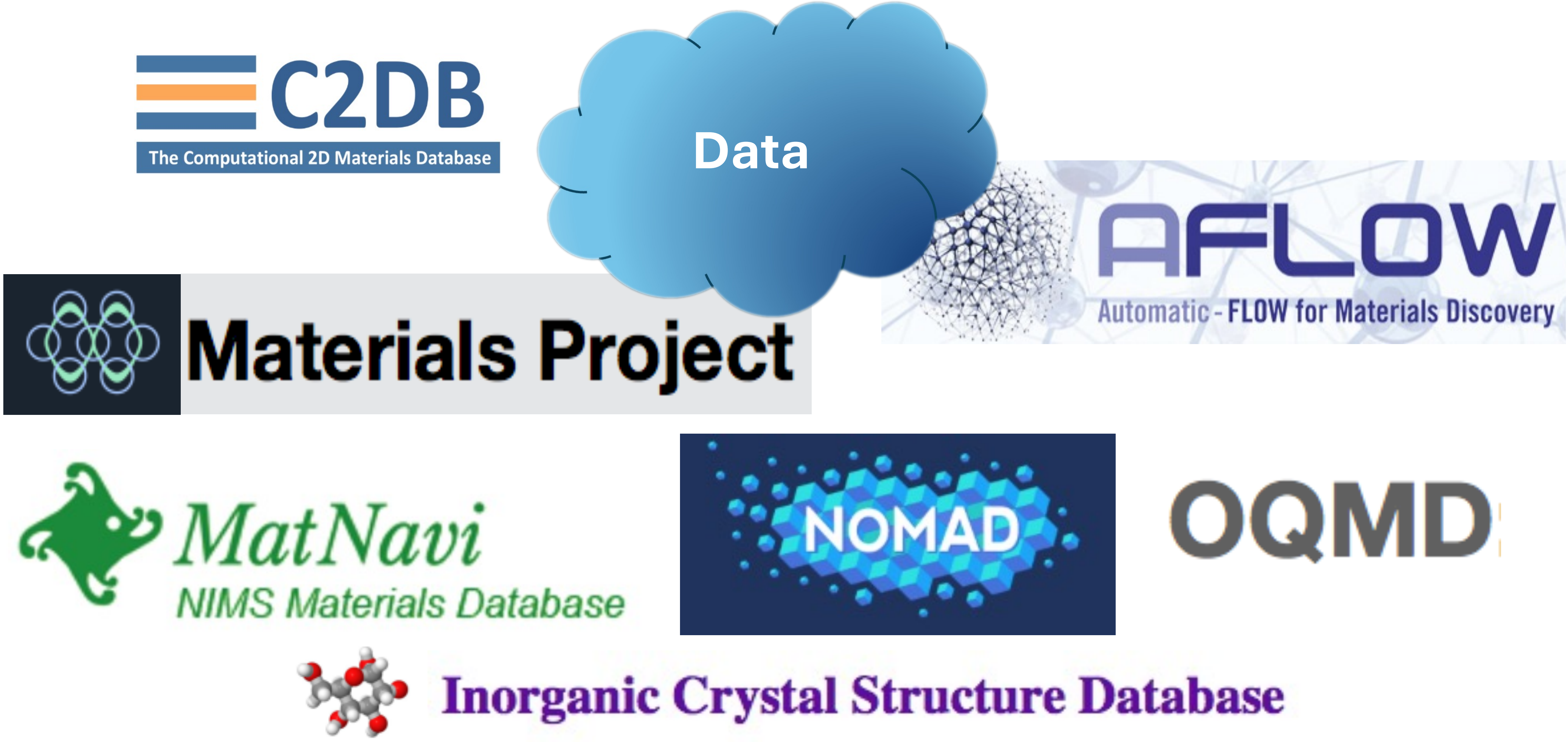
Machine Learning in materials physics



- ○ The rise of the materials databases
  - Data are accessible
- ○ Chemical space descriptors exist
  - Coulomb Kernel[1], Bag of bonds representation[2]
- ○ Datascience tools exist
  - Scikit learn, Google's TensorFlow, Pytorch

[1] M. Rupp, et al., Phys Rev Lett. 108, 058301 (2012) [2] K. Hansen et al., J. Phys. Chem. Lett. 6, 2326–2331 (2015)

Machine Learning in materials physics


Data
C2DB
The Computational 2D Materials Database
AFLOW
Automatic - FLOW for Materials Discovery
Materials Project
MatNavi
NIMS Materials Database
NOMAD
OQMD
Inorganic Crystal Structure Database

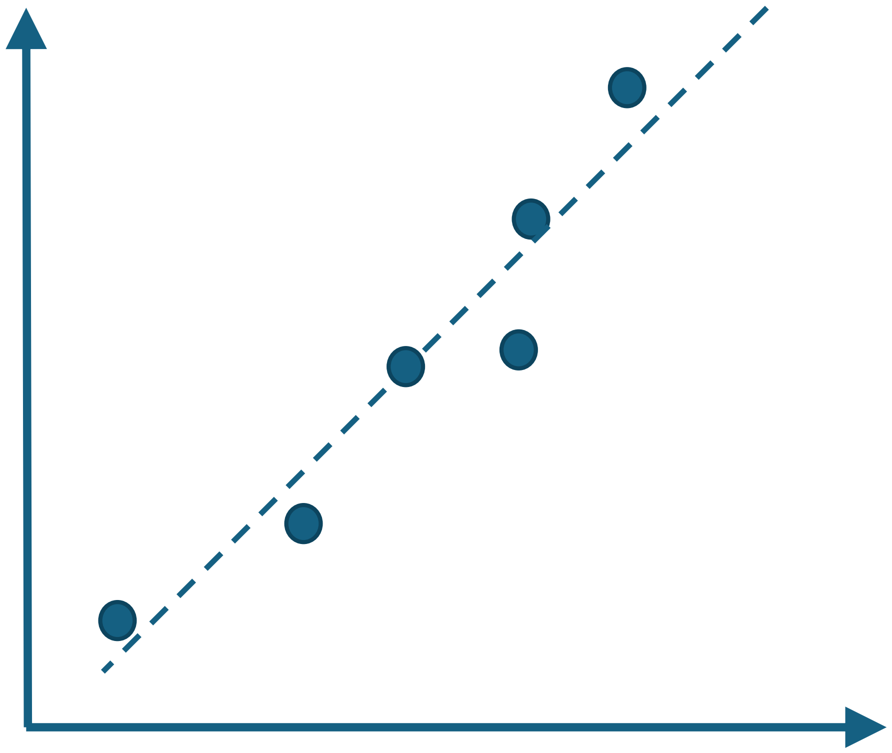
Machine Learning in materials physics

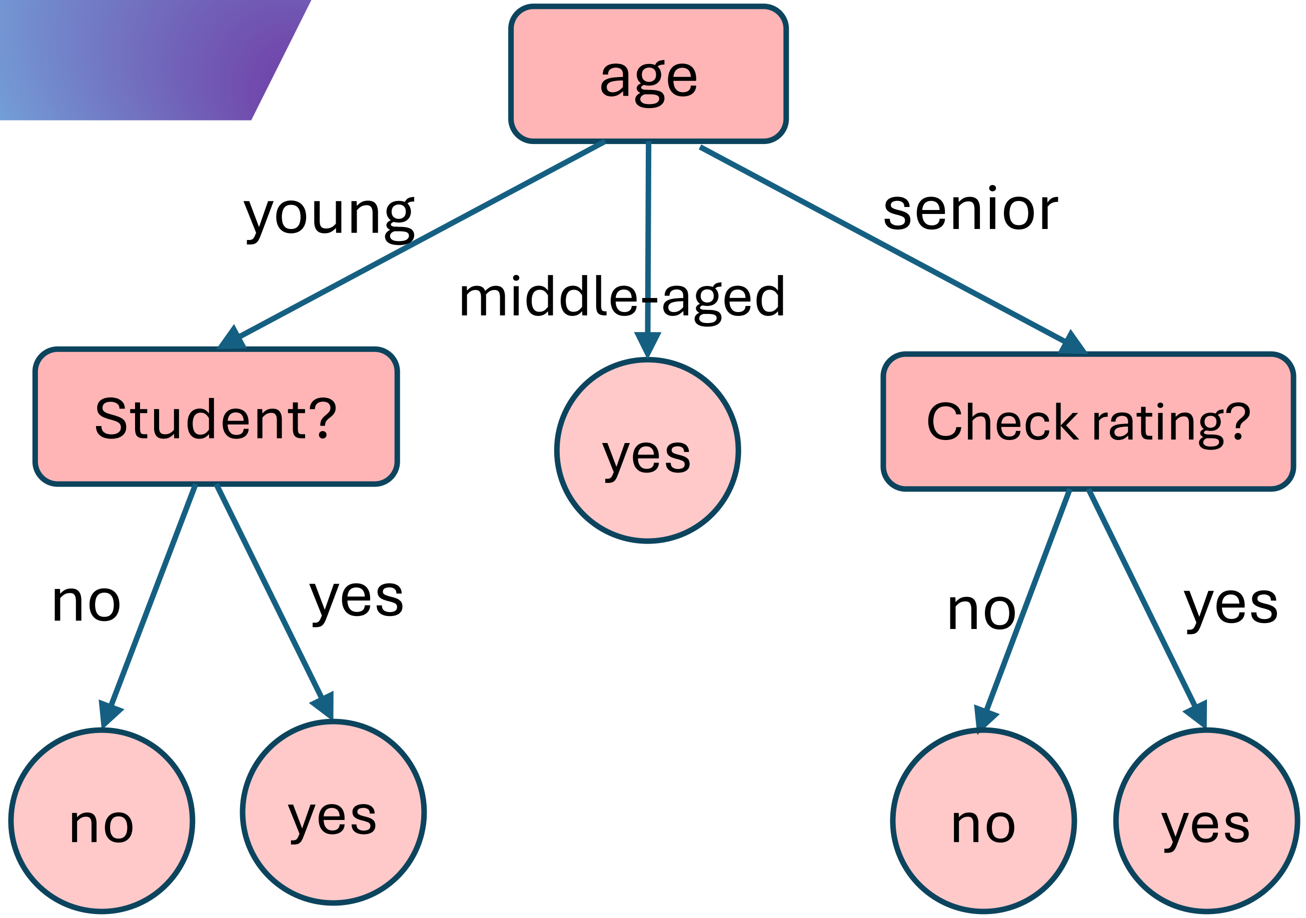
Descriptors

- Description of material
  - Atomic properties
  - Mathematical representations of crystal structure
- Example:
  - Ionic compounds have atoms in different columns of periodic table
    - Descriptor: column # of the periodic table

Machine Learning in materials physics

Statistical models





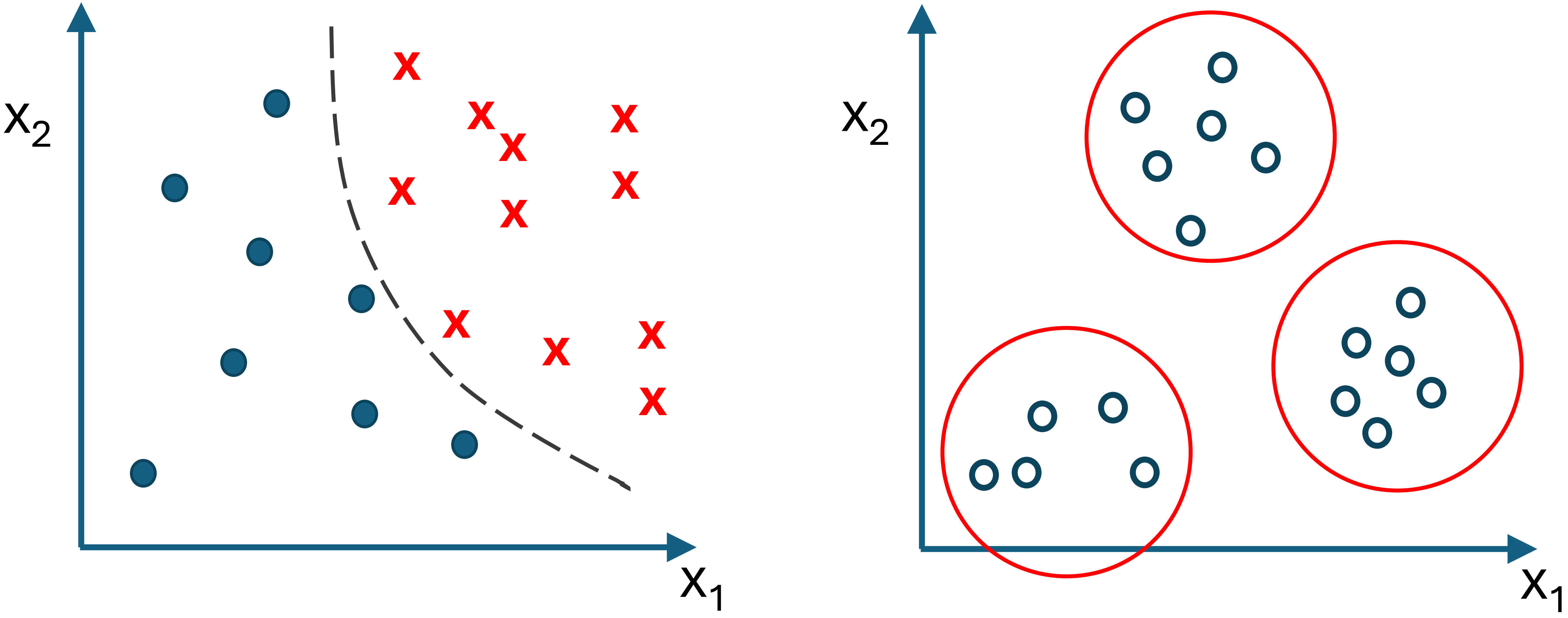
Machine Learning in materials physics

Data science ~ data visualization + machine learning

$$\underbrace{y}_{\text{Target property}} = f(\underbrace{x_1, x_2, \ldots, x_N}_{\text{Inputs of machine learning model}}) + \boldsymbol{\varepsilon}$$

Goal: learn or quantify some relationship

Supervised versus Unsupervised learning



Supervised versus Unsupervised learning

Supervised learning: For each observation of the predictor measurement(s) x_i, i = 1,...,n there can be an associated response measurement y_i.

We wish to fit a model that relates the response to the predictors to:

- *accurately predicting the response for future observations* (prediction) or
- better understanding the relationship between the response and the predictors (inference).

Unsupervised learning describes the situation in which for every observation i = 1,...,n, we observe a vector of measurements x_i but no associated response y_i

Unsupervised learning

- Dimensionality reduction
    - PCA (principal components analysis) looks to find a low-dimensional representation of the observations that explain a good fraction of the variance
    - tSNE (t-distributed stochastic neighbor embedding)
- Clustering looks to find homogeneous subgroups among the observations
    - K-means clustering
    - Hierarchical clustering
- Associative learning
    - Apriori algorithm
    - FP-growth algorithm
- Generative models
    - Variational autoencoder (VAE)
    - Generative adversarial network (GAN)
    - Transformer based generative models, eg. GPT

Clustering

- Finds subgroups, or clusters, in a data set.
  - Partition observations in a dataset into distinct groups so that
    1. observations within each group are similar, and
    2. observations in different groups are different
- Clustering could be used to find subgroups which can be mapped to subclasses of materials, for example:
  - Nonmagnetic materials
  - ferromagnets and
  - antiferromagnets
- Types of clustering algorithms
  - K-means clustering
  - Hierarchical clustering

Gareth James, Daniela Witten, Trevor Hastie Robert Tibshirani , An Introduction to Statistical Learning , DOI 10.1007/978-1-4614-7138-7

K-means Clustering

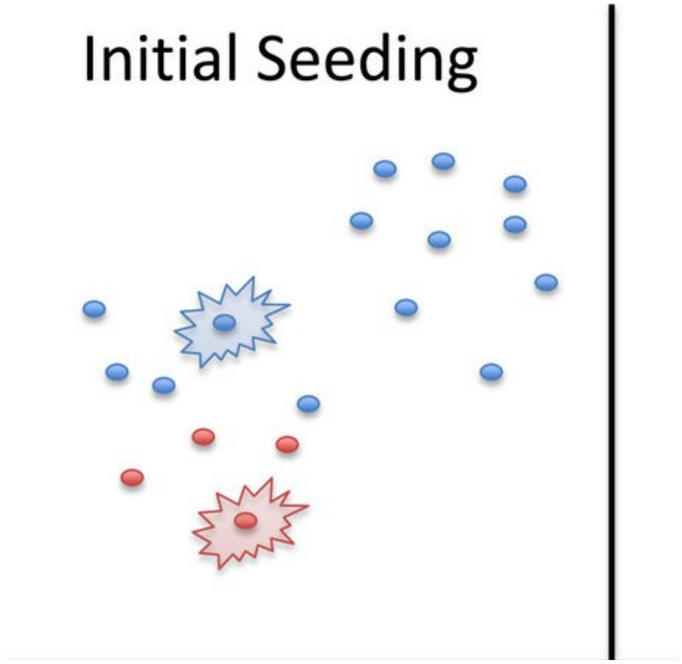
- Partition observations into a pre-specified clustering number of clusters
- Goal is to partition data set into K distinct, non-overlapping clusters
  - first specify the desired number of clusters K
  - the K-means algorithm assigns each observation to exactly one of the K clusters

$$\underset{C_1,\ldots,C_K}{\text{minimize}} \left\{ \sum_{k=1}^{K} W(C_k) \right\}$$

$$W(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2$$

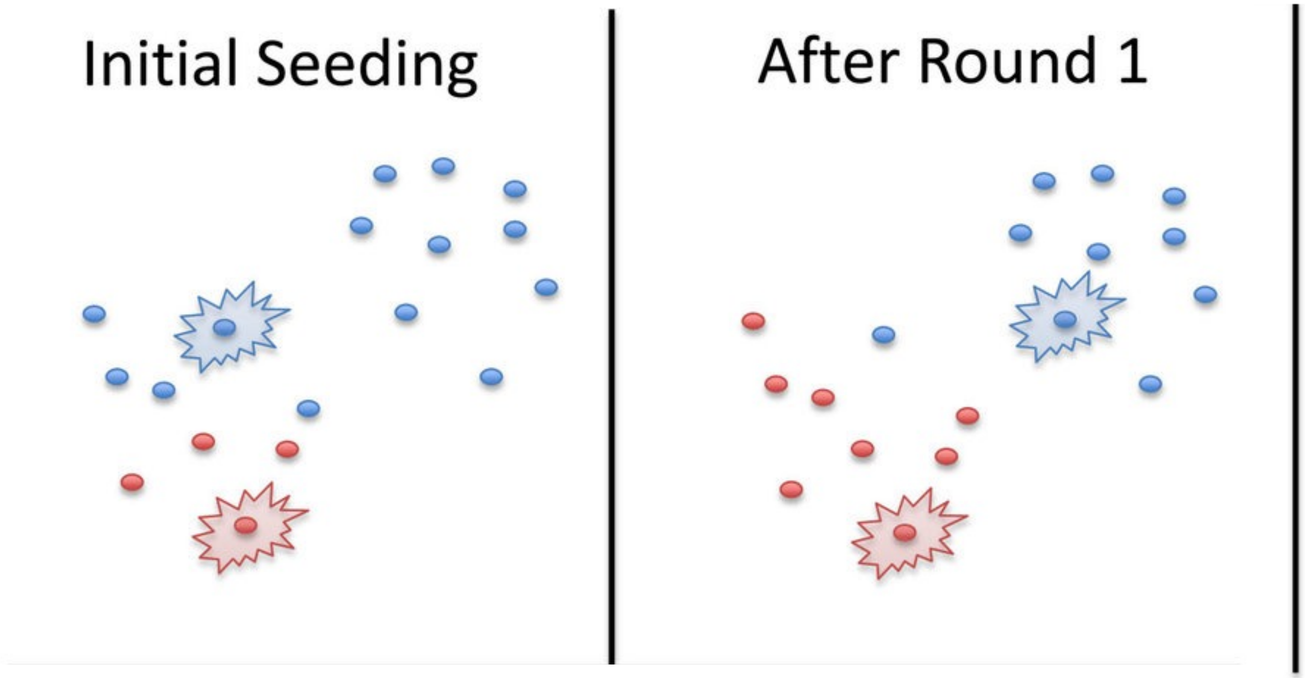
Gareth James, Daniela Witten, Trevor Hastie Robert Tibshirani , An Introduction to Statistical Learning , DOI 10.1007/978-1-4614-7138-7

K-means Clustering



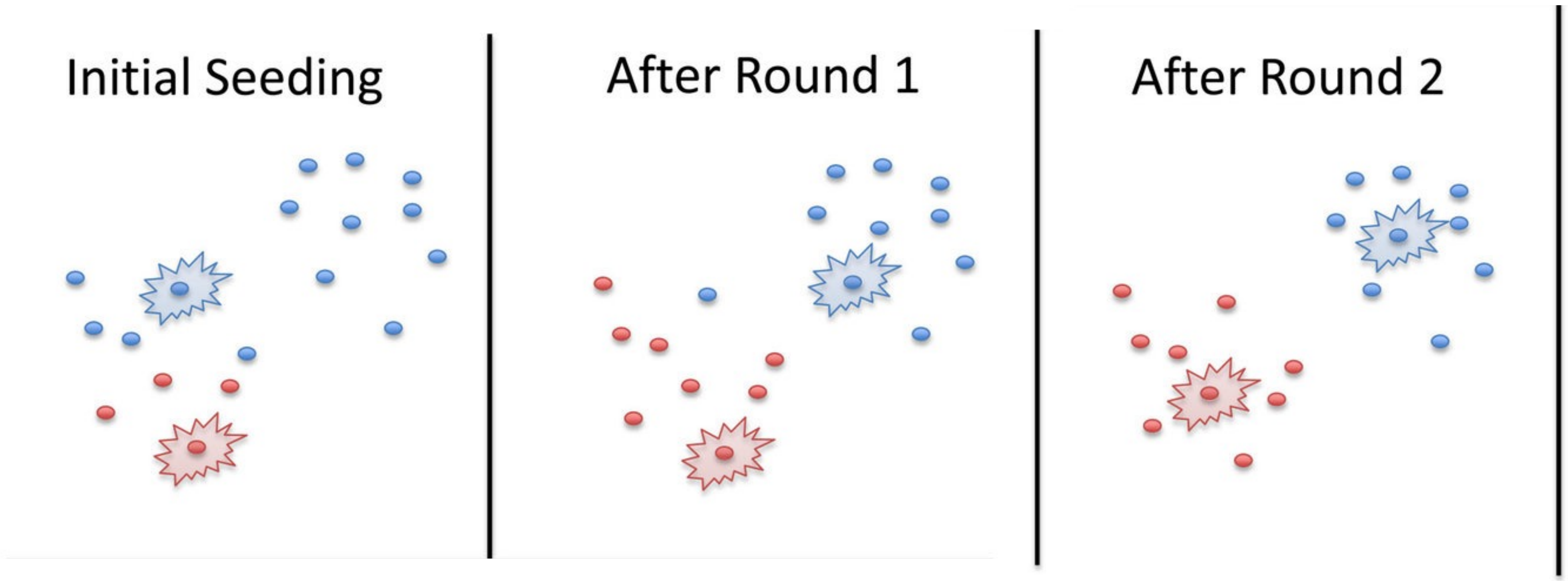
https://devopedia.org/k-means-clustering

K-means Clustering



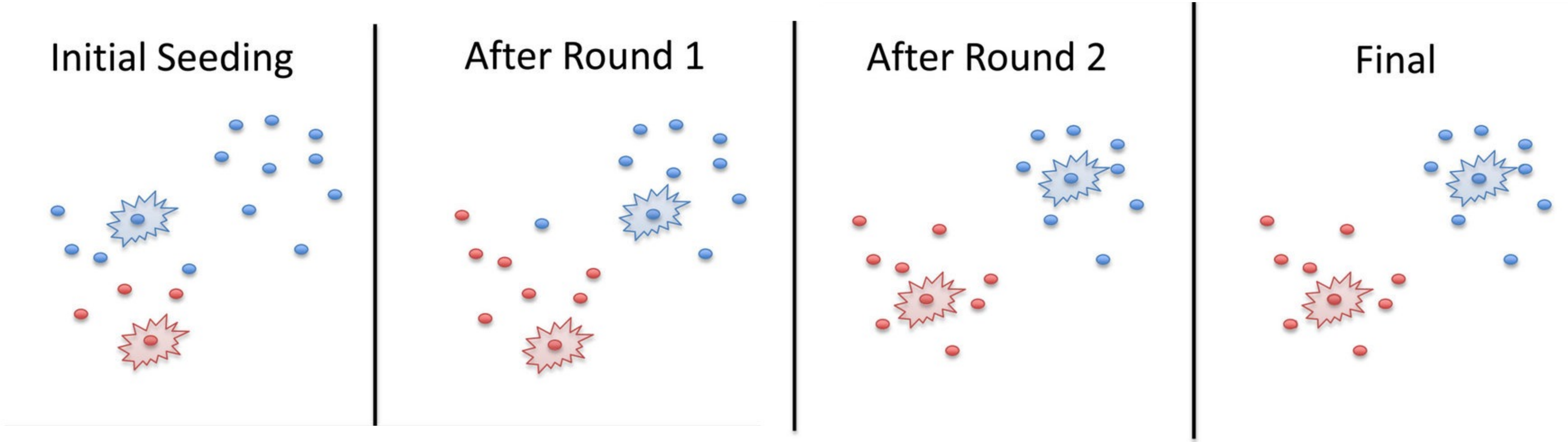
https://devopedia.org/k-means-clustering

K-means Clustering



https://devopedia.org/k-means-clustering

K-means Clustering



https://devopedia.org/k-means-clustering

K-means Clustering

- Hands-on example

Dimensionality reduction

- o Dimension Reduction
  - Project p predictors into an M-dimensional subspace, where M < p
  - These M projections can be used as predictors to fit a regression model
  - M=2, 3 projections can be used to create visualizations that create insight into patterns in the data
- Dimensionality reduction methods
  - Principal components analysis (PCA)
  - t-distributed stochastic neighbor embedding (tSNE)
  - Autoencoders

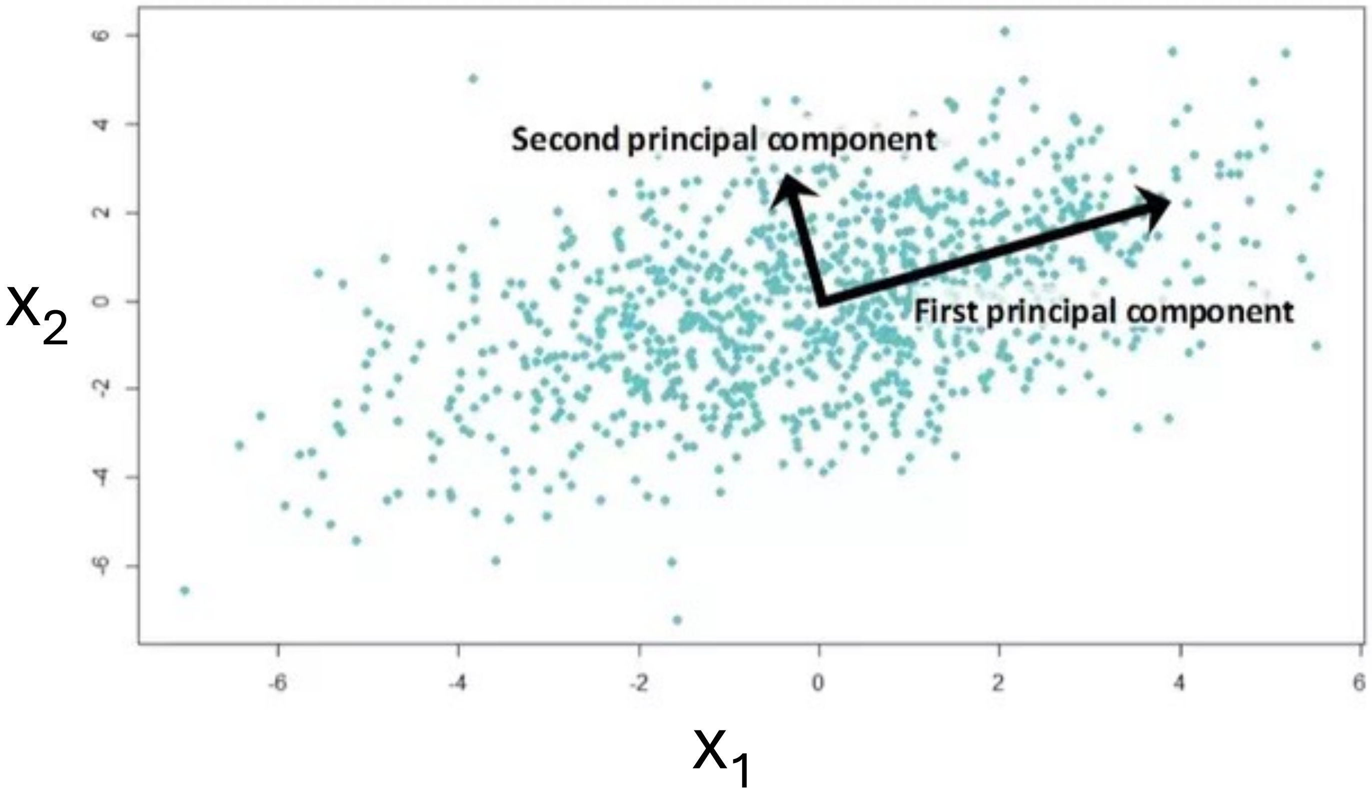
Dimension Reduction Methods

Principal Components Analysis

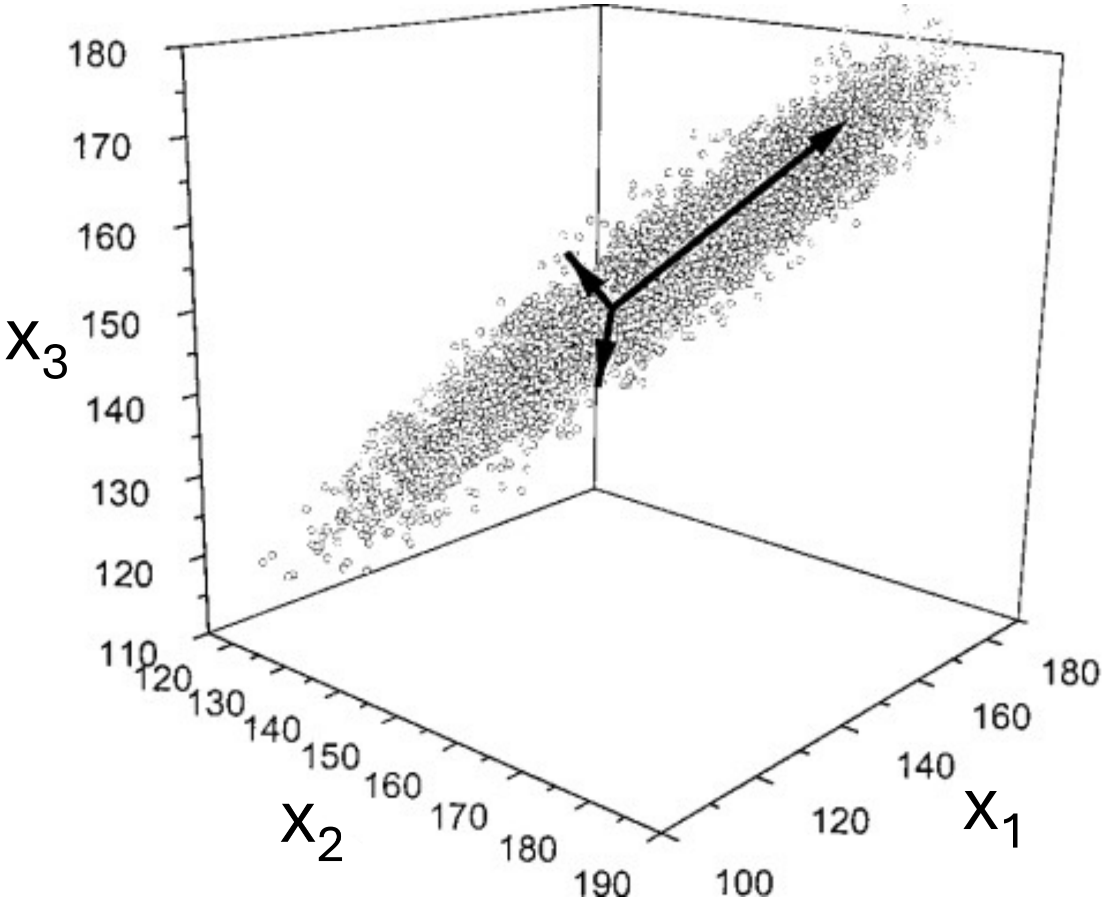
- Principal components summarize a data set with a smaller number of variables that collectively explain most of the variability in the original set
- PCA is an unsupervised approach, since it involves only a set of features $X_1, X_2, \dots, X_p$, and no associated response Y
- Uses:
  - Produces variables for use in supervised learning
  - Serves as a data visualization tool

Principal Component Analysis

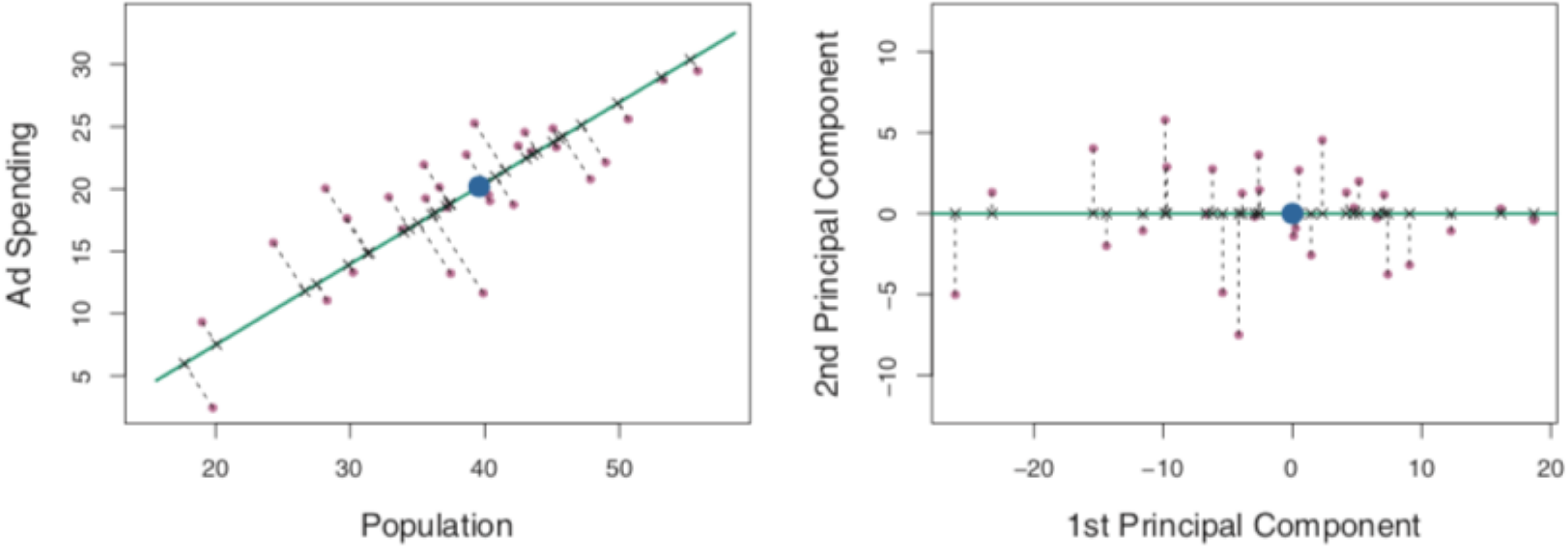
Two-dimensions



Three-dimensions



Principal Component Analysis



Considerations in High Dimensions

- Traditional statistical techniques for regression/classification are intended for the low-dimensional setting in which n (the # of observations) >> p (the # of features)
- In the past 20 years, new technologies have changed the way that data are collected. We now collect a large number of features, p.
- The number of observations n is often limited due to sample availability, etc.
- E.g., Rather than predicting blood pressure on the basis of age, gender, and BMI, one might *also* collect measurements for half a million low-dimensional single nucleotide polymorphisms (i.e. DNA mutations)
  - Then $n \approx 200$ and $p \approx 500{,}000$.
- Data sets containing more features than observations are often referred to as high-dimensional.

Principal Component Analysis

- Hands-on example

Associative rule learning

- Association rule learning is a rule-based ML method for discovering interesting relations between variables in large databases
- It identifies strong rules discovered in databases using several measures
- In any given transaction with a variety of items, association rules are meant to discover the rules that determine how certain items are connected

Associative rule learning

Algorithm	Use case	Efficiency	Cons
Apriori	Small the medium datasets	Slow on large datasets	High computational cost due to candidate generation
FP-Growth	Large datasets with complex relationships	Faster than Apriori	More complex

Associative rule learning

Apriori algorithm: identifies frequent itemsets and generates association rules from them

- Support
  - The proportion of materials that contain an item or itemset
- Confidence
  - The likelihood that item Y appears given that item X is already present
- Lift
  - Measures how much more likely Y is to appear with X compared to chance

$$\text{Support}(X) = \frac{\text{Transactions containing } X}{\text{Total transactions}}$$

$$\text{Confidence}(X \Rightarrow Y) = \frac{\text{Support}(X \cup Y)}{\text{Support}(X)}$$

$$\text{Lift}(X \Rightarrow Y) = \frac{\text{Confidence}(X \Rightarrow Y)}{\text{Support}(Y)}$$

Lift > 1: X and Y are positively correlated.
Lift = 1: No correlation.
Lift < 1: X and Y are negatively correlated.

Associative rule learning

- Step 1: Find Frequent Itemsets
  - Scan the dataset and find items that appear in at least a given support threshold
  - Example: If we set **support ≥ 10%**, only elements appearing in **at least 10%** of materials are considered
- Step 2: Generate Candidate Itemsets
  - Combine frequent single elements into pairs, then triplets, and so on
  - Example:
    - Frequent **single elements**: {Ti}, {O}, {Fe}, {Ni}
    - Generate **pairs**: {Ti, O}, {Fe, O}, {Ni, O}, etc.

Associative rule learning

- Step 3: Prune Non-Frequent Itemsets
  - If a set is not frequent, its supersets cannot be frequent either.
  - Example:
    - If {Ti, O} is frequent, we check {Ti, O, Fe}.
    - If {Ti, Fe} is not frequent, we discard {Ti, O, Fe}.
- Step 4: Generate Association Rules
  - From frequent itemsets, we generate rules with high confidence.
  - Example:
    - Rule: {Ti, O} → {High Band Gap}
    - Confidence: 80% (If a material contains Ti and O, it has a high band gap in 80% of cases).
    - Lift: 1.5 (Having Ti and O increases the likelihood of a high band gap).

Associative rule learning

- Example for oxide materials and band gap

Rule	Support	Confidence	Lift
{Ti, O} → {High band gap}	0.25	0.80	1.5
{Fe, O} → {Low band gap}	0.20	0.70	1.3
{Ti, O} → {High chemical stability}	0.30	0.82	1.8

- Rule #1: If a materials contains Ti and O, there’s an 80% probability that is has a high band gap
- Rule #2: If a material contains Fe and O. it’s more likely to have a low band gap

Associative rule learning

- Hands-on example

Generative models

- Generative models
  - learn the underlying data distribution and generate new data samples that resemble the original dataset
- Discriminative models
  - predict labels, generative models create new data

	Generative models	**Discriminative models**
Goal	Learn data distribution and generate materials	Learn decision boundary for classification
Examples	VAEs, GANs, Diffusion models	Decision Trees, SVM, Neural networks
Output	New materials	Labels or class probabilities

Types of Generative Models

- Probabilistic Generative Models
- Deep Learning-Based Generative Models
- Transformer-Based Generative Models

Deep Learning-Based Generative Models

Neural networks used to learn complex data distributions

- Variational Autoencoders (VAEs)
  - Encodes data into a latent space, then reconstructs it.
  - Used in drug discovery, and materials discovery
- Generative Adversarial Networks (GANs)
  - Comprises a Generator (creates fake data) and a Discriminator (distinguishes real from fake)
  - Used for microscopy image synthesis, material microstructure simulation, and molecule generation
- Diffusion Models
  - Adds noise to data, then learns to denoise the data
  - Used to generate novel molecules and crystal structures

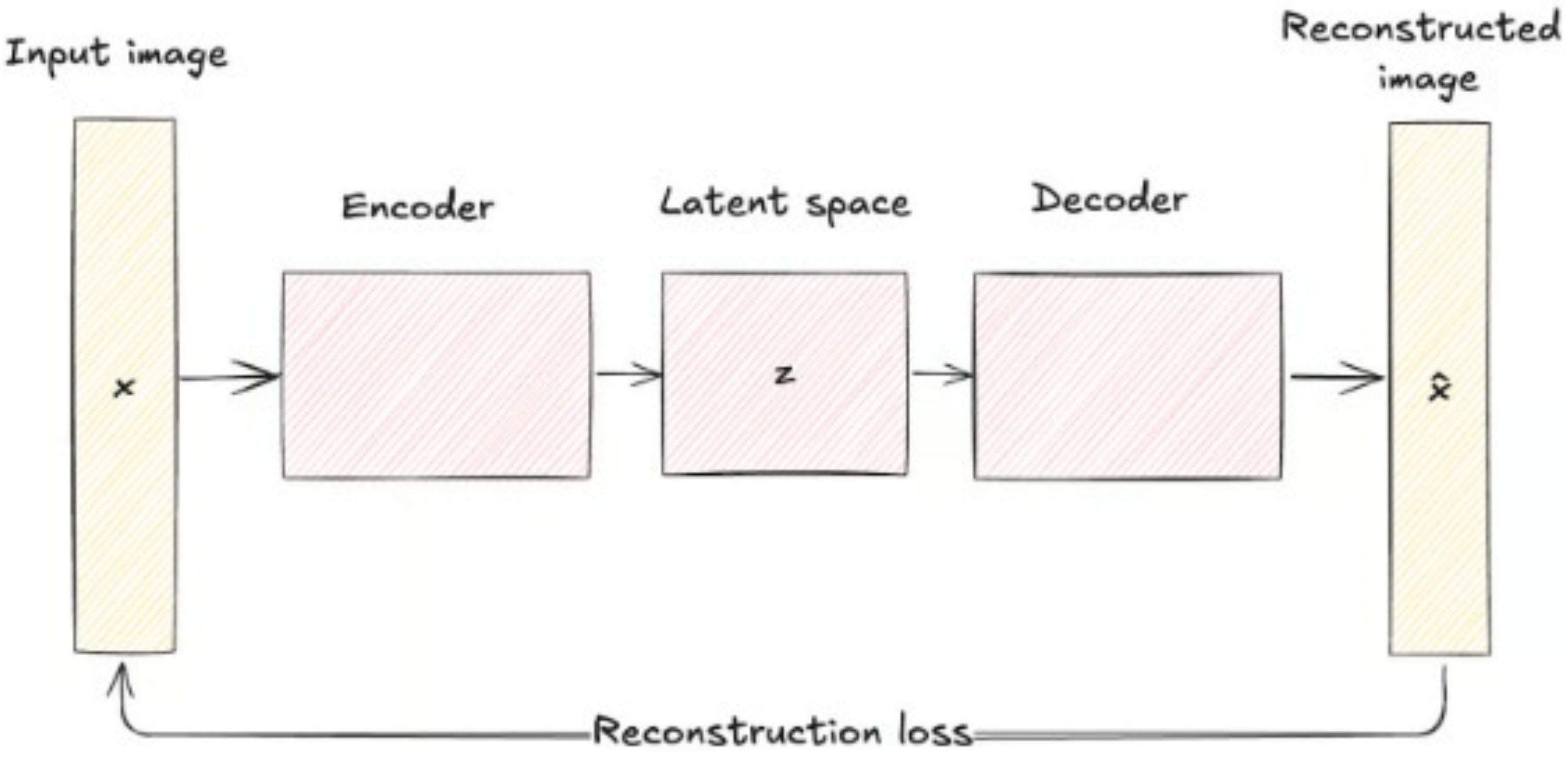
Transformer-Based Generative Models

These use self-attention mechanisms to model long-range dependencies

- GPT (Generative Pre-trained Transformer)
  - Used for text generation, code generation, and language modeling
  - Example: data-mining materials science research papers

Variational autoencoder (VAE)

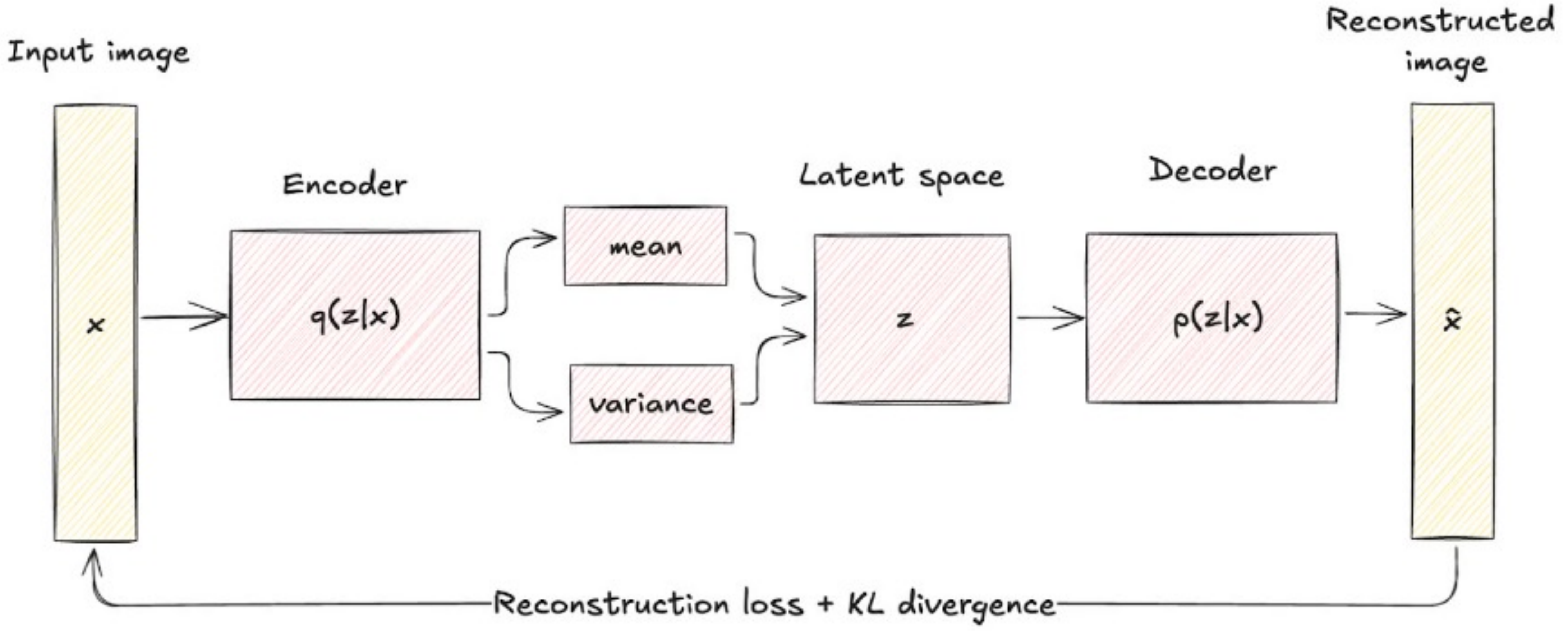
- The VAE architecture
  - Similar to the autoencoder architecture



https://www.datacamp.com/tutorial/variational-autoencoders

Variational autoencoder (VAE)

- The VAE architecture



https://www.datacamp.com/tutorial/variational-autoencoders

Variational autoencoder (VAE)

○ The VAE architecture

The VAE introduces a probabilistic element into the encoding process

- The VAE encoder maps the input data to a probability distribution over the latent variables
- This is modeled as a Gaussian distribution with mean μ and variance σ^2

- VAEs incorporate regularization through the KL divergence
- The KL divergence makes the latent space continuous and well-structured

https://www.datacamp.com/tutorial/variational-autoencoders

Generative Models

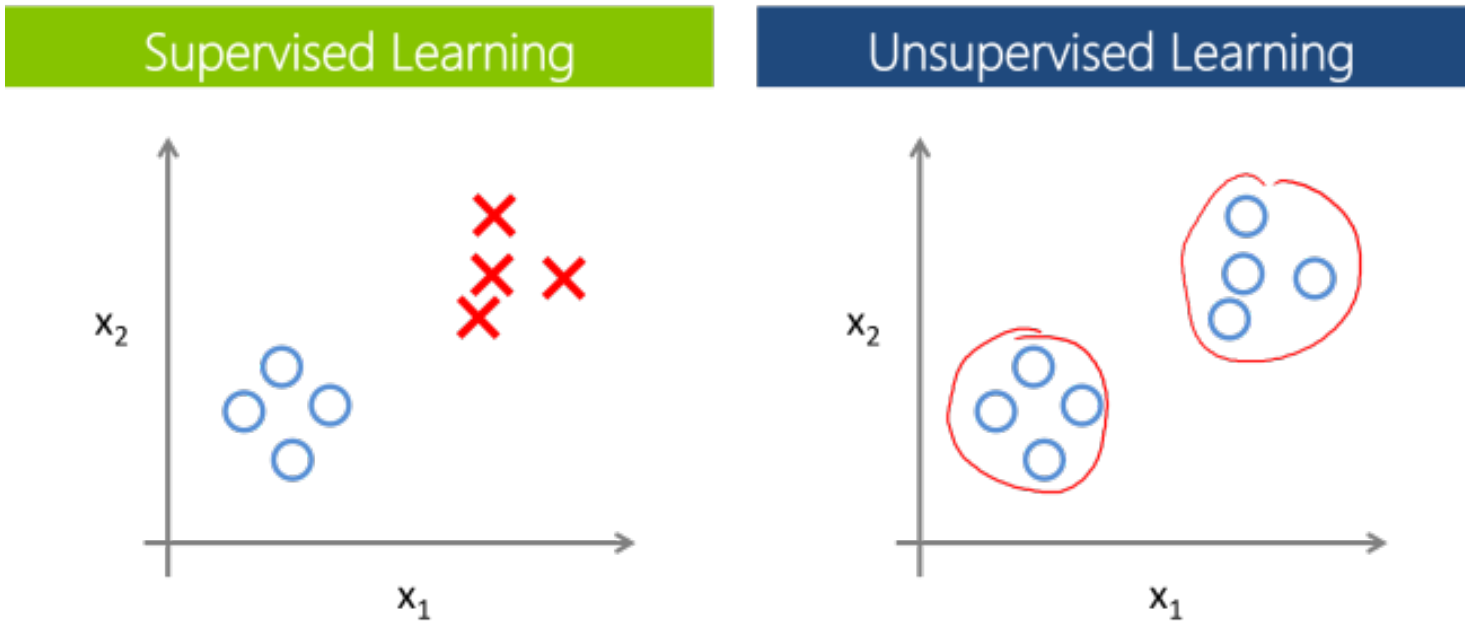
- Hands-on example

Keras and Tensorflow tutorials:
https://keras.io/examples/generative/vae
https://www.tensorflow.org/tutorials/generative/autoencoder

Summary

Unsupervised learning can identify patterns in data

- Clustering
  - K-means clustering
- Dimensionality reduction
  - PCA
- Associative rule learning
  - Apriori
- Generative models
  - Variational autoencoders



Supervised Learning

Unsupervised Learning

Outlook

- Semi-supervised learning
  - Unsupervised + supervised learning
  - Leverage unlabeled data to improve supervised learning tasks

Artificial Intelligence Guided Studies of van der Waals Magnets

T. D. Rhone *et al.*, “Artificial Intelligence Guided Studies of van der Waals Magnets,” *Adv. Theory Simulations*, vol. 6, no. 6, p. 2300019, 2023.

Hands-on project

- $CrGeTe_3$-type vdW magnets dataset
- Identify patterns in data using unsupervised learning
- $CrGeTe_3$ materials in this dataset can be ferromagnetic (FM) or antiferromagnetic (AFM)
- Can you find clusters of FM and AFM materials?
  - Data exploration
    - Data visualization
    - PCA
  - K-means clustering
  - How to choose materials descriptors?

T. D. Rhone *et al.*, “Data-driven studies of magnetic two-dimensional materials,” *Sci. Rep.*, vol. 10, no. 1, p. 15795, 2020.

Want more information?

www.materials-informatics.com

- GDS Webinar series on YouTube
- Jarvis Tutorials by NIST
- Andrew Ng's Machine Learning course on Coursera

Apply to Rensselaer Polytechnic Institute (RPI)

Rensselaer

Patterns emerge free,
atoms whisper their secrets,
no guide, yet they learn.