Berkeley Engineering | BerkeleyHaas

# PROFESSIONAL CERTIFICATE IN MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Let’s give everyone a couple of minutes to join…

## Module 6

## Data Clustering and Principal Component Analysis

Office Hours with Viviana Márquez
October 10, 2024

# AGENDA

- Assignment: Practical Applications I Feedback QA
- Required activities for Module 6
- Content review Module 6: Data Clustering and PCA
- Questions

# AGENDA

- Assignment: Practical Applications I Feedback QA
- Required activities for Module 6
- Content review Module 6: Data Clustering and PCA
- Questions


*on finishing*

## Section 1: Foundations of ML/AI


*on finishing*

## Section 1: Foundations of ML/AI

*This Section*

## Section 2: ML/AI Techniques

Program Topics

# Content review Module 5: Practical Applications I

- Things we were looking for:
  - **README file**
    The README is the first thing viewers see when they visit your project, and having your summary/analysis there provides immediate clarity and accessibility. This ensures that your project makes a strong impression, offering a concise overview while directing viewers to the full notebook.

  - **Good coding practices**
    This helps make your code more readable and error-free, which saves time and resources in the long run, especially when codes need to be updated or debugged.

  - **Comments**
    Clarify the purpose of code segments, making it easier for others (and yourself in the future) to understand, thus aiding in collaboration and maintenance.

  - **Visualizations**
    Only using bar charts and pie charts is somewhat basic and repetitive. More advanced visualizations like stacked bar plots (to compare accepted vs rejected in a single graph) or faceted plots (showing demographics alongside acceptance rates) could further enhance clarity.

  - **Conclusions**
    This practice transforms raw data and code into actionable insights, providing clear direction for decision-makers and ensuring that the value and implications of your analysis are readily accessible and understandable.

# Content review Module 5: Practical Applications I

Best practices around Jupyter Notebooks

- Please ensure to 'Clear All Outputs' and subsequently 'Run All Cells' before submission. This process confirms that all code cells are executed in the correct sequence and verifies that no cells are missing or misordered, thereby maintaining the integrity and full functionality of the notebook.
- When submitting homework on GitHub, please upload the `.ipynb` file unless otherwise instructed
- To make sure images work, check on **nbviewer**: https://nbviewer.org/
- Check that the link you're submitting works

# AGENDA

- ✅ Assignment: Practical Applications I Feedback QA
- Required activities for Module 6
- Content review Module 6: Data Clustering and PCA
- Questions

# Required Activities for Module 6

- Knowledge Check 6.1: SVD and PCA
- Knowledge Check 6.2: Interpretation of PCA
- Codio Assignment 6.1: Analyzing the Results of PCA
- Knowledge Check 6.3: Clustering and k-means
- Knowledge Check 6.4: Applying k-means, k-means++, and DBSCAN using Scikit-Learn
- Codio Assignment 6.2: Conducting the k-means Algorithm in Python
- Codio Assignment 6.3: Running PCA with Clustering
- Knowledge Check 6.5: DBSCAN
- Codio Assignment 6.4: Running DBSCAN
- Capstone Assignment 6.1: Draft the Problem Statement

# Required Activities for Module 6

- Knowledge Check 6.1: SVD and PCA
- Knowledge Check 6.2: Interpretation of PCA
- Codio Assignment 6.1: Analyzing the Results of PCA
- Knowledge Check 6.3: Clustering and k-means
- Knowledge Check 6.4: Applying k-means, k-means++, and DBSCAN using Scikit-Learn
- Codio Assignment 6.2: Conducting the k-means Algorithm in Python
- Codio Assignment 6.3: Running PCA with Clustering
- Knowledge Check 6.5: DBSCAN
- Codio Assignment 6.4: Running DBSCAN
- **Capstone Assignment 6.1: Draft the Problem Statement**

# CAPSTONE PROJECT

- **Roadmap**
  - **Module 6**
    Draft the Problem Statement
  - **Module 11**
    Define your problem statement and develop a prospectus of the project
  - **Modules 12 to 15**
    First 1:1 With Your Learning Facilitator
  - **Module 17**
    Problem Statement
  - **Module 20**
    Initial Report, EDA, build candidate models
  - **Modules 21 to 23**
    Second 1:1 With Your Learning Facilitator
  - **Module 24**
    Final Analysis and Report

# Capstone Assignment 6.1: Draft the Problem Statement

- A general overview of the question you will be asking/solving for (1-2 sentences)
- The data you think you'll need to answer this question (no official data is needed at this time, just ideas of what type of data will help you)
- List 1–3 techniques you might use to answer this question based on what you've learned so far

Canvas link: https://classroom.emeritus.org/courses/9414/assignments/259205?module_item_id=1930633

The capstone project will be the opportunity for you to create a data science project and apply what you will learn in the program to a problem that is of your interest. One of the objectives of the capstone project is that you will end up with a project that you can include in your portfolio to showcase your data science skills.

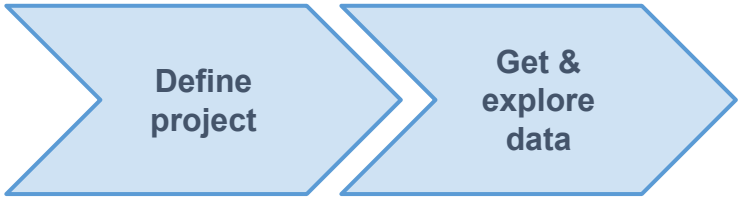
# The Data Science Lifecycle


**Define project**

- Specify business problem
- Acquire domain knowledge

- A general overview of the question you will be asking/solving for (1-2 sentences)

# The Data Science Lifecycle




## Define project

- Specify business problem
- Acquire domain knowledge

## Get and explore data

- Find appropriate data
- Exploratory Data Analysis
- Clean and pre-process data
- Feature engineering

- A general overview of the question you will be asking/solving for (1-2 sentences)
- The data you think you'll need to answer this question (no official data is needed at this time, just ideas of what type of data will help you)

# Where to get data?

- Here are some ideas…

**Open data repositories:**

OpenML.org (https://openml.org)
Kaggle.com (https://kaggle.com/datasets)
PapersWithCode.com (https://paperswithcode.com/datasets)
UC Irvine Machine Learning Repository (https://archive.ics.uci.edu/ml)
Amazon's AWS datasets (https://registry.opendata.aws)
TensorFlow datasets (https://tensorflow.org/datasets)
Google's data search engine: (https://datasetsearch.research.google.com/)

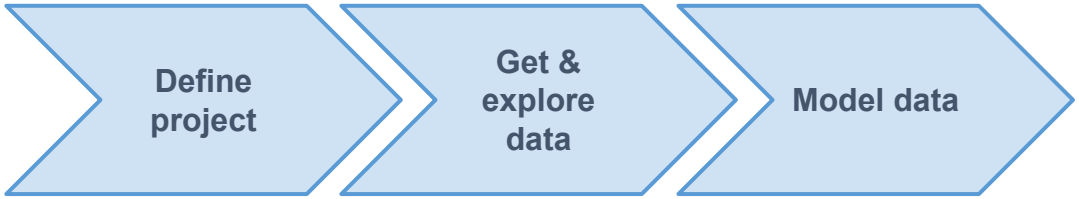
**Meta portals and other pages listing datasets:**

DataPortals.org (https://dataportals.org/)
OpenDataMonitor.eu (https://opendatamonitor.eu/frontend/web/index.php?r=dashboard%2Findex)
Wikipedia's list of machine learning datasets
(https://en.wikipedia.org/wiki/List_of_datasets_for_machine-learning_research)
Quora's list (https://www.quora.com/Where-can-I-find-large-datasets-open-to-the-public)
Reddit's dataset (https://www.reddit.com/r/datasets)
GitHub * (https://github.com/)

**Location-specific:**

San Francisco Open Data (https://datasf.org/opendata/)
NYC Open Data (https://opendata.cityofnewyork.us/)
You can also google "open data + location" to get data about your desired location

# The Data Science Lifecycle




## Define project

- Specify business problem
- Acquire domain knowledge

## Get and explore data

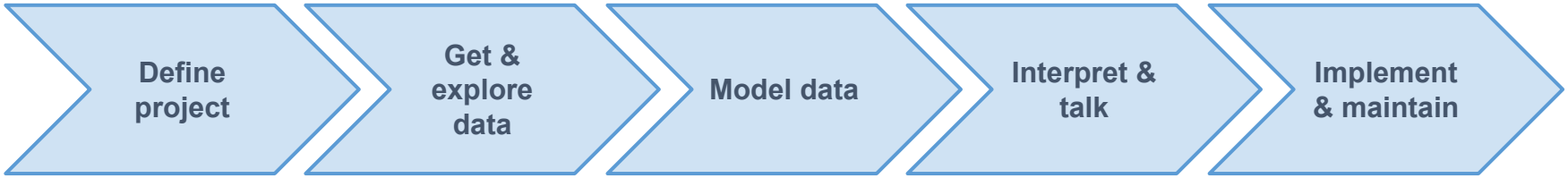
- Find appropriate data
- Exploratory Data Analysis
- Clean and pre-process data
- Feature engineering

## Model data

- Determine ML task
- Build candidate models
- Select model based on performance metrics

- A general overview of the question you will be asking/solving for (1-2 sentences)
- The data you think you'll need to answer this question (no official data is needed at this time, just ideas of what type of data will help you)
- List 1–3 techniques you might use to answer this question based on what you've learned so far

# The Machine Learning pipeline




### Define project

- Specify business problem
- Acquire domain knowledge

### Get and explore data

- Find appropriate data
- Exploratory Data Analysis
- Clean and pre-process data
- Feature engineering

### Model data

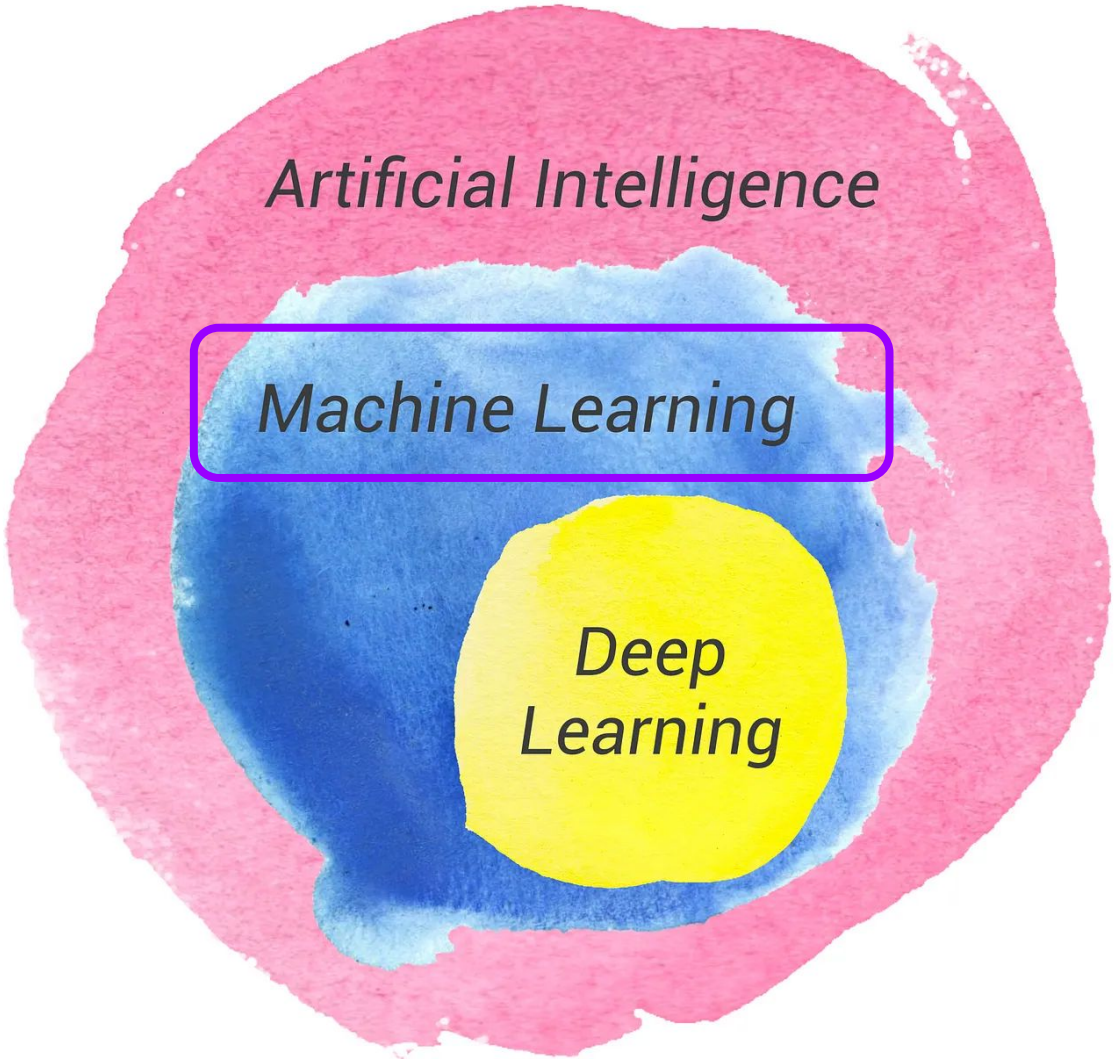
- Determine ML task
- Build candidate models
- Select model based on performance metrics

### Interpret & talk

- Interpret model
- Communicate model insights

### Implement & maintain

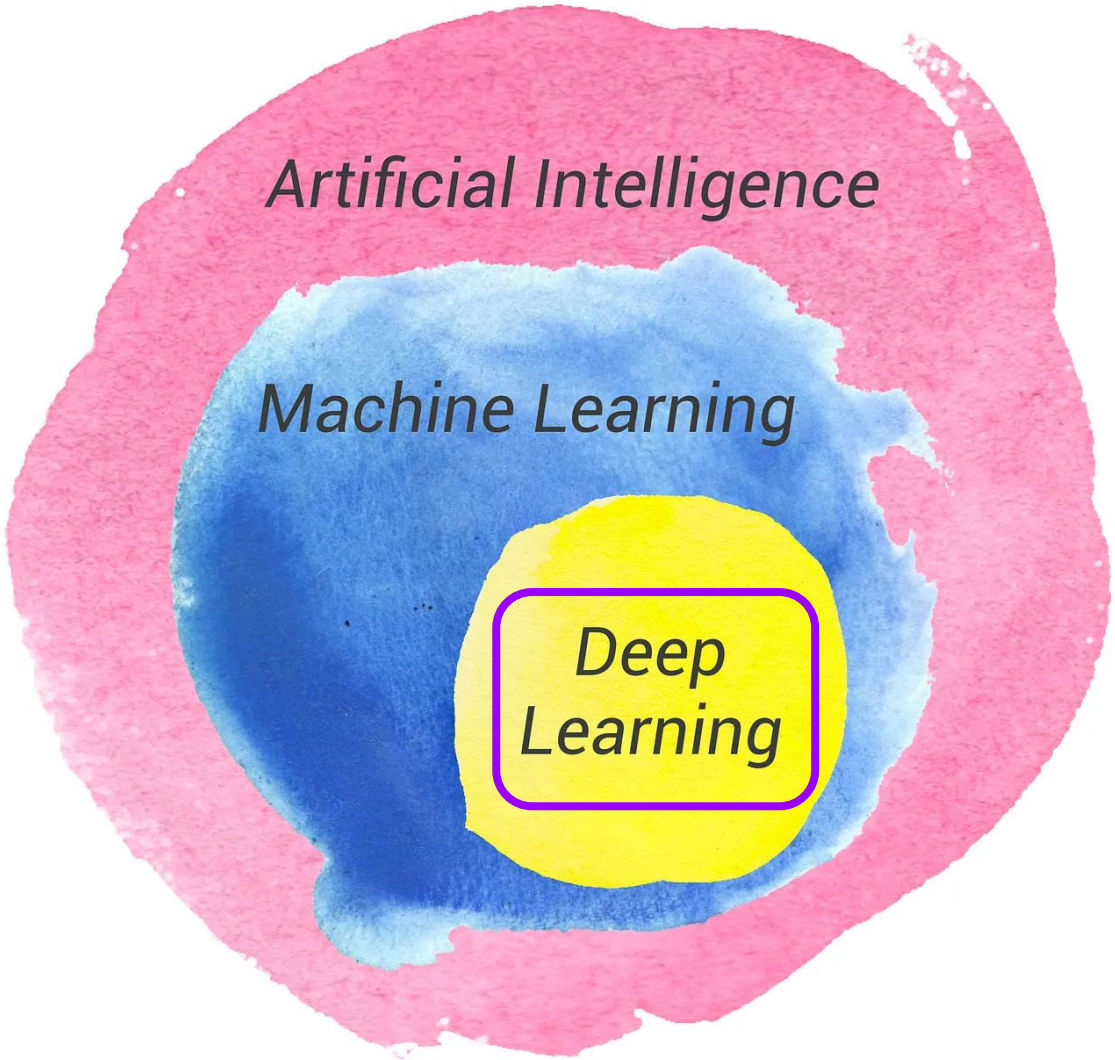
- Set up function to predict on new data
- Document process
- Monitor and maintain model




## Machine Learning

ML is a subset of AI that focuses on creating systems that can learn and improve from experience or data, without being explicitly programmed for every task.

**Scope:** Moderate. It includes various techniques such as regression, classification, clustering, and ensemble models.

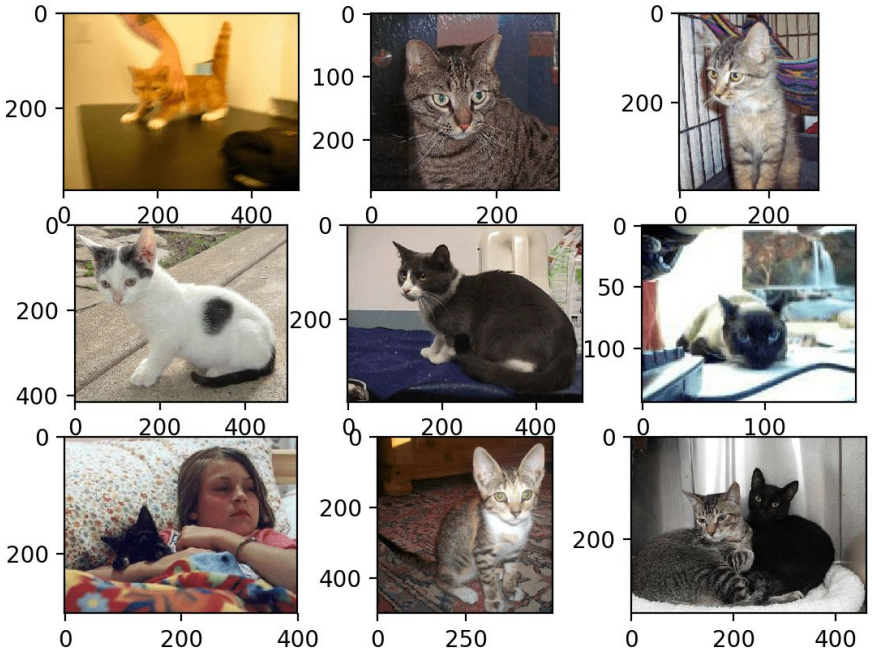



## Deep Learning

DL is a specialized type of machine learning that uses neural networks with many layers to analyze vast amounts of data and learn complex patterns, often achieving results comparable to human performance in areas like image recognition or language translation.

**Scope:** Narrow. DL is a specific, yet powerful, form of machine learning.

# What is Machine Learning?

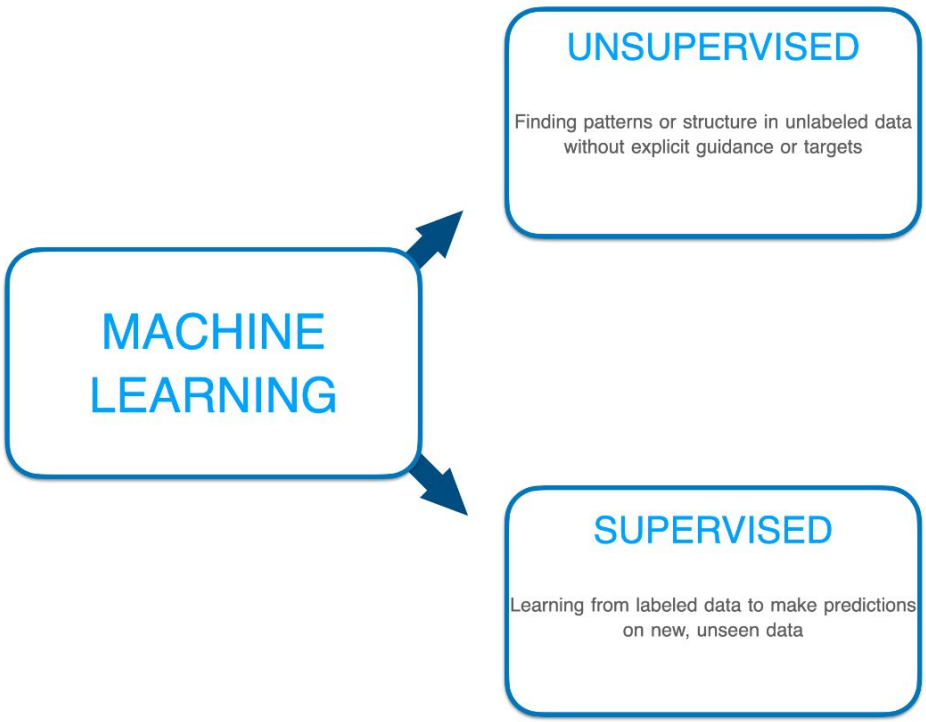


In machine learning, instead of explicitly programming a computer with specific instructions to perform a task, we provide it with large amounts of data and allow it to learn how to **generalize** from that data.

# The Machine Learning landscape

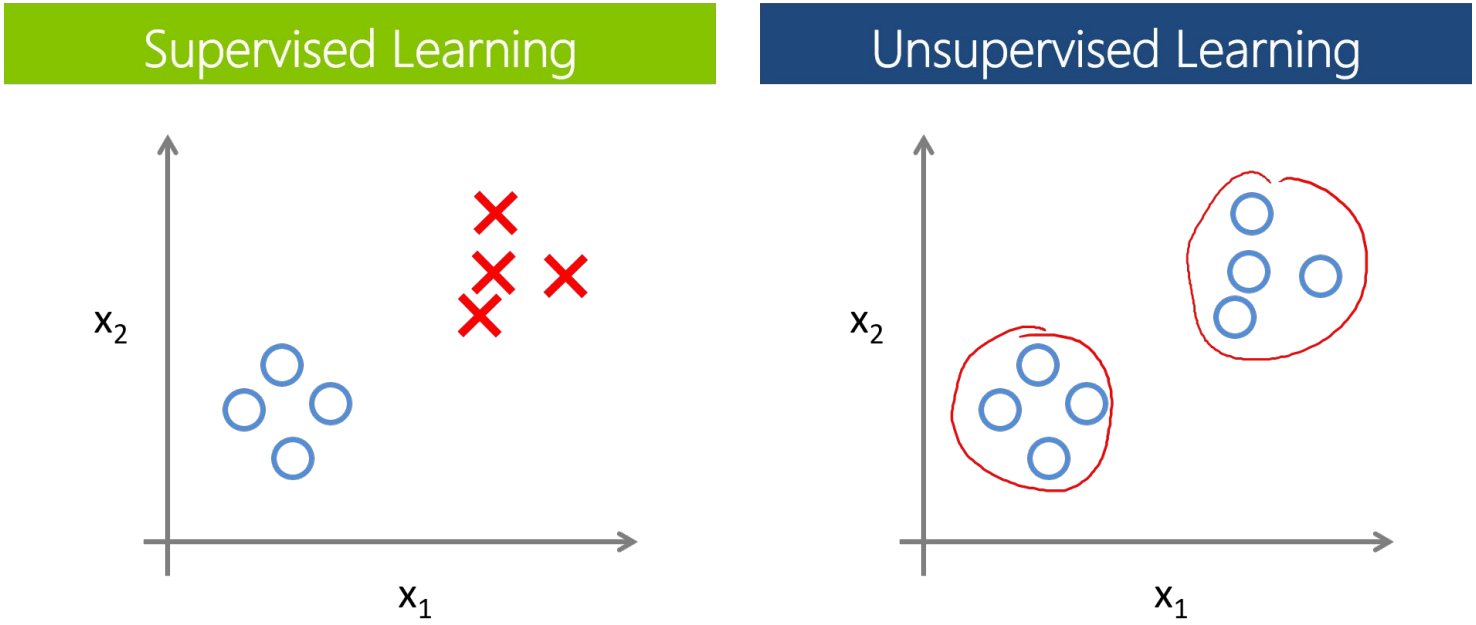
Which tool would you use to hit a nail?

# First Question: Do we have labels?



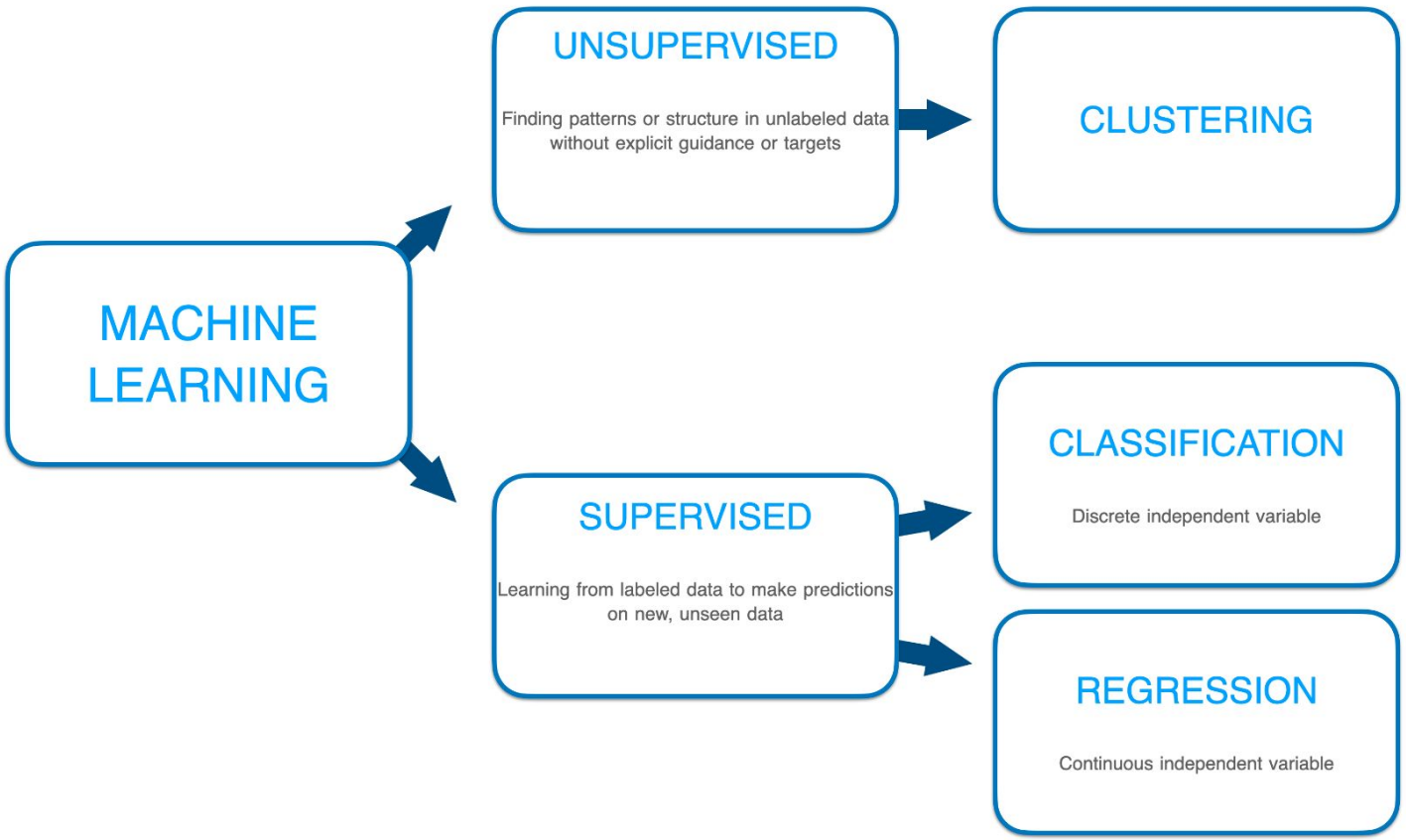
# The Machine Learning landscape

## Supervised learning vs Unsupervised learning



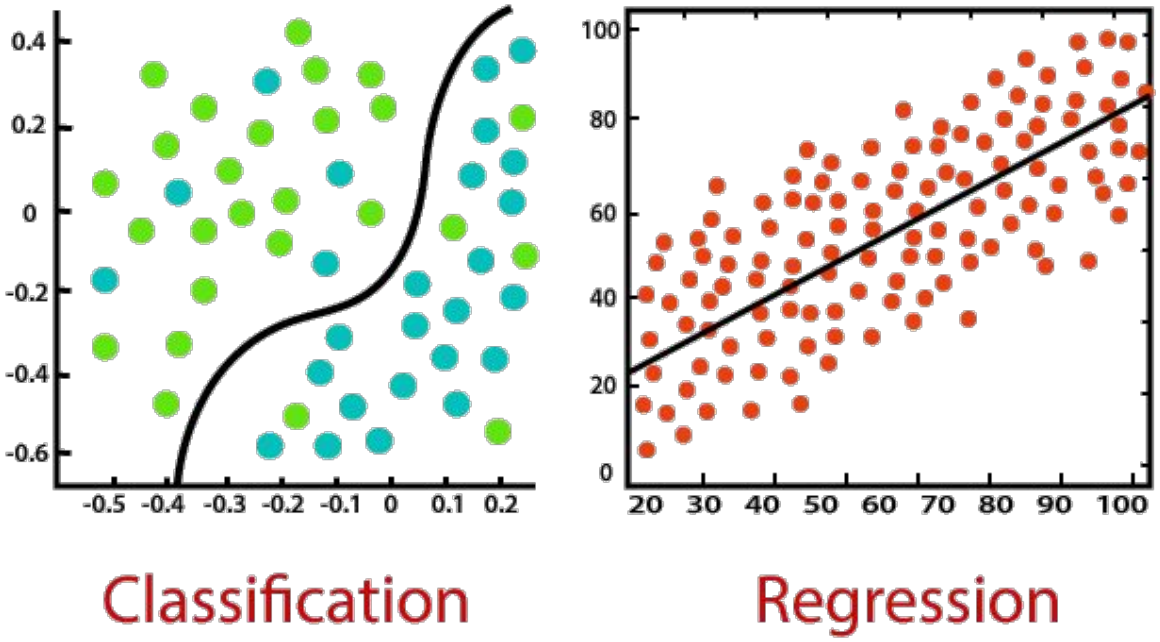

- **Supervised learning:** Problems with labels. Its aim is to predict data based on the labeled information.
- **Unsupervised learning:** Problems without labels. Its goal is to uncover patterns, structures, and relationships.

# First Question: Do we have labels?
# Second Question: Are our labels categorical or numerical?



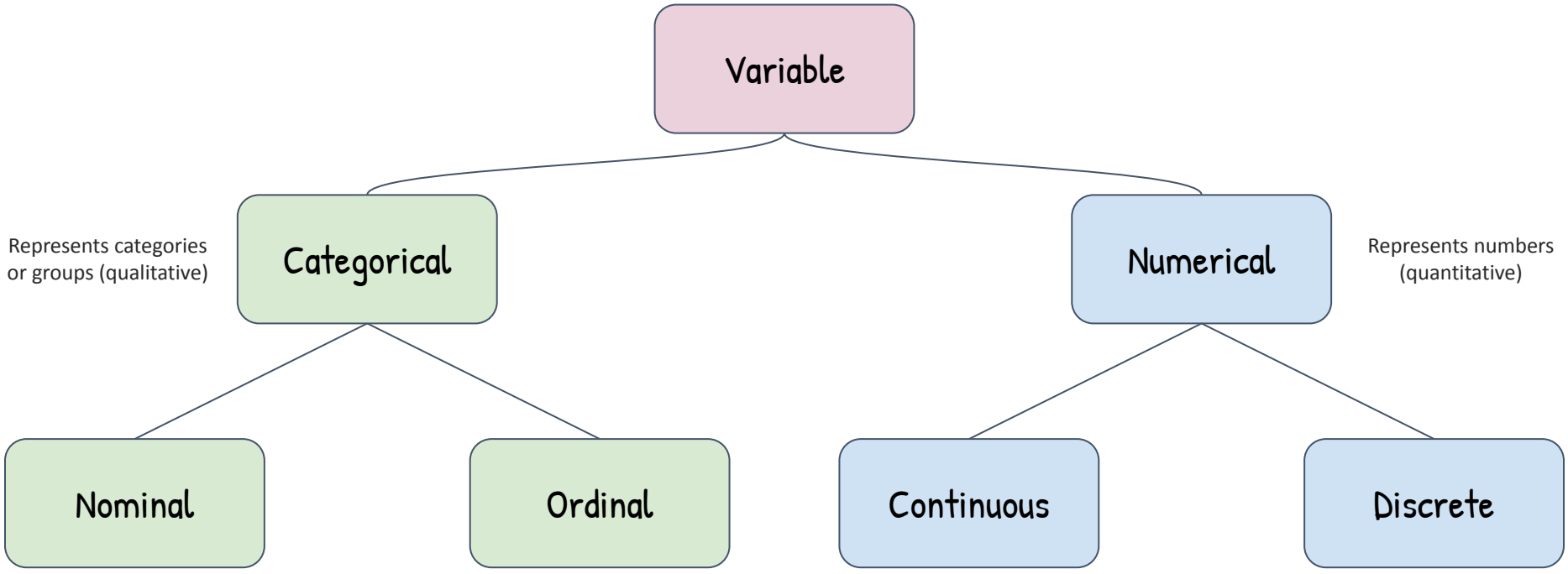
# The Machine Learning landscape

## Regression vs Classification



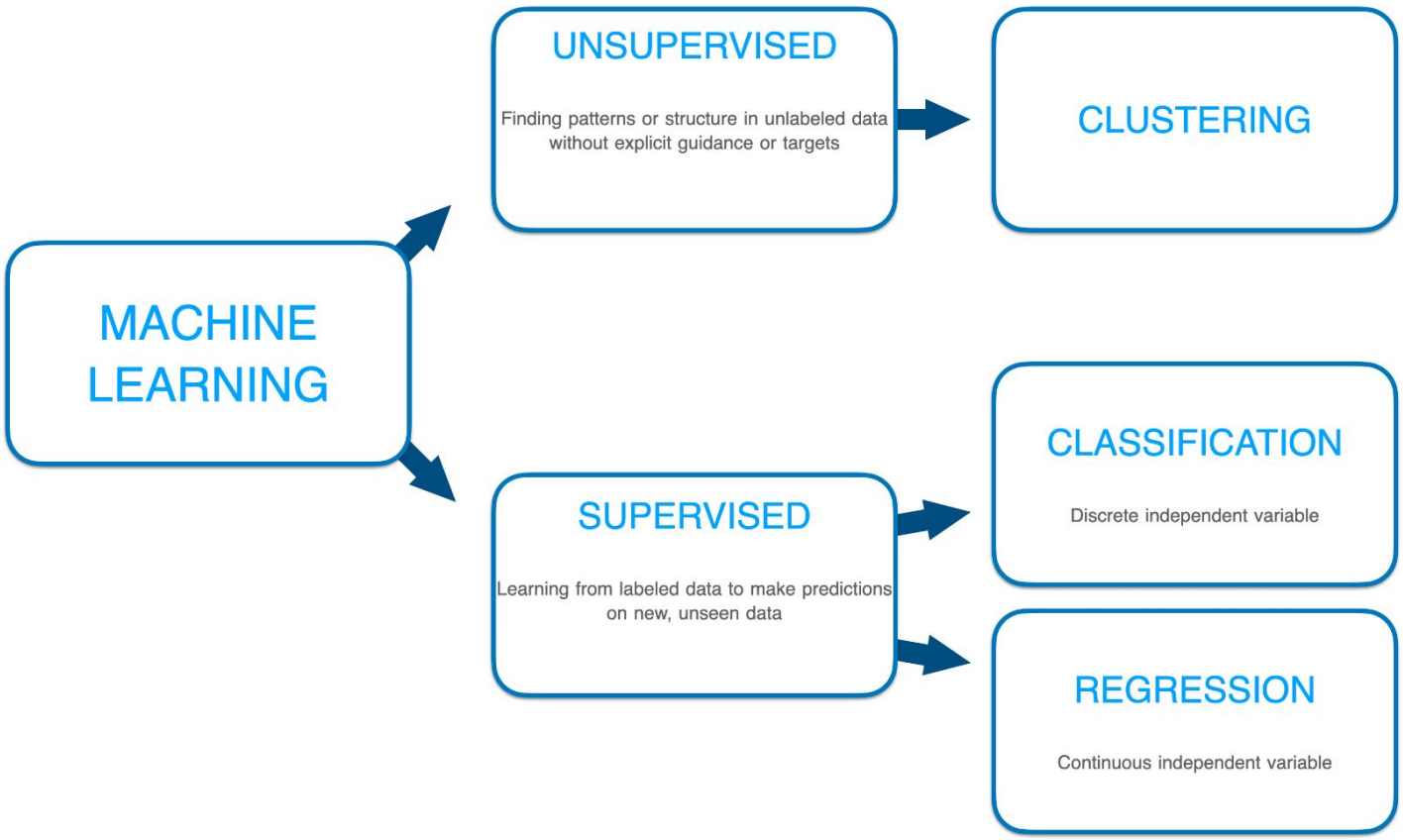

- **Regression:** Quantitative (continuous/numerical) target variable
- **Classification:** Qualitative (discrete/categorical) target variable

# A variable based on its type can be categorical or numerical




| **Example** | **Example** | **Example** | **Example** |
|---|---|---|---|
| • Eye color (Blue/Green/Brown) | • Education level (Bachelor's/Master's/PhD) | • Weight in pounds | • Number of cows a farmer owns |
| As a target variable, in ML<br>It's a **Classification** model | As a target variable, in ML<br>It's a **Regression or Classification** model | As a target variable, in ML<br>It's a **Regression** model | As a target variable, in ML<br>It's a **Regression or Classification** model |

# First Question: Do we have labels?
# Second Question: Are our labels categorical or numerical?

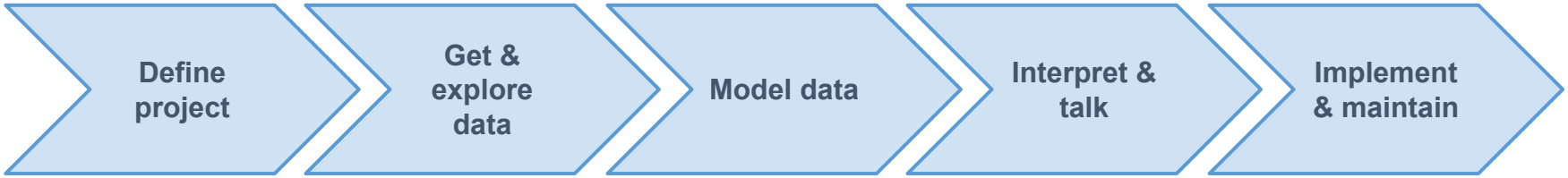



**Clustering:**
K-Means Clustering
Hierarchical Clustering
DBSCAN
Gaussian Mixture Models (GMM)
Spectral Clustering

**Classification:**
Logistic Regression
Decision Trees
Random Forest
Support Vector Machines (SVM)
Naive Bayes
K-Nearest Neighbors (KNN)
Gradient Boosting (XGBoost, GBM, etc.)
AdaBoost (Adaptive Boosting)
Multi-Layer Perceptron (MLP)
Convolutional Neural Networks (for images)
Recurrent Neural Networks (for sequences)

**Regression:**
Linear Regression
Polynomial Regression
Ridge Regression
Lasso Regression
Elastic Net Regression
Support Vector Regression (SVR)
Decision Trees
Random Forest
K-Nearest Neighbors (KNN)
Support Vector Machines (SVM)
Gradient Boosting (XGBoost, GBM, etc.)
Multi-Layer Perceptron (MLP)
Recurrent Neural Networks (for time series)

# The Data Science Lifecycle




### Define project
- Specify business problem
- Acquire domain knowledge

### Get and explore data
- Find appropriate data
- Exploratory Data Analysis
- Clean and pre-process data
- Feature engineering

### Model data
- Determine ML task
- Build candidate models
- Select model based on performance metrics

### Interpret & talk
- Interpret model
- Communicate model insights

### Implement & maintain
- Set up function to predict on new data
- Document process
- Monitor and maintain model

# Capstone Assignment 6.1: Draft the Problem Statement

- A general overview of the question you will be asking/solving for (1-2 sentences)
- The data you think you'll need to answer this question (no official data is needed at this time, just ideas of what type of data will help you)
- List 1–3 techniques you might use to answer this question based on what you've learned so far

Canvas link: https://classroom.emeritus.org/courses/9414/assignments/259205?module_item_id=1930633

The capstone project will be the opportunity for you to create a data science project and apply what you will learn in the program to a problem that is of your interest. One of the objectives of the capstone project is that you will end up with a project that you can include in your portfolio to showcase your data science skills.

# CAPSTONE PROJECT

- If you want to go the extra mile…

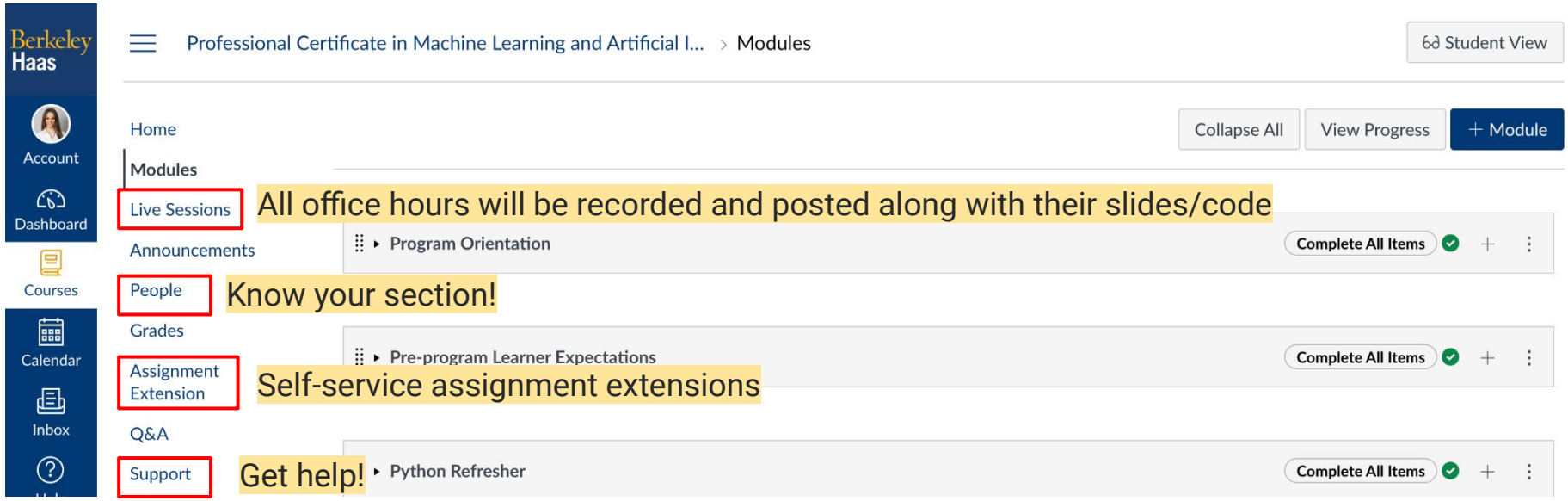
Dash by plotly





github:pages

# CANVAS


Berkeley Haas
Account
Dashboard
Courses
Calendar
Inbox
Professional Certificate in Machine Learning and Artificial I... > Modules
Student View
Home
Modules
Live Sessions
Announcements
People
Grades
Assignment Extension
Q&A
Support
Collapse All
View Progress
+ Module
All office hours will be recorded and posted along with their slides/code
Program Orientation
Complete All Items
Know your section!
Pre-program Learner Expectations
Complete All Items
Self-service assignment extensions
Get help!
Python Refresher
Complete All Items

# AGENDA

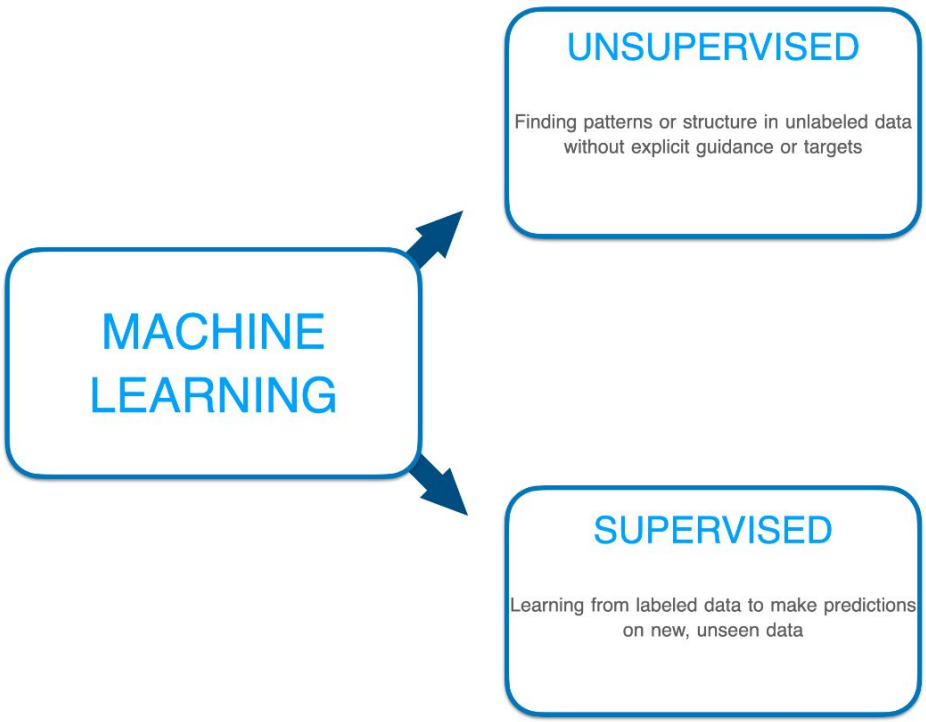
- ✅ Assignment: Practical Applications I Feedback QA
- ✅ Required activities for Module 6
- Content review Module 6: Data Clustering and PCA
- Questions

# Module 6: Data Clustering and PCA

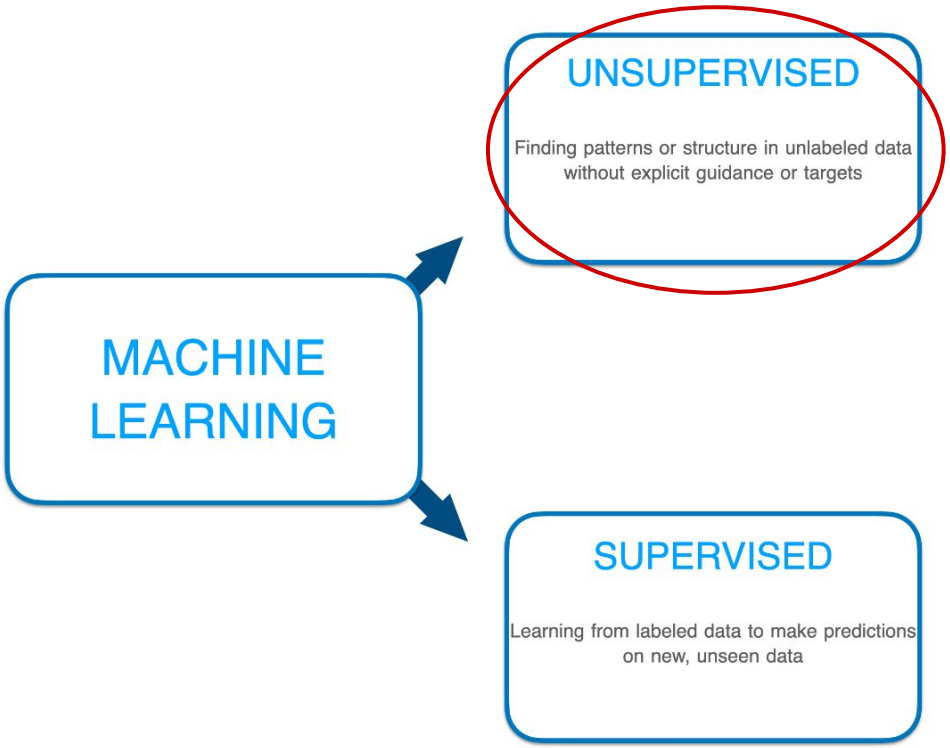
- Unsupervised Learning
  - Clustering
    - K-Means
  - Dimensionality Reduction
    - PCA

# Unsupervised learning

# First Question: Do we have labels?



# First Question: Do we have labels?



# Unsupervised Learning

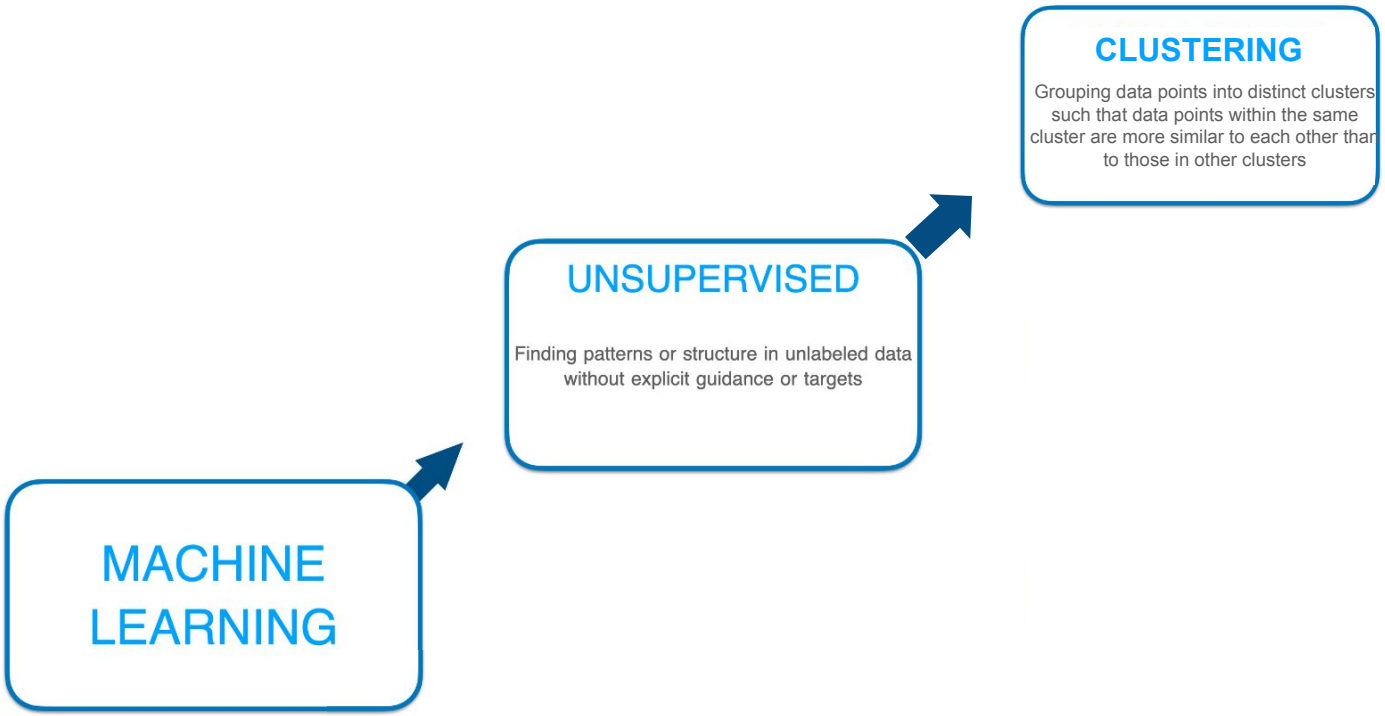

- Algorithm learns from unlabelled data
  - We don’t need (*have*) labels
  - **Goal**: Find underlying patterns, structures, or relationships in the data

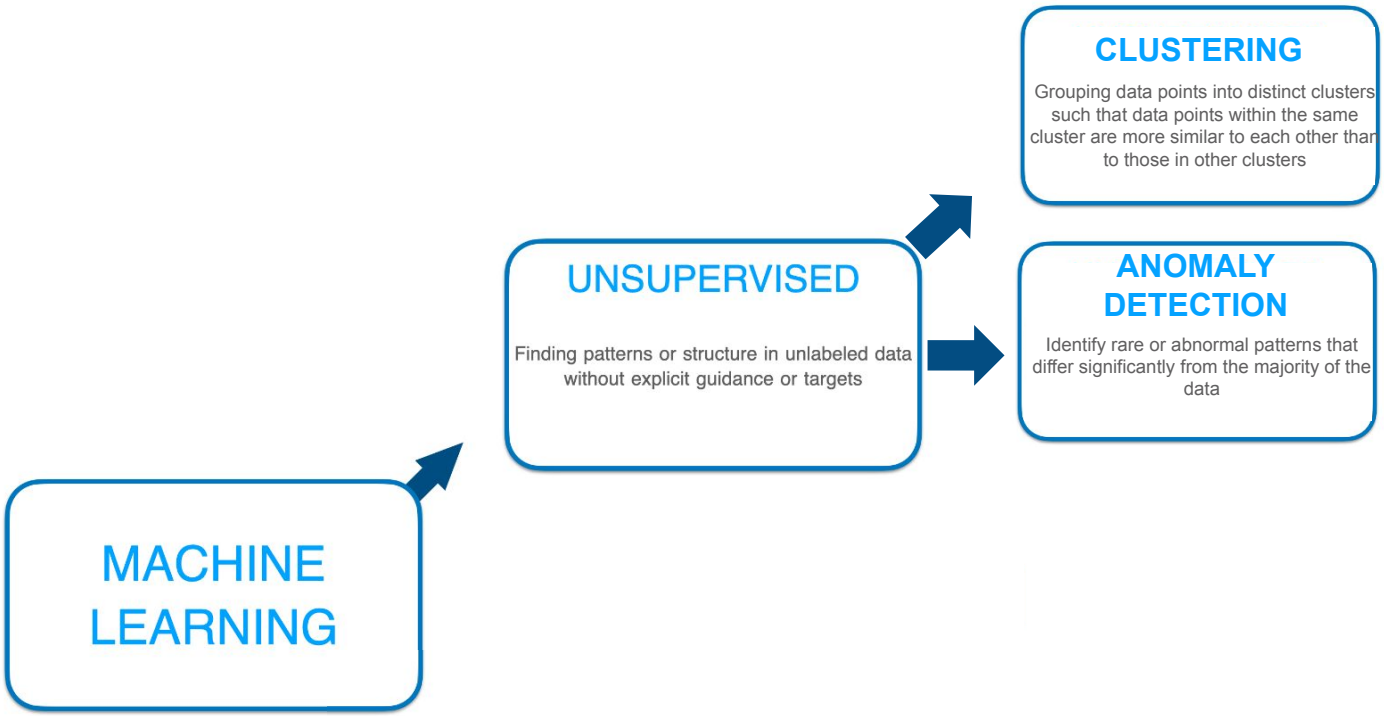
- Common applications
  - Clustering
  - Anomaly Detection
  - Dimensionality Reduction
  - and more…

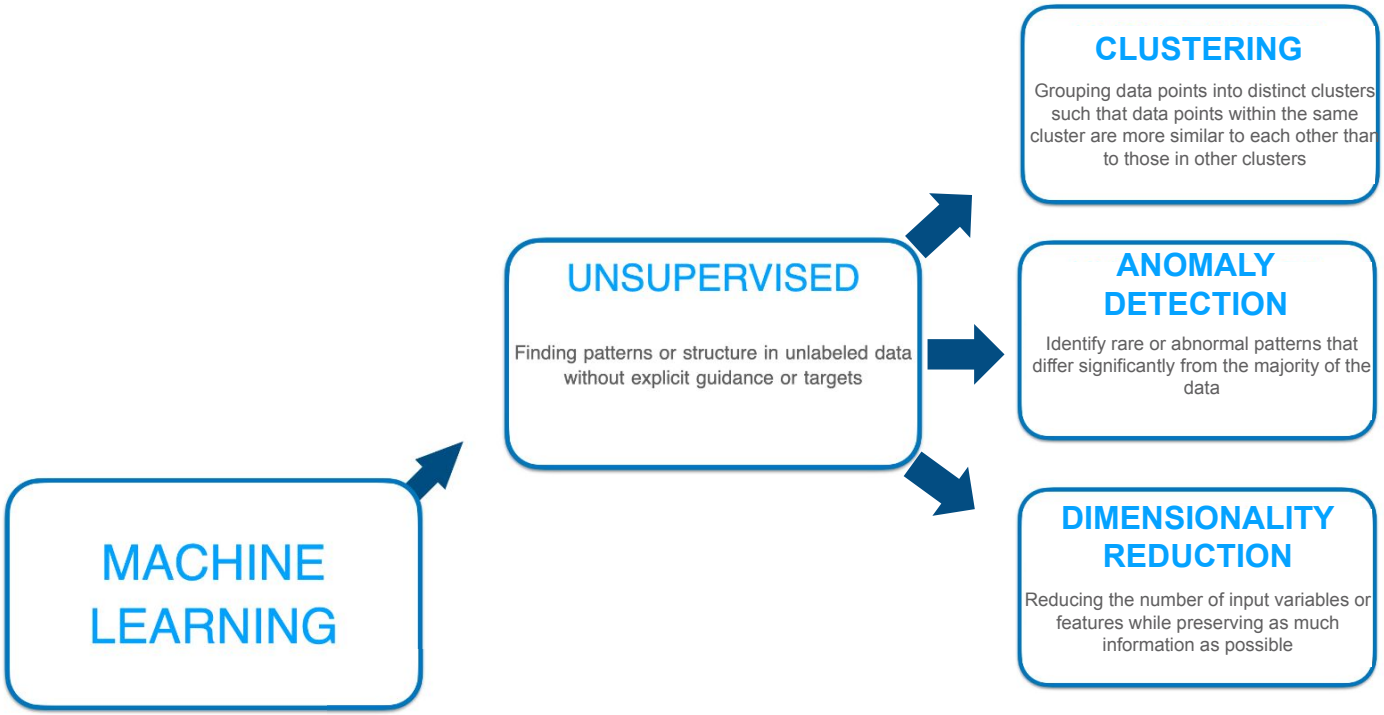
# Unsupervised Learning

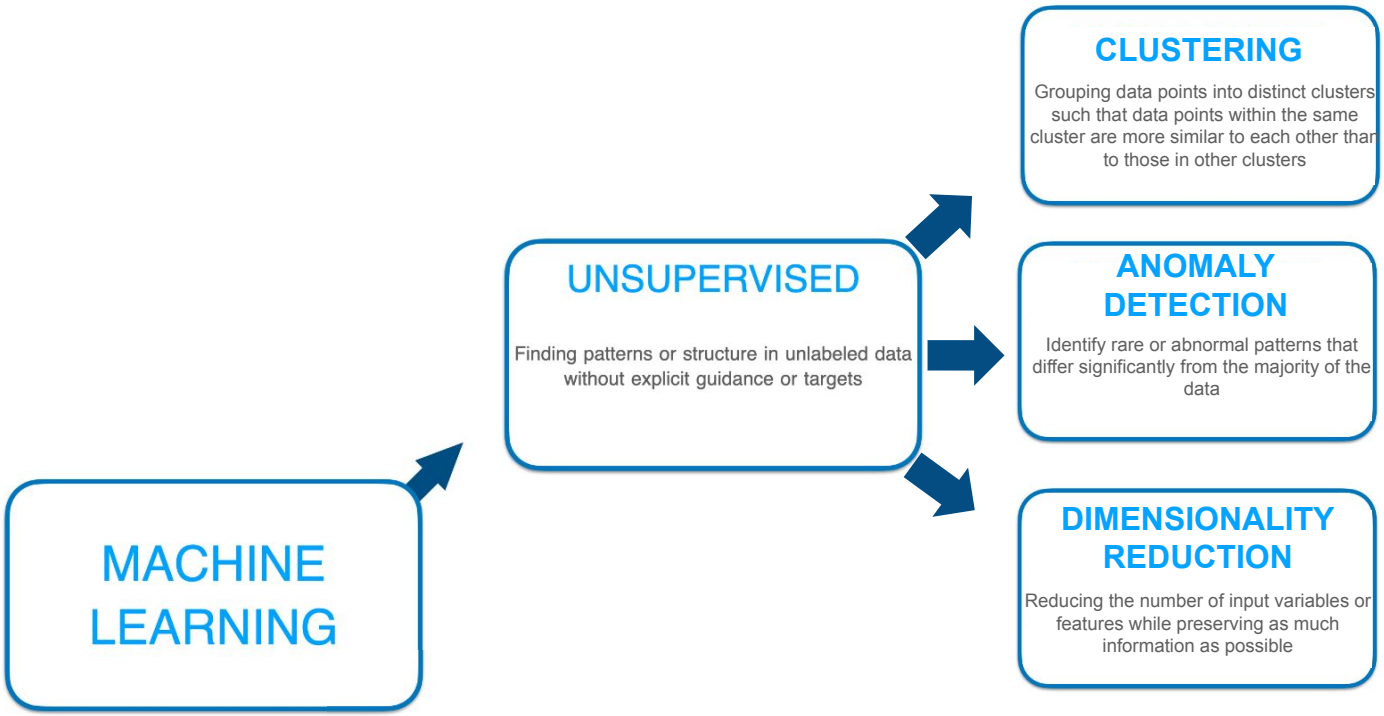

- **Benefits**
  - Discover hidden patterns in the data that might not be evident or intuitive
  - Much of the data in the real world is unlabelled
  - Methods like PCA can reduce the number of features, making other machine learning tasks more efficient

- **Challenges**
  - Without a clear metric, it's often harder to evaluate the results
  - The results, like clusters, might be hard to interpret. It's often not immediately clear what the clusters represent in terms of underlying patterns or properties


MACHINE LEARNING
UNSUPERVISED
Finding patterns or structure in unlabeled data without explicit guidance or targets
CLUSTERING
Grouping data points into distinct clusters such that data points within the same cluster are more similar to each other than to those in other clusters


MACHINE LEARNING
UNSUPERVISED
Finding patterns or structure in unlabeled data without explicit guidance or targets
CLUSTERING
Grouping data points into distinct clusters such that data points within the same cluster are more similar to each other than to those in other clusters
ANOMALY DETECTION
Identify rare or abnormal patterns that differ significantly from the majority of the data

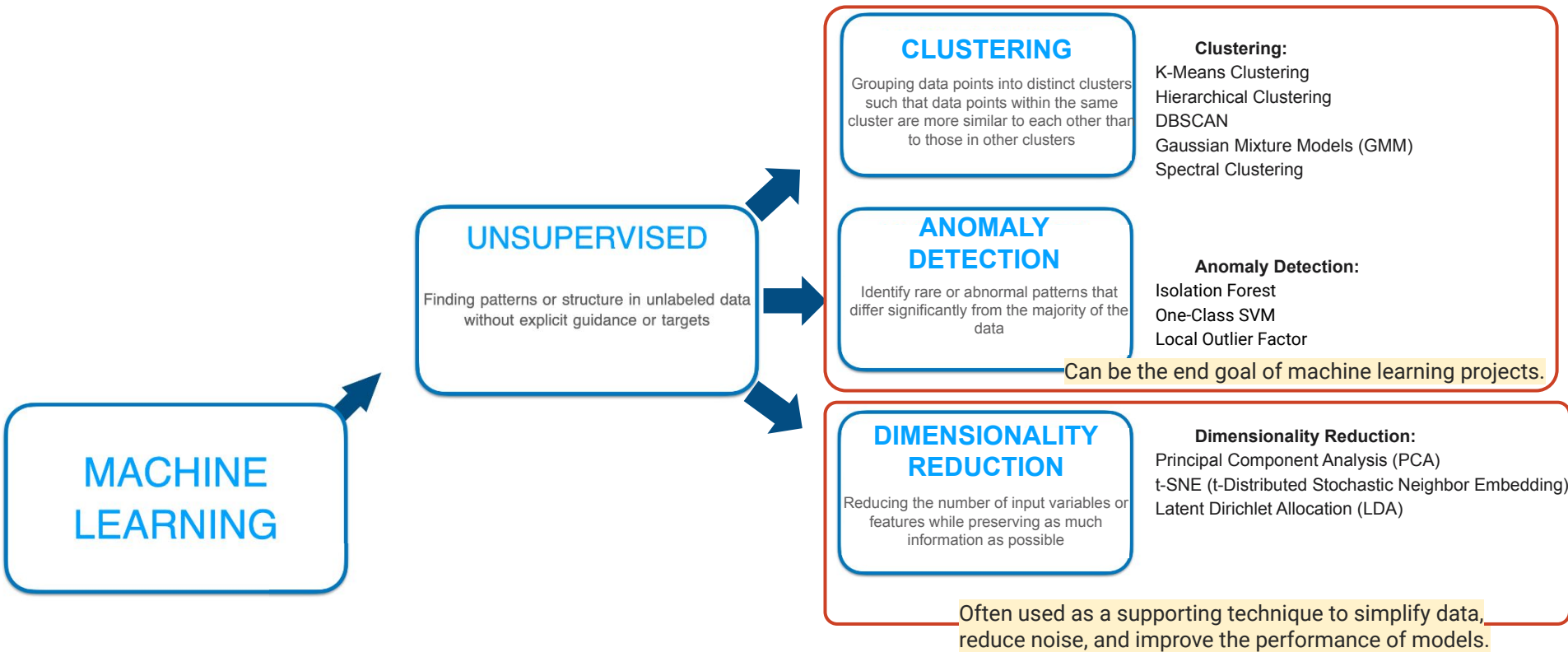

MACHINE LEARNING
UNSUPERVISED
Finding patterns or structure in unlabeled data without explicit guidance or targets
CLUSTERING
Grouping data points into distinct clusters such that data points within the same cluster are more similar to each other than to those in other clusters
ANOMALY DETECTION
Identify rare or abnormal patterns that differ significantly from the majority of the data
DIMENSIONALITY REDUCTION
Reducing the number of input variables or features while preserving as much information as possible

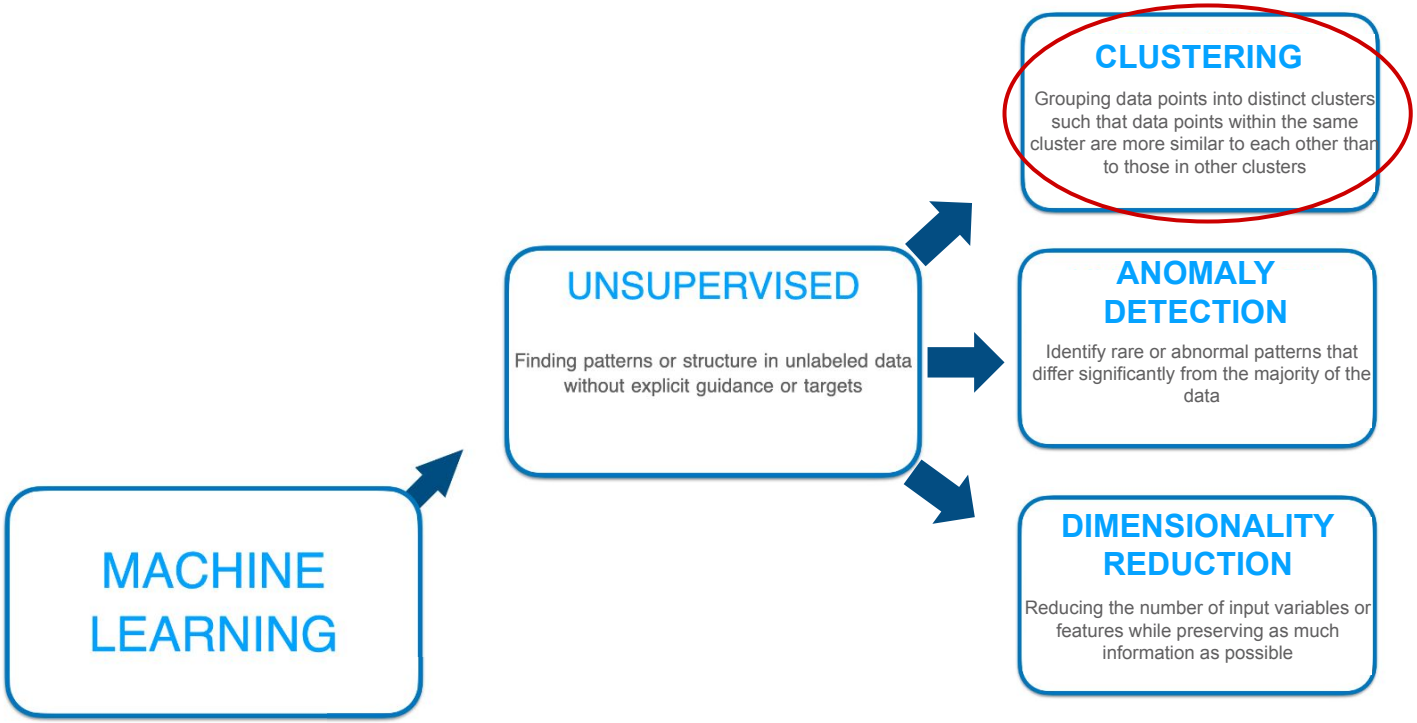



**Clustering:**
K-Means Clustering
Hierarchical Clustering
DBSCAN
Gaussian Mixture Models (GMM)
Spectral Clustering

**Anomaly Detection:**
Isolation Forest
One-Class SVM
Local Outlier Factor

**Dimensionality Reduction:**
Principal Component Analysis (PCA)
t-SNE (t-Distributed Stochastic Neighbor Embedding)
Latent Dirichlet Allocation (LDA)


MACHINE LEARNING
UNSUPERVISED
Finding patterns or structure in unlabeled data without explicit guidance or targets
CLUSTERING
Grouping data points into distinct clusters such that data points within the same cluster are more similar to each other than to those in other clusters
Clustering:
K-Means Clustering
Hierarchical Clustering
DBSCAN
Gaussian Mixture Models (GMM)
Spectral Clustering
ANOMALY DETECTION
Identify rare or abnormal patterns that differ significantly from the majority of the data
Anomaly Detection:
Isolation Forest
One-Class SVM
Local Outlier Factor
Can be the end goal of machine learning projects.
DIMENSIONALITY REDUCTION
Reducing the number of input variables or features while preserving as much information as possible
Dimensionality Reduction:
Principal Component Analysis (PCA)
t-SNE (t-Distributed Stochastic Neighbor Embedding)
Latent Dirichlet Allocation (LDA)
Often used as a supporting technique to simplify data, reduce noise, and improve the performance of models.

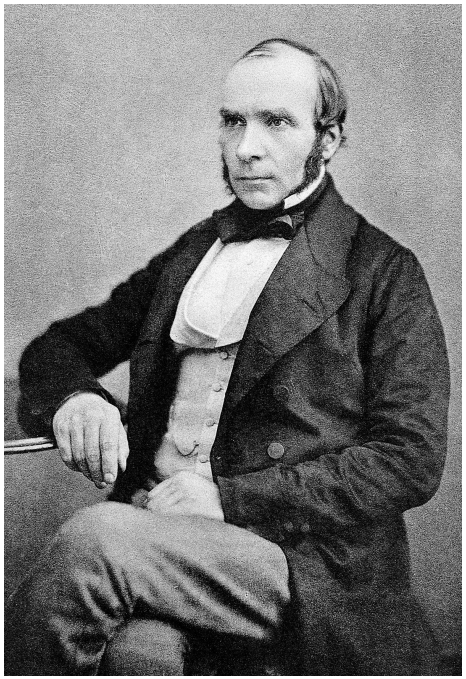

MACHINE LEARNING
UNSUPERVISED
Finding patterns or structure in unlabeled data without explicit guidance or targets
CLUSTERING
Grouping data points into distinct clusters such that data points within the same cluster are more similar to each other than to those in other clusters
ANOMALY DETECTION
Identify rare or abnormal patterns that differ significantly from the majority of the data
DIMENSIONALITY REDUCTION
Reducing the number of input variables or features while preserving as much information as possible

# Clustering algorithms

# First (?) clustering application

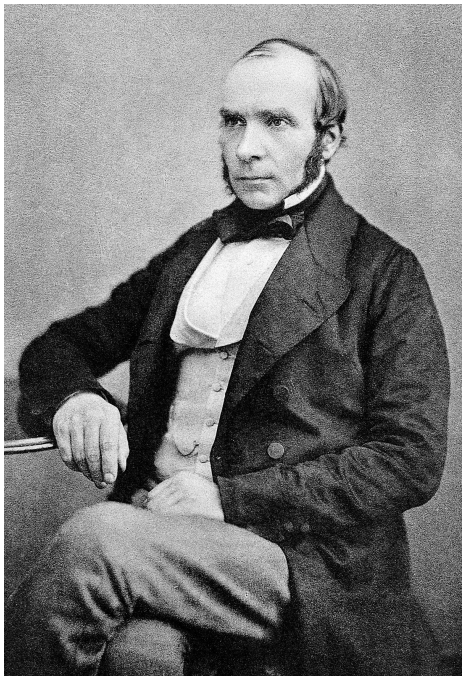
- In the 1850s, a London physician named John Snow plotted the location of cholera deaths on a map

# First (?) clustering application



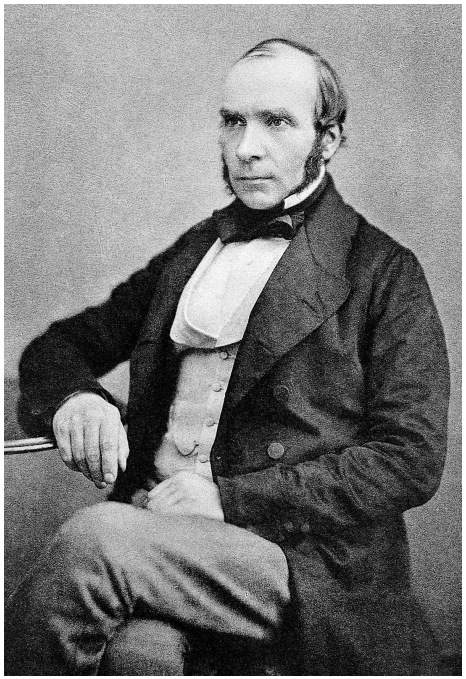
- In the 1850s, a London physician named John Snow plotted the location of cholera deaths on a map

# First (?) clustering application



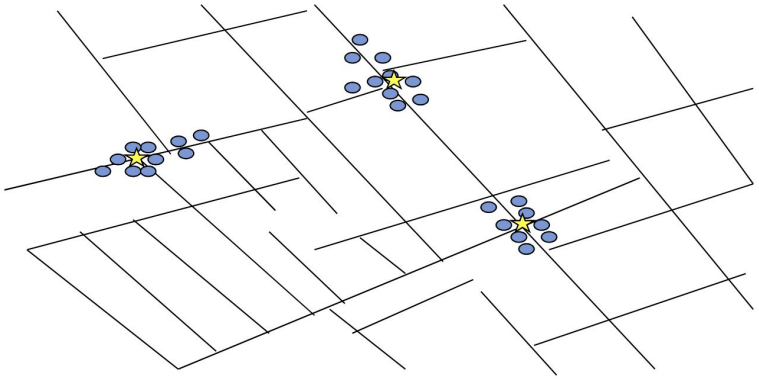
- In the 1850s, a London physician named John Snow plotted the location of cholera deaths on a map
- The locations showed that the cases were clustered near certain intersections where there were contaminated wells -- thus, exposing both the problem and the solution

# First (?) clustering application

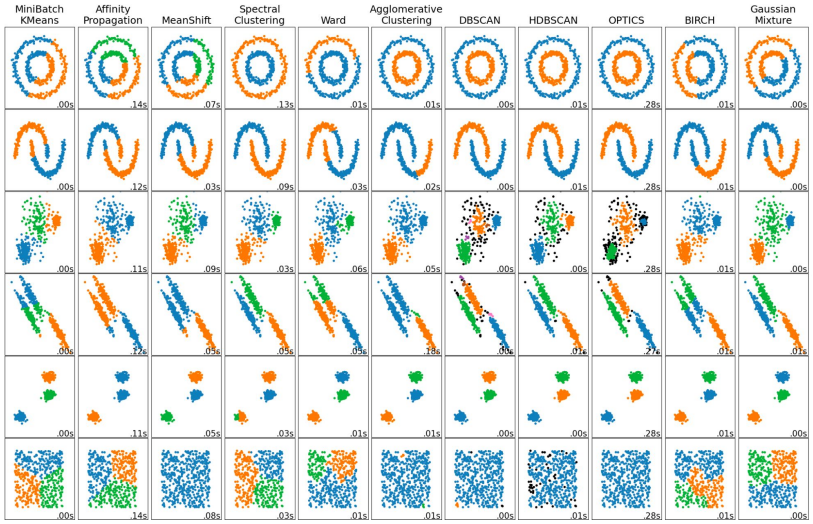


- In the 1850s, a London physician named John Snow plotted the location of cholera deaths on a map

- The locations showed that the cases were clustered near certain intersections where there were contaminated wells -- thus,



# Clustering algorithms



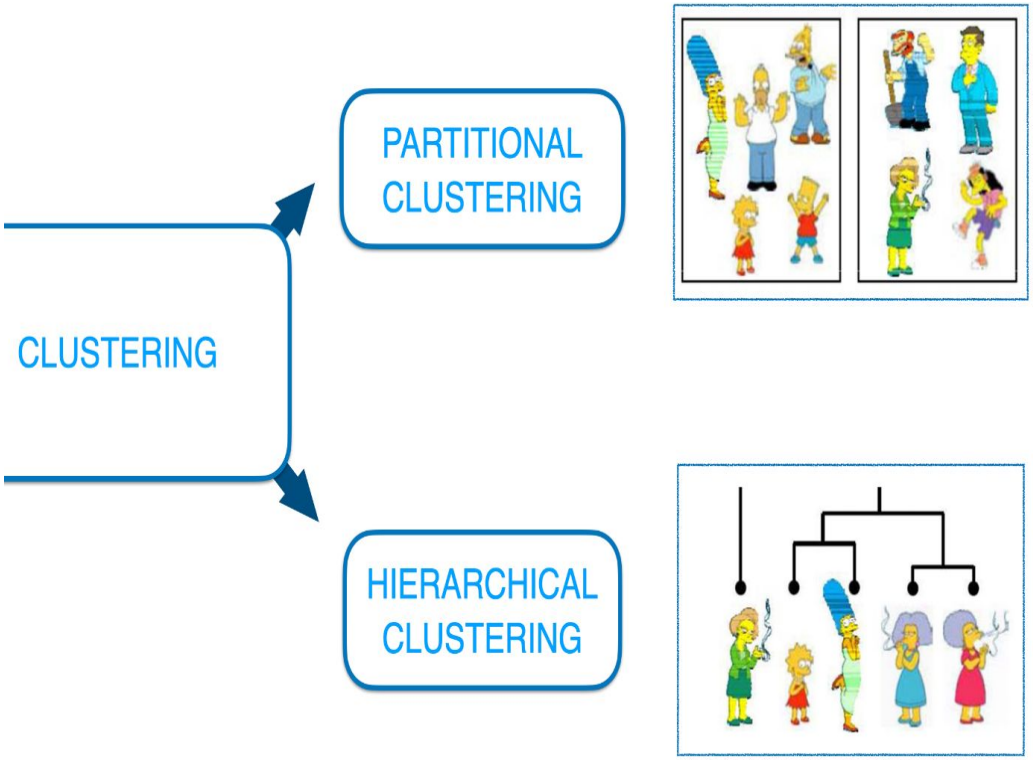

- **Goal**: Grouping data points into distinct clusters such that data points within the same cluster are more similar to each other than to those in other clusters

- https://scikit-learn.org/stable/modules/clustering.html

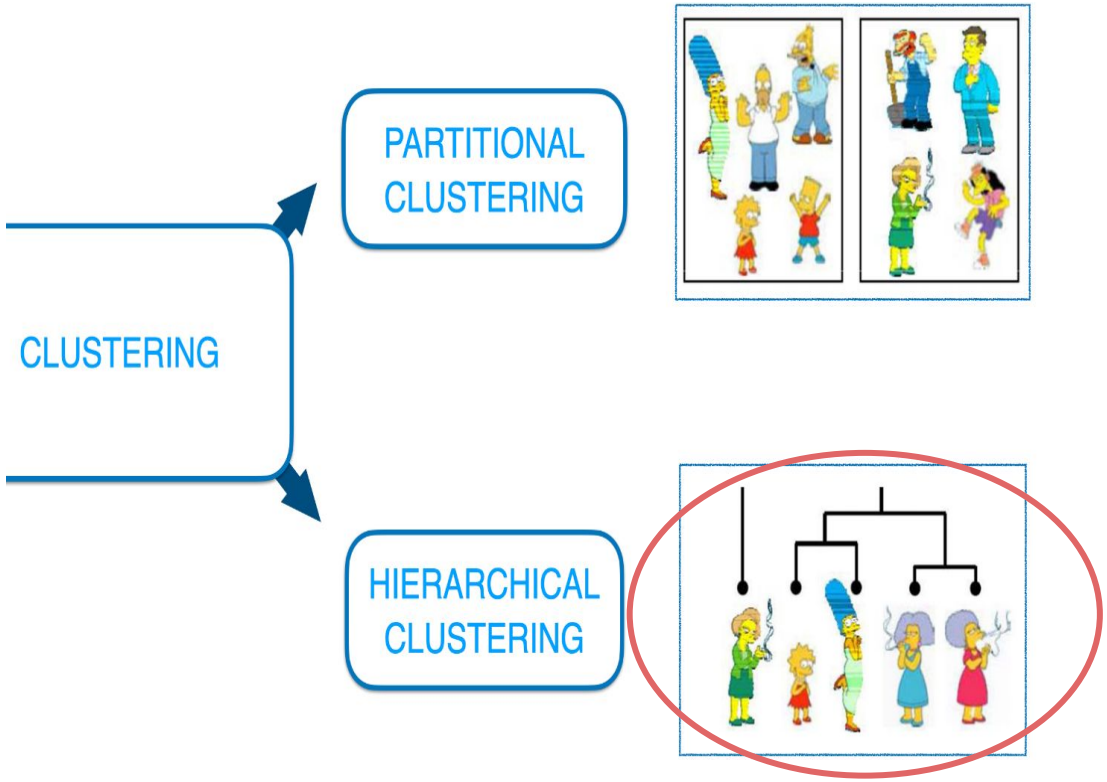
# Clustering algorithms

- **When to use them?** When we don't know what we are looking for
- ... but, **beware**, it can turn into gibberish! ⚠️
- The data set must have:
  - High intra-class similarity
  - Low inter-class similarity

# Clustering algorithms



# Clustering algorithms

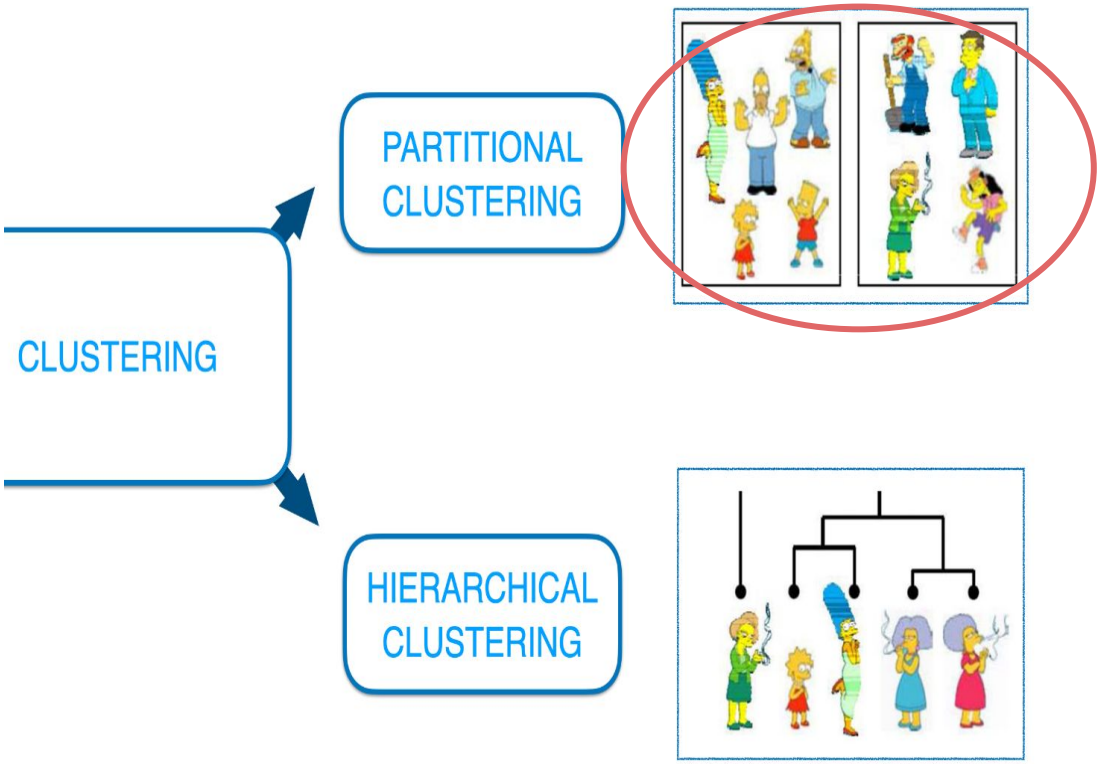


# Hierarchical clustering algorithms

- Hierarchical clustering creates a tree of clusters, offering a multilevel hierarchy where each level represents a particular granularity of clustering

- Common algorithms:
  - Agglomerative Hierarchical Clustering
  - Divisive Hierarchical Clustering
  - and more…

# Clustering algorithms



# Partitional clustering algorithms

- Partitional clustering, also known as non-hierarchical clustering, divides a dataset into a set of non-overlapping clusters. Unlike hierarchical clustering, partitional clustering doesn't set up a tree structure for clusters

- Common algorithms:
  - **K-Means**
  - K-Medoids
  - DBSCAN
  - Spectral clustering
  - and more…

# K-Means

# K-Means

- Unsupervised Machine Learning
  - Clustering
    - Partitioning

# K-Means

- Just like any clustering algorithm, K-Means groups data points into a set number of clusters

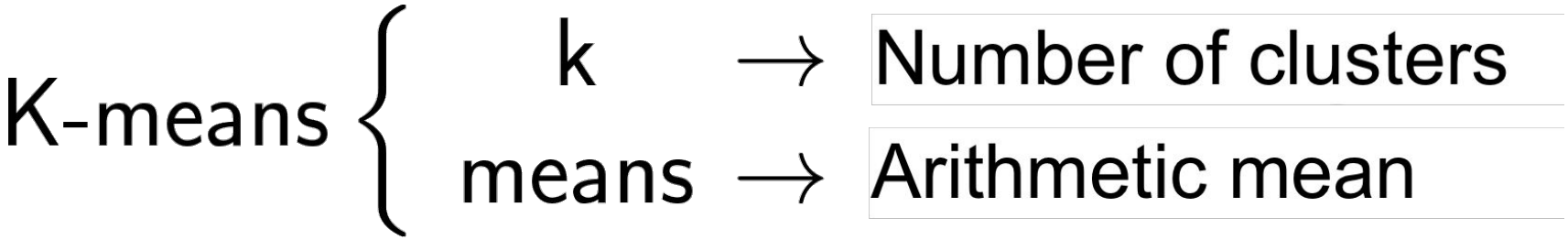
# K-Means

- Just like any clustering algorithm, K-Means groups data points into a set number of clusters

$$\text{K-means} \begin{cases} k \rightarrow \text{Number of clusters} \\ \text{means} \rightarrow \text{Arithmetic mean} \end{cases}$$

# K-Means

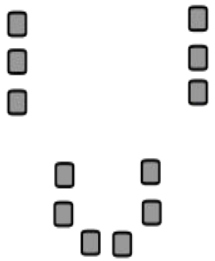
- Just like any clustering algorithm, K-Means groups data points into a set number of clusters



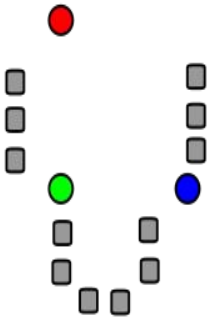

- K-means is an iterative algorithm that partitions a set of **N** observations into **K** groups in which each observation belongs to the group whose mean value is the closest

# Step 1: Initialize

# Step 1: Initialize

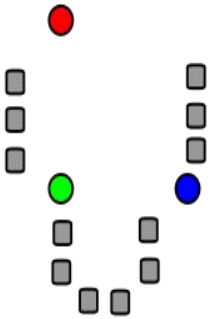


## Step 1: Initialize



- Choose a number K of clusters
- Randomly choose K points as centroids

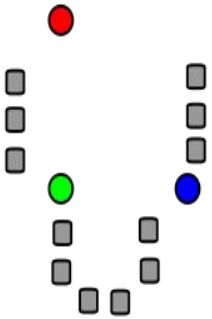
## Step 1: Initialize



- Choose a number K of clusters
- Randomly choose K points as centroids

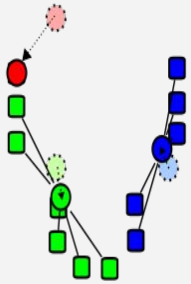
## Step 2: Repeat

## Step 1: Initialize

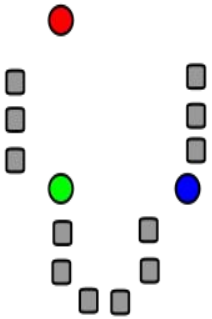


- Choose a number K of clusters
- Randomly choose K points as centroids

## Step 2: Repeat

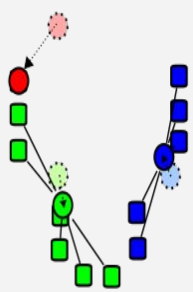


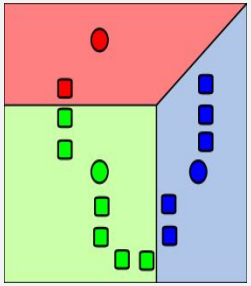
## Step 1: Initialize



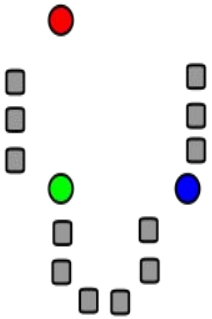
- Choose a number K of clusters
- Randomly choose K points as centroids

## Step 2: Repeat



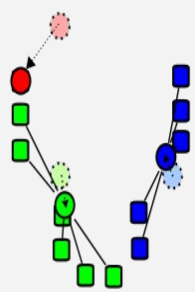


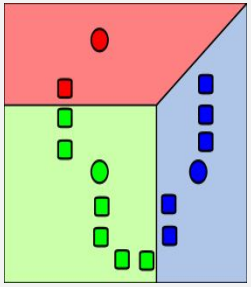
## Step 1: Initialize



- Choose a number K of clusters
- Randomly choose K points as centroids

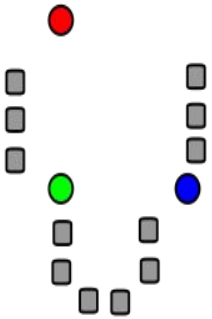
## Step 2: Repeat





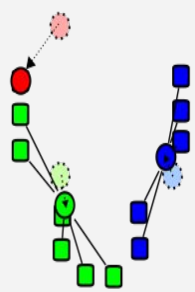
- K clusters are created by associating each observation with the nearest mean
- The new centroid of each of the K clusters is the mean of its observations

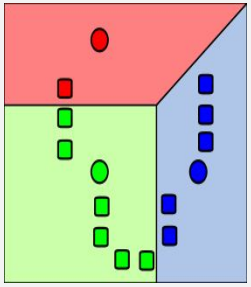
## Step 1: Initialize



- Choose a number K of clusters
- Randomly choose K points as centroids

## Step 2: Repeat

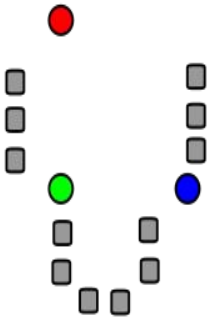




- K clusters are created by associating each observation with the nearest mean
- The new centroid of each of the K clusters is the mean of its observations

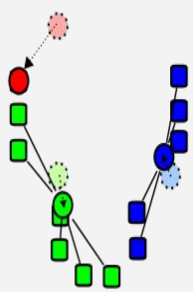
## Step 3: Stop

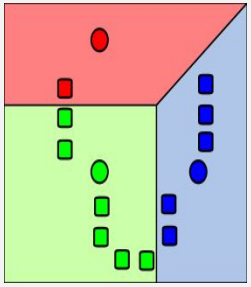
## Step 1: Initialize



- Choose a number K of clusters
- Randomly choose K points as centroids

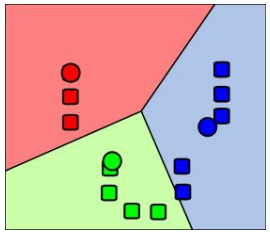
## Step 2: Repeat



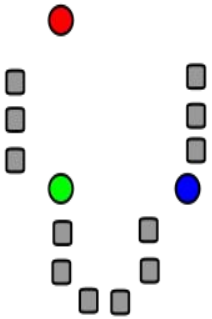


- K clusters are created by associating each observation with the nearest mean
- The new centroid of each of the K clusters is the mean of its observations

## Step 3: Stop

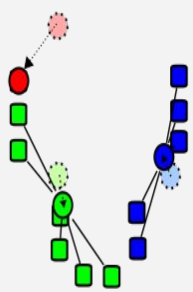


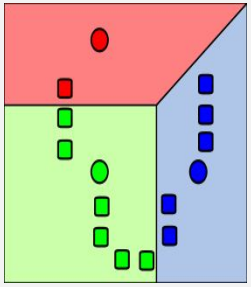
## Step 1: Initialize



- Choose a number K of clusters
- Randomly choose K points as centroids

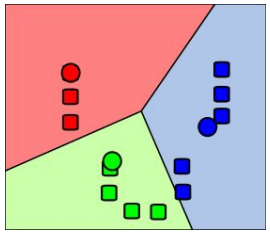
## Step 2: Repeat





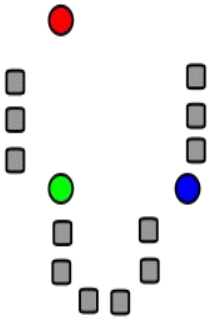
- K clusters are created by associating each observation with the nearest mean
- The new centroid of each of the K clusters is the mean of its observations

## Step 3: Stop



- Repeat steps 1 y 2
- The algorithm terminates when there is no longer a change in the cluster centroids, the cluster observations remain the same, or the maximum number of iterations is reached

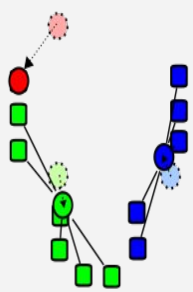
## Step 1: Initialize

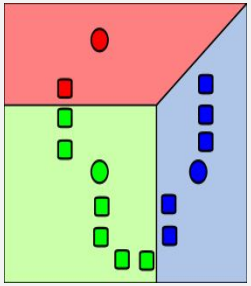


- Choose a number K of clusters
- Randomly choose K points as centroids

- Math behind it
- Video

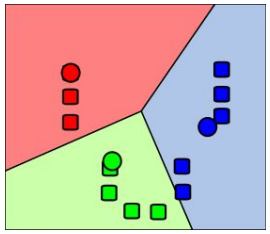
## Step 2: Repeat





- K clusters are created by associating each observation with the nearest mean
- The new centroid of each of the K clusters is the mean of its observations

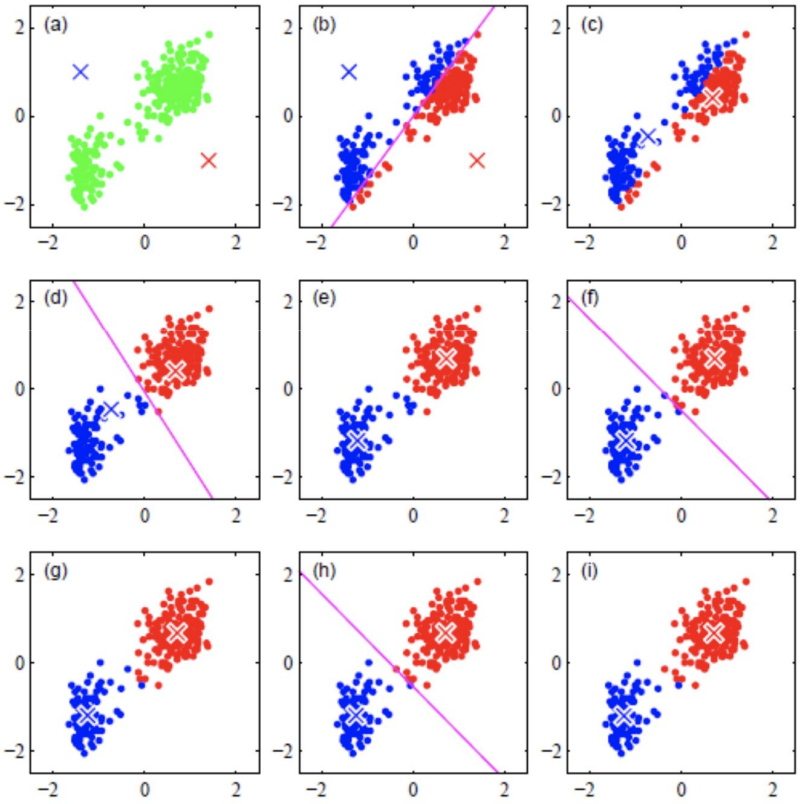
## Step 3: Stop



- Repeat steps 1 y 2
- The algorithm terminates when there is no longer a change in the cluster centroids, the cluster observations remain the same, or the maximum number of iterations is reached

# Checkpoint

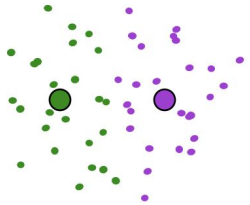
1. What's the number of K?
2. What is happening at each step?



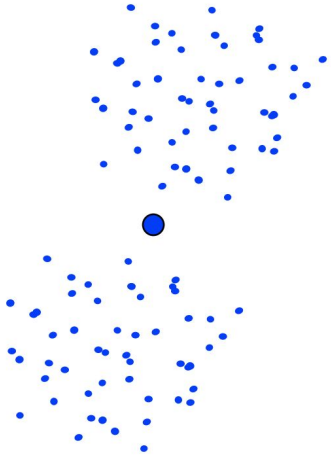
# 🤔 How do we pick K?

- It's important to pick a good number for K

A local optimum:



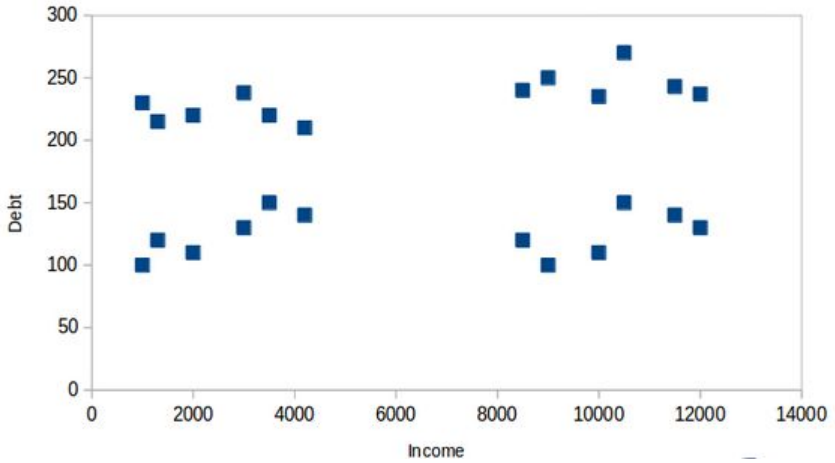
Would be better to have one cluster here



… and two clusters here

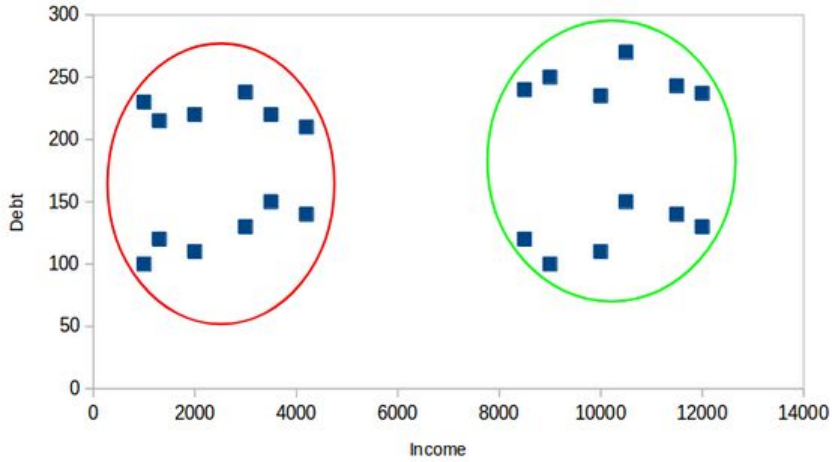
# 🤔 How do we pick K?

- How many clusters do we have here?



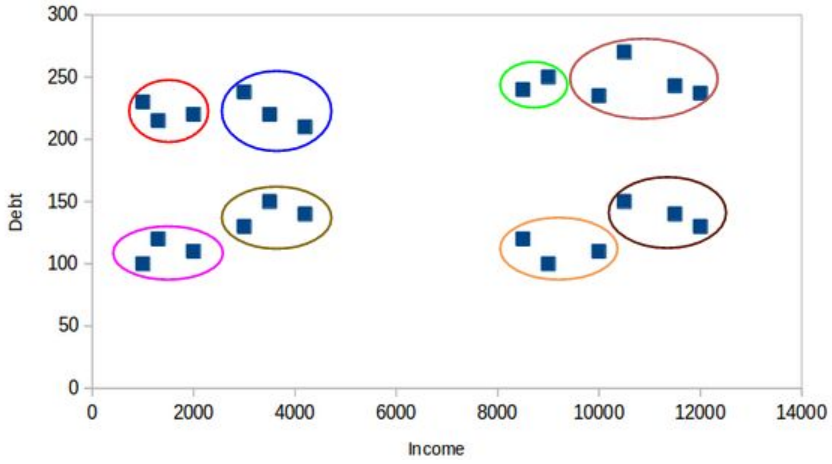
# 🤔 How do we pick K?

- How many clusters do we have here? Two?



# 🤔 How do we pick K?

- How many clusters do we have here? Eight?



# 👮‍♀️ Checkpoint

1. What's the minimum possible value for K?

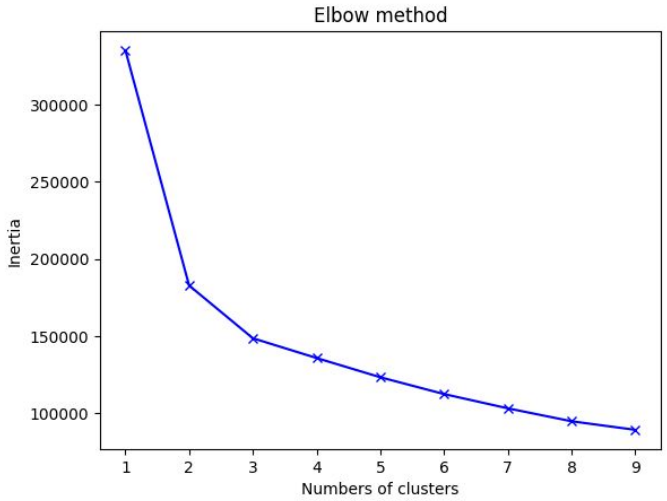
# 👮‍♀️ Checkpoint

1. What's the minimum possible value for K?

2. What's the maximum possible value for K?

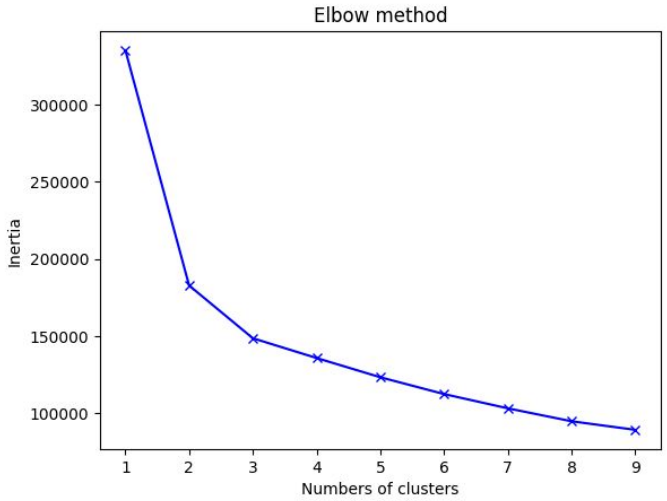
# Elbow method

- One method used to select the best K is known as the elbow method
- This decision can be subjective and is crucial because the number of clusters can significantly impact the insights derived from the clustering process



# Elbow method

- One method used to select the best K is known as the elbow method
- This decision can be subjective and is crucial because the number of clusters can significantly impact the insights derived from the clustering process

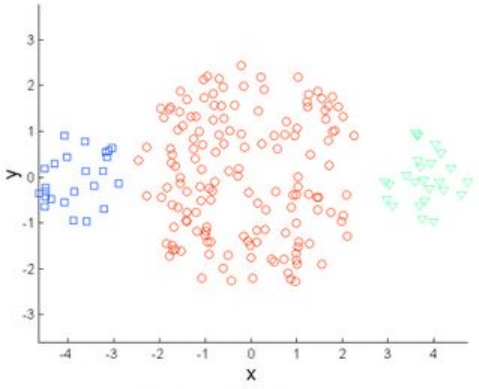



1. Compute K-means Clustering for Different Values of K
2. Calculate Inertia (sum of squared distances between each data point and the centroid of the cluster to which it is assigned)
3. Plot the Elbow Curve
4. Locate the "Elbow"

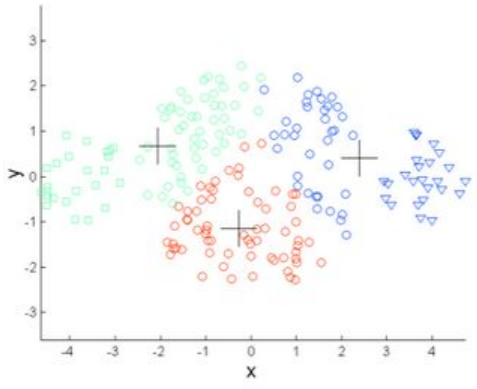
# 😕 What can go wrong?

- Disadvantages of K-means:
  - It's computationally intensive
  - Each observation belongs to a single cluster
  - Very sensitive to outlier observations
  - Cannot model complex relationships

# When not to use K-means

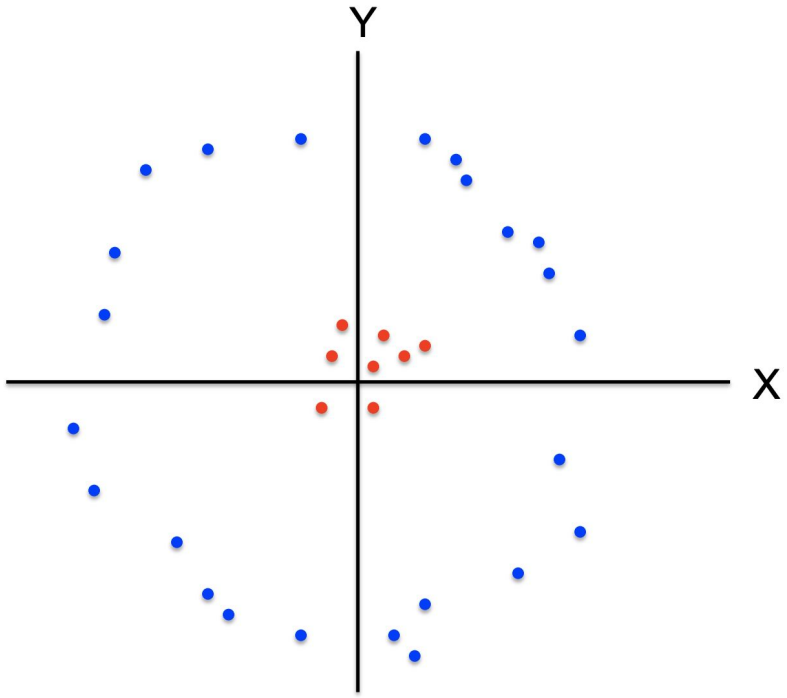



Original Points

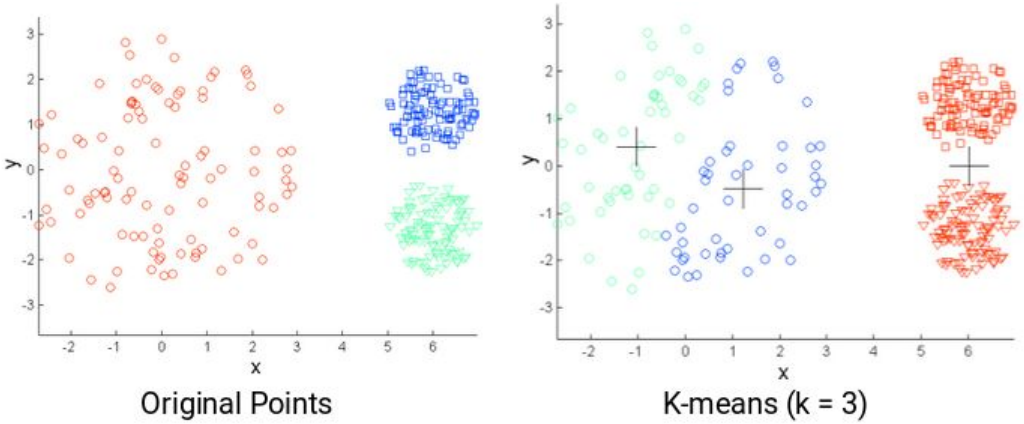



K-means (k = 3)

# When not to use K-means

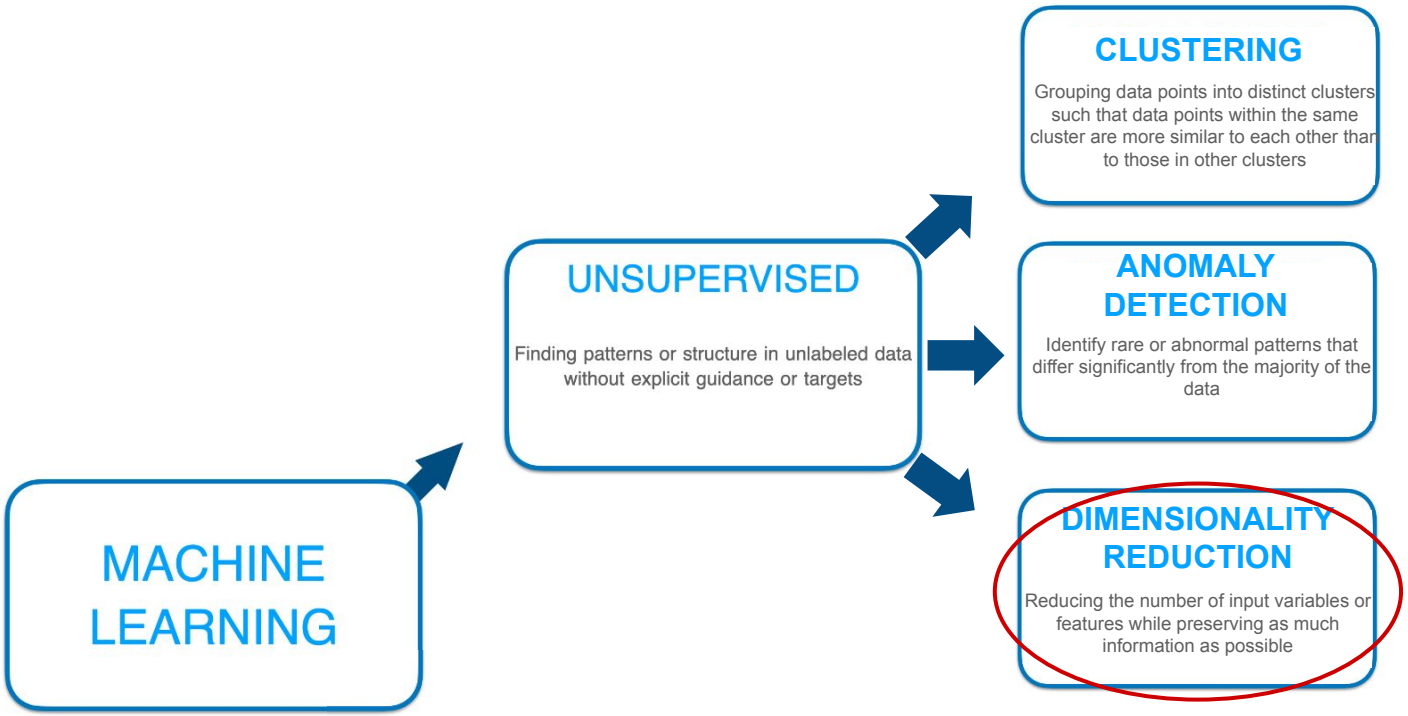


# When not to use K-means



# Code

- K-Means


MACHINE LEARNING
UNSUPERVISED
Finding patterns or structure in unlabeled data without explicit guidance or targets
CLUSTERING
Grouping data points into distinct clusters such that data points within the same cluster are more similar to each other than to those in other clusters
ANOMALY DETECTION
Identify rare or abnormal patterns that differ significantly from the majority of the data
DIMENSIONALITY REDUCTION
Reducing the number of input variables or features while preserving as much information as possible

# PCA

# Intuition behind PCA

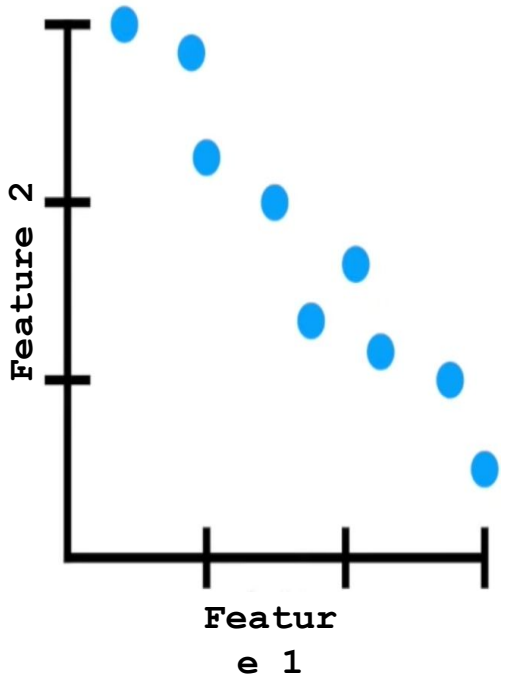
| | Feature 1 | Feature 2 | Feature 3 | Feature 4 | … |
|---|---|---|---|---|---|
| Observation 1 | 3 | 0.25 | 2.4 | 0.1 | |
| Observation 2 | 2.9 | 0.8 | 2.2 | 1.8 | |
| Observation 3 | 2.2 | 1 | 1.5 | 3.2 | |
| Observation 4 | 2 | 1.4 | 2 | 0.3 | |
| Observation 5 | 1.3 | 1.6 | 1.6 | 0 | |
| … | | | | | |

# Intuition behind PCA

| | Feature 1 | Feature 2 |
|---|---|---|
| Observation 1 | 3 | 0.25 |
| Observation 2 | 2.9 | 0.8 |
| Observation 3 | 2.2 | 1 |
| Observation 4 | 2 | 1.4 |
| Observation 5 | 1.3 | 1.6 |
| … | | |

# Intuition behind PCA

| | Feature 1 | Feature 2 |
|---|---|---|
| Observation 1 | 3 | 0.25 |
| Observation 2 | 2.9 | 0.8 |
| Observation 3 | 2.2 | 1 |
| Observation 4 | 2 | 1.4 |
| Observation 5 | 1.3 | 1.6 |
| ... | | |

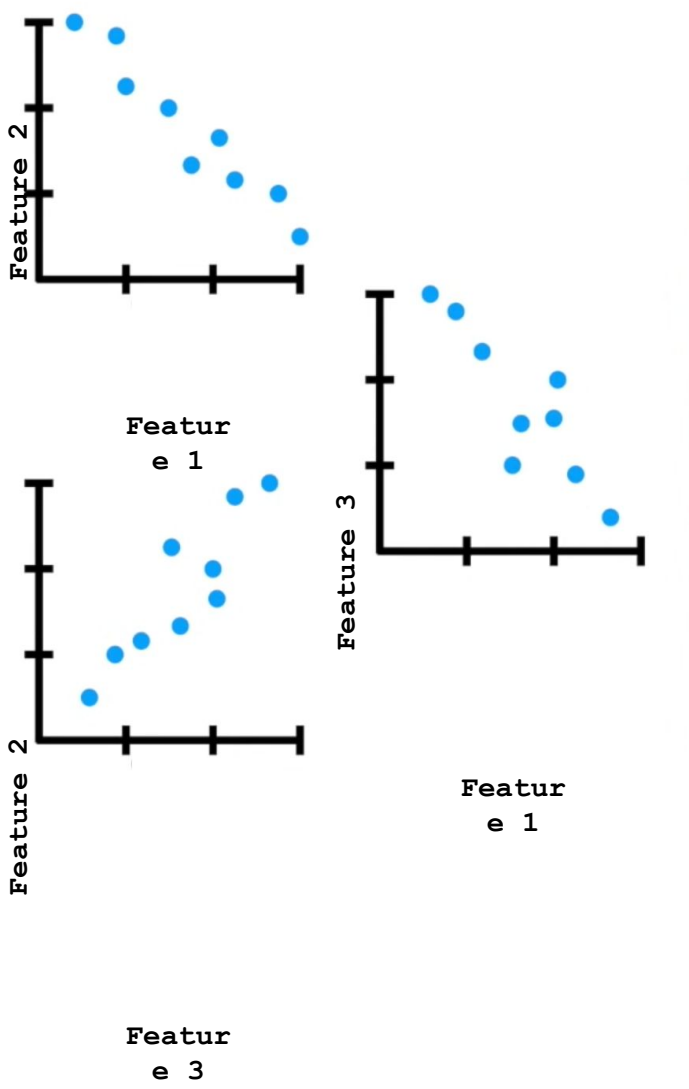


# Intuition behind PCA

| | Feature 1 | Feature 2 | Feature 3 |
|---|---|---|---|
| Observation 1 | 3 | 0.25 | 2.4 |
| Observation 2 | 2.9 | 0.8 | 2.2 |
| Observation 3 | 2.2 | 1 | 1.5 |
| Observation 4 | 2 | 1.4 | 2 |
| Observation 5 | 1.3 | 1.6 | 1.6 |
| ... | | | |

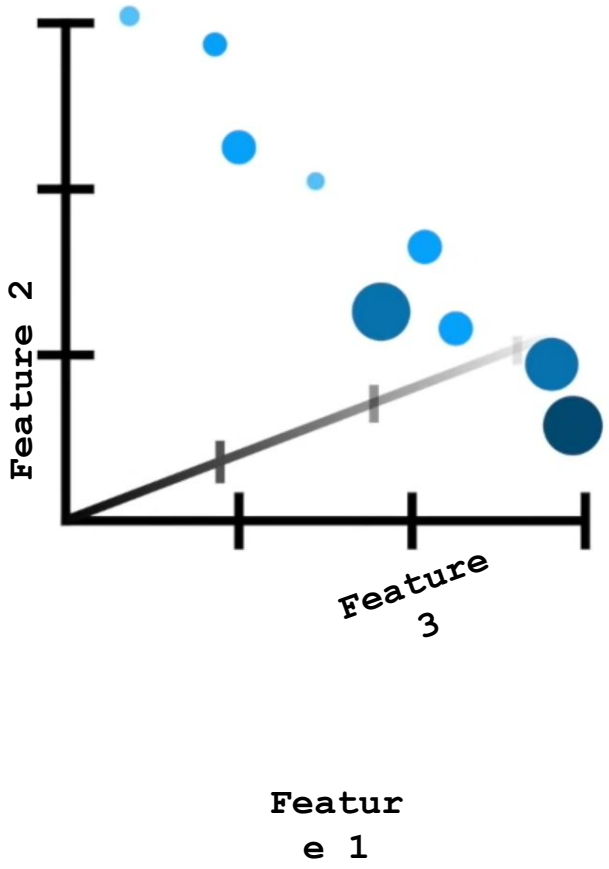
# Intuition behind PCA

| | Feature 1 | Feature 2 | Feature 3 |
|---|---|---|---|
| Observation 1 | 3 | 0.25 | 2.4 |
| Observation 2 | 2.9 | 0.8 | 2.2 |
| Observation 3 | 2.2 | 1 | 1.5 |
| Observation 4 | 2 | 1.4 | 2 |
| Observation 5 | 1.3 | 1.6 | 1.6 |
| ... | | | |



# Intuition behind PCA

| | Feature 1 | Feature 2 | Feature 3 |
|---|---|---|---|
| Observation 1 | 3 | 0.25 | 2.4 |
| Observation 2 | 2.9 | 0.8 | 2.2 |
| Observation 3 | 2.2 | 1 | 1.5 |
| Observation 4 | 2 | 1.4 | 2 |
| Observation 5 | 1.3 | 1.6 | 1.6 |
| ... | | | |



# Intuition behind PCA

| | Feature 1 | Feature 2 | Feature 3 | Feature 4 | … |
|---|---|---|---|---|---|
| Observation 1 | 3 | 0.25 | 2.4 | 0.1 | |
| Observation 2 | 2.9 | 0.8 | 2.2 | 1.8 | |
| Observation 3 | 2.2 | 1 | 1.5 | 3.2 | |
| Observation 4 | 2 | 1.4 | 2 | 0.3 | |
| Observation 5 | 1.3 | 1.6 | 1.6 | 0 | |
| … | | | | | |

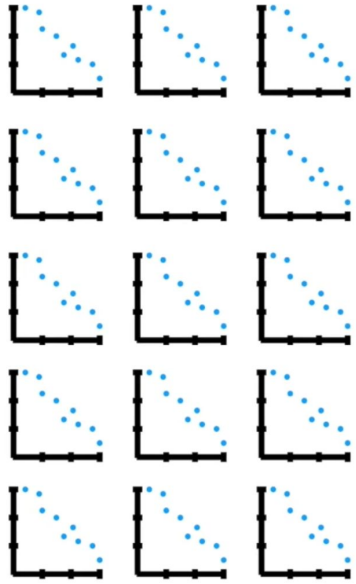
# Intuition behind PCA

| | Feature 1 | Feature 2 | Feature 3 | Feature 4 | … |
|---|---|---|---|---|---|
| Observation 1 | 3 | 0.25 | 2.4 | 0.1 | |
| Observation 2 | 2.9 | 0.8 | 2.2 | 1.8 | |
| Observation 3 | 2.2 | 1 | 1.5 | 3.2 | |
| Observation 4 | 2 | 1.4 | 2 | 0.3 | |
| Observation 5 | 1.3 | 1.6 | 1.6 | 0 | |
| … | | | | | |

???

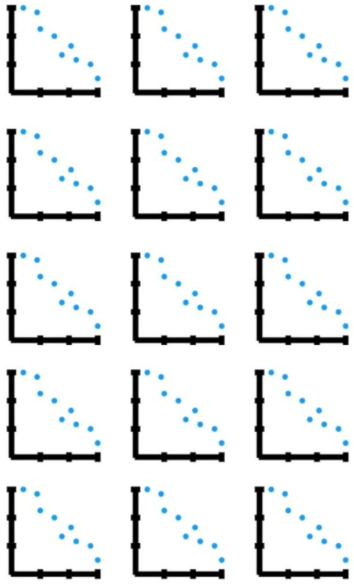
# Intuition behind PCA

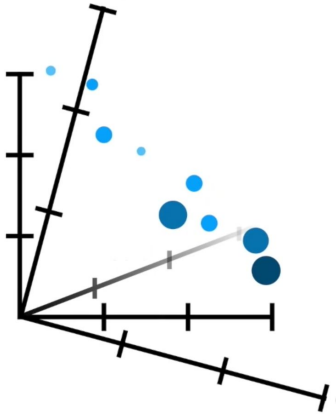
| | Feature 1 | Feature 2 | Feature 3 | Feature 4 | … |
|---|---|---|---|---|---|
| Observation 1 | 3 | 0.25 | 2.4 | 0.1 | |
| Observation 2 | 2.9 | 0.8 | 2.2 | 1.8 | |
| Observation 3 | 2.2 | 1 | 1.5 | 3.2 | |
| Observation 4 | 2 | 1.4 | 2 | 0.3 | |
| Observation 5 | 1.3 | 1.6 | 1.6 | 0 | |
| … | | | | | |



# Intuition behind PCA

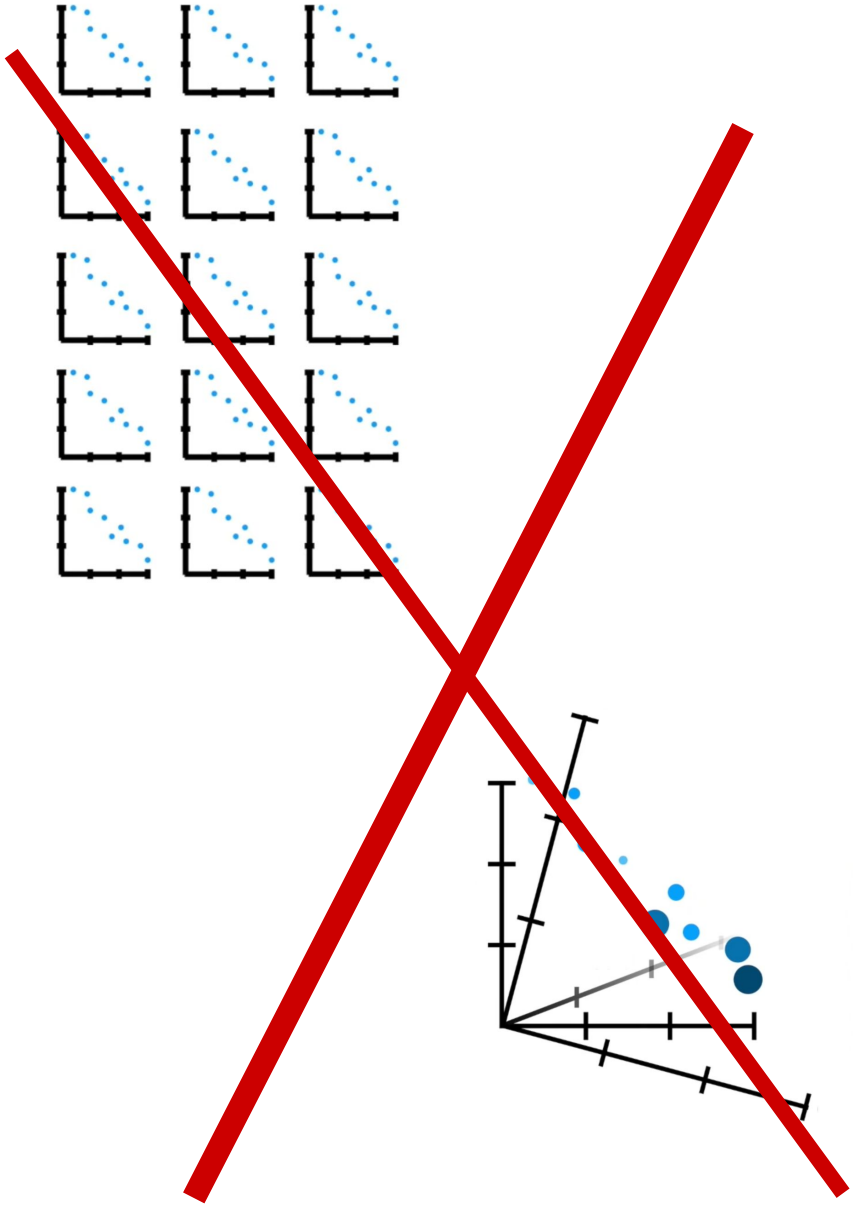
| | Feature 1 | Feature 2 | Feature 3 | Feature 4 | … |
|---|---|---|---|---|---|
| Observation 1 | 3 | 0.25 | 2.4 | 0.1 | |
| Observation 2 | 2.9 | 0.8 | 2.2 | 1.8 | |
| Observation 3 | 2.2 | 1 | 1.5 | 3.2 | |
| Observation 4 | 2 | 1.4 | 2 | 0.3 | |
| Observation 5 | 1.3 | 1.6 | 1.6 | 0 | |
| … | | | | | |





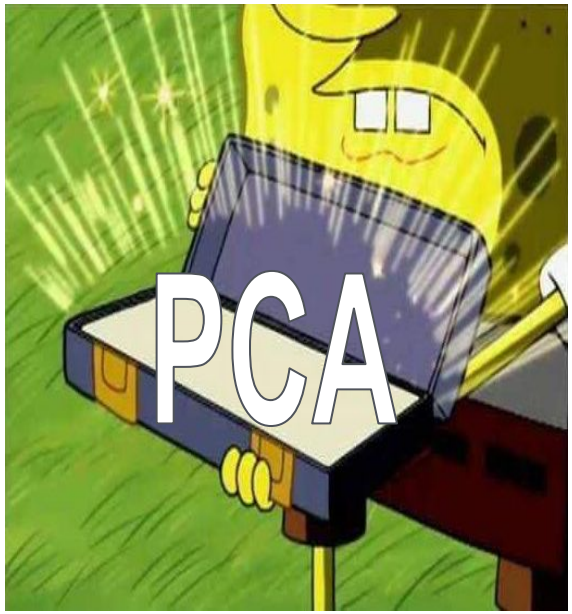
# Intuition behind PCA

| | Feature 1 | Feature 2 | Feature 3 | Feature 4 | … |
|---|---|---|---|---|---|
| Observation 1 | 3 | 0.25 | 2.4 | 0.1 | |
| Observation 2 | 2.9 | 0.8 | 2.2 | 1.8 | |
| Observation 3 | 2.2 | 1 | 1.5 | 3.2 | |
| Observation 4 | 2 | 1.4 | 2 | 0.3 | |
| Observation 5 | 1.3 | 1.6 | 1.6 | 0 | |
| … | | | | | |



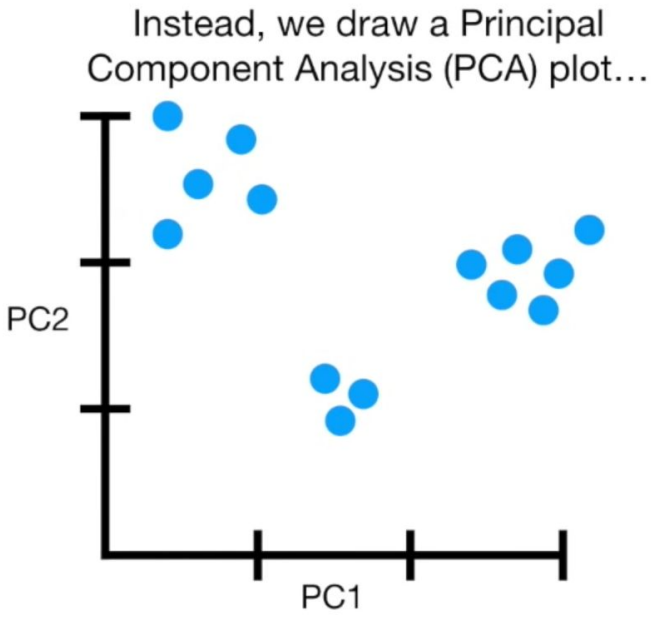
# Intuition behind PCA

| | Feature 1 | Feature 2 | Feature 3 | Feature 4 | … |
|---|---|---|---|---|---|
| Observation 1 | 3 | 0.25 | 2.4 | 0.1 | |
| Observation 2 | 2.9 | 0.8 | 2.2 | 1.8 | |
| Observation 3 | 2.2 | 1 | 1.5 | 3.2 | |
| Observation 4 | 2 | 1.4 | 2 | 0.3 | |
| Observation 5 | 1.3 | 1.6 | 1.6 | 0 | |
| … | | | | | |



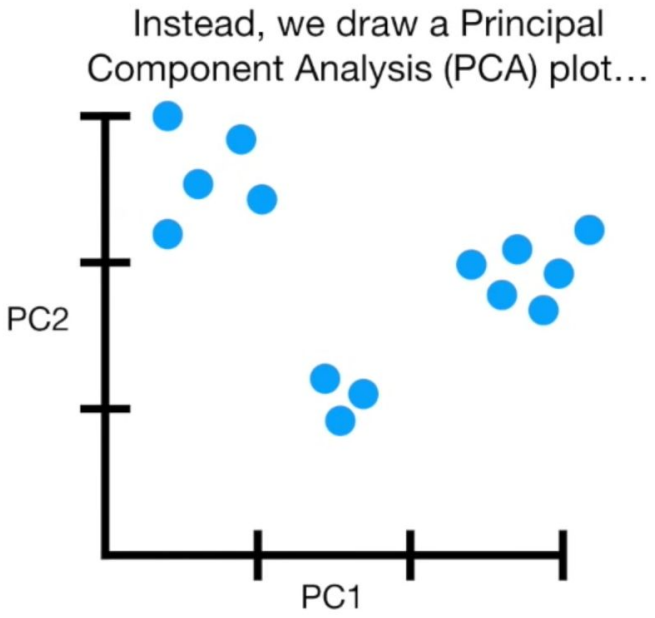
# Intuition behind PCA

| | Feature 1 | Feature 2 | Feature 3 | Feature 4 | … |
|---|---|---|---|---|---|
| Observation 1 | 3 | 0.25 | 2.4 | 0.1 | |
| Observation 2 | 2.9 | 0.8 | 2.2 | 1.8 | |
| Observation 3 | 2.2 | 1 | 1.5 | 3.2 | |
| Observation 4 | 2 | 1.4 | 2 | 0.3 | |
| Observation 5 | 1.3 | 1.6 | 1.6 | 0 | |
| … | | | | | |



# Intuition behind PCA

| | Feature 1 | Feature 2 | Feature 3 | Feature 4 | … |
|---|---|---|---|---|---|
| Observation 1 | 3 | 0.25 | 2.4 | 0.1 | |
| Observation 2 | 2.9 | 0.8 | 2.2 | 1.8 | |
| Observation 3 | 2.2 | 1 | 1.5 | 3.2 | |
| Observation 4 | 2 | 1.4 | 2 | 0.3 | |
| Observation 5 | 1.3 | 1.6 | 1.6 | 0 | |
| … | | | | | |



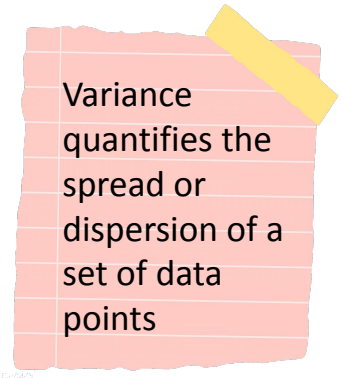

Differences along PC1 are more important than differences along PC2 and so on...

# Principal Component Analysis

- Powerful technique used for dimensionality reduction, feature extraction, and data visualization
- PCA transforms the original variables of a dataset into a new set of variables, called **principal components**
- These principal components are orthogonal to each other (uncorrelated), and they reflect the maximum variance in the data
- The first principal component captures the most variance, the second one the second most, and so on

# Principal Component Analysis

- Powerful technique used for dimensionality reduction, feature extraction, and data visualization
- PCA transforms the original variables of a dataset into a new set of variables, called **principal components**
- These principal components are orthogonal to each other (uncorrelated), and they reflect the maximum variance in the data
- The first principal component captures the most variance, the second one the second most, and so on



# Why use PCA?

- **Dimensionality Reduction**: In datasets with a large number of variables, it might be that much of the variation in data can be captured by a smaller number of principal components. This reduced feature set can make subsequent analysis simpler and more robust.

- **Noise Reduction**: By retaining only the significant principal components (those capturing major variance), one can discard components that are likely capturing noise.

- **Visualization**: It's challenging to visualize high-dimensional data. PCA can reduce data to 2 or 3 dimensions, suitable for visualization.

- **Feature Extraction**: The principal components can serve as input features for machine learning models.

# Limitations of PCA

- **Linearity**: PCA assumes a linear relationship among variables. It might not capture non-linear relationships well.
- **Loss of Interpretability**: Original features get transformed into principal components, which might not have clear, interpretable meanings.
- **Sensitive to Scaling**: The results of PCA can change based on the scaling of variables, hence standardization is recommended.

# How does PCA work?

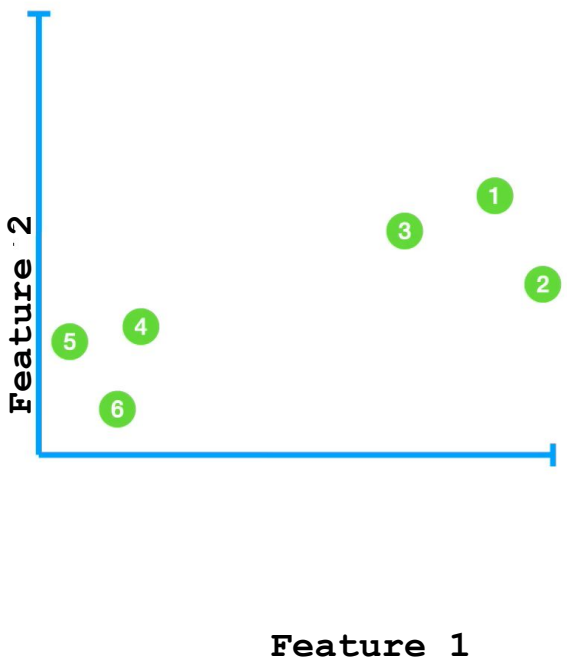
| | Feature 1 | Feature 2 | Feature 3 | Feature 4 |
|---|---|---|---|---|
| Observation 1 | 10 | 6 | 12 | 5 |
| Observation 2 | 11 | 4 | 9 | 7 |
| Observation 3 | 8 | 5 | 10 | 6 |
| Observation 4 | 3 | 3 | 2.5 | 2 |
| Observation 5 | 1 | 2.8 | 1.3 | 4 |
| Observation 6 | 2 | 1 | 2 | 7 |

# How does PCA work?

| | Feature 1 |
|---|---|
| Observation 1 | 10 |
| Observation 2 | 11 |
| Observation 3 | 8 |
| Observation 4 | 3 |
| Observation 5 | 1 |
| Observation 6 | 2 |

# How does PCA work?

| | Feature 1 | Feature 2 |
|---|---|---|
| Observation 1 | 10 | 6 |
| Observation 2 | 11 | 4 |
| Observation 3 | 8 | 5 |
| Observation 4 | 3 | 3 |
| Observation 5 | 1 | 2.8 |
| Observation 6 | 2 | 1 |



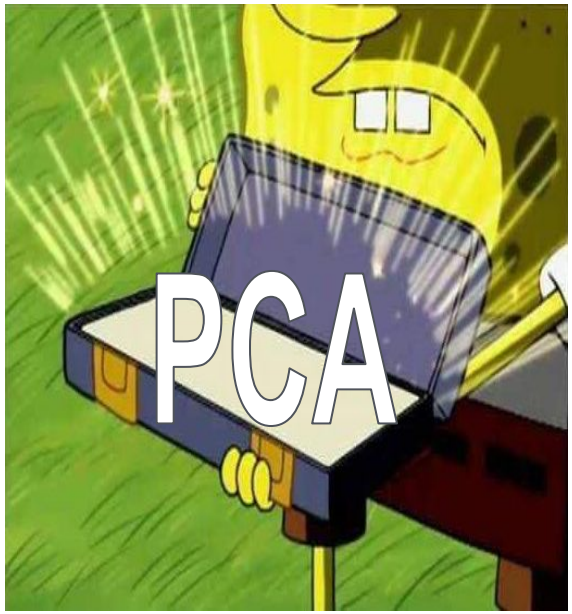
# How does PCA work?

| | Feature 1 | Feature 2 | Feature 3 | Feature 4 |
|---|---|---|---|---|
| Observation 1 | 10 | 6 | 12 | 5 |
| Observation 2 | 11 | 4 | 9 | 7 |
| Observation 3 | 8 | 5 | 10 | 6 |
| Observation 4 | 3 | 3 | 2.5 | 2 |
| Observation 5 | 1 | 2.8 | 1.3 | 4 |
| Observation 6 | 2 | 1 | 2 | 7 |

???

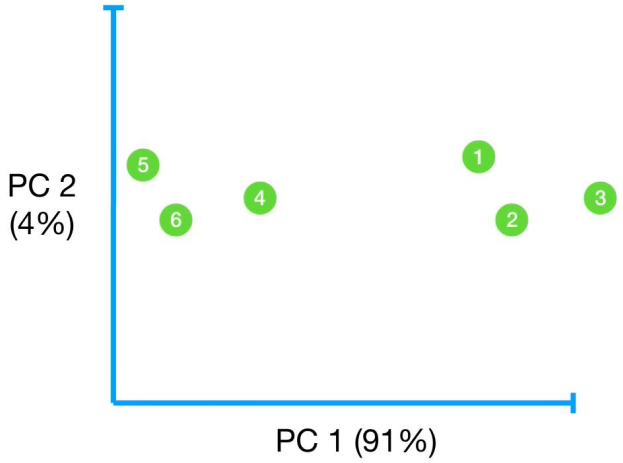
# How does PCA work?

| | Feature 1 | Feature 2 | Feature 3 | Feature 4 |
|---|---|---|---|---|
| Observation 1 | 10 | 6 | 12 | 5 |
| Observation 2 | 11 | 4 | 9 | 7 |
| Observation 3 | 8 | 5 | 10 | 6 |
| Observation 4 | 3 | 3 | 2.5 | 2 |
| Observation 5 | 1 | 2.8 | 1.3 | 4 |
| Observation 6 | 2 | 1 | 2 | 7 |



# How does PCA work?

| | Feature 1 | Feature 2 | Feature 3 | Feature 4 |
|---|---|---|---|---|
| Observation 1 | 10 | 6 | 12 | 5 |
| Observation 2 | 11 | 4 | 9 | 7 |
| Observation 3 | 8 | 5 | 10 | 6 |
| Observation 4 | 3 | 3 | 2.5 | 2 |
| Observation 5 | 1 | 2.8 | 1.3 | 4 |
| Observation 6 | 2 | 1 | 2 | 7 |

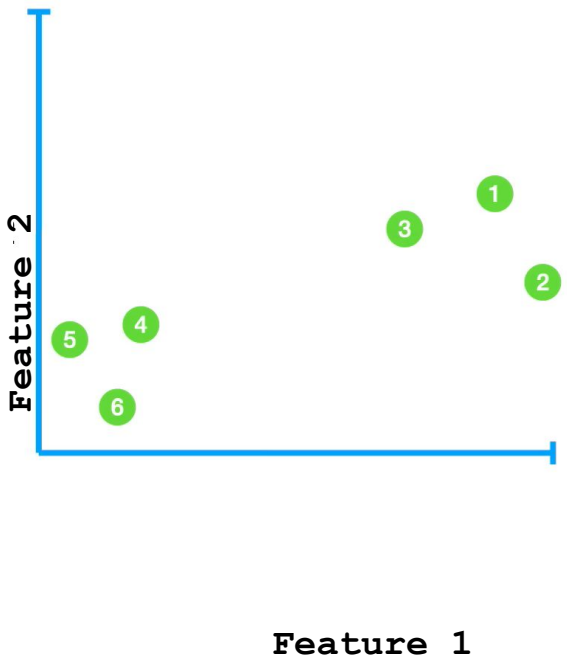



- PCA shows us that similar observations cluster together
- PCA tells us which feature is the most valuable to cluster the data
- PCA can tell us how accurate the 2D graph is

# PCA steps

- Plot the data

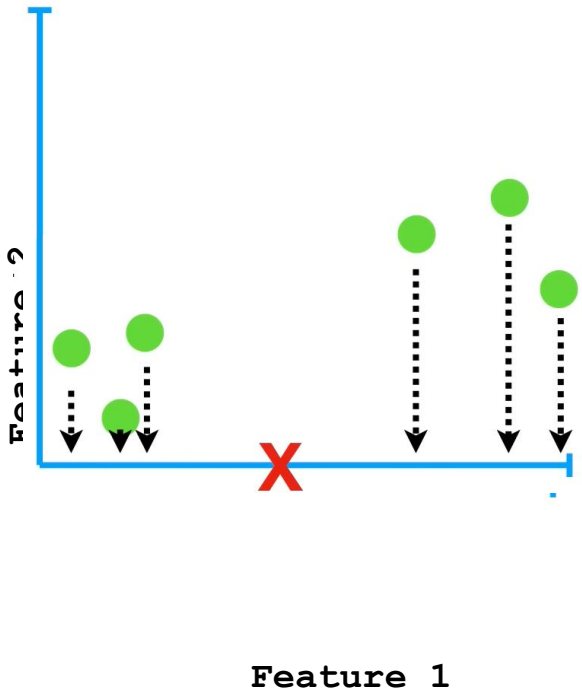
| | Feature 1 | Feature 2 |
|---|---|---|
| Observation 1 | 10 | 6 |
| Observation 2 | 11 | 4 |
| Observation 3 | 8 | 5 |
| Observation 4 | 3 | 3 |
| Observation 5 | 1 | 2.8 |
| Observation 6 | 2 | 1 |



# PCA steps

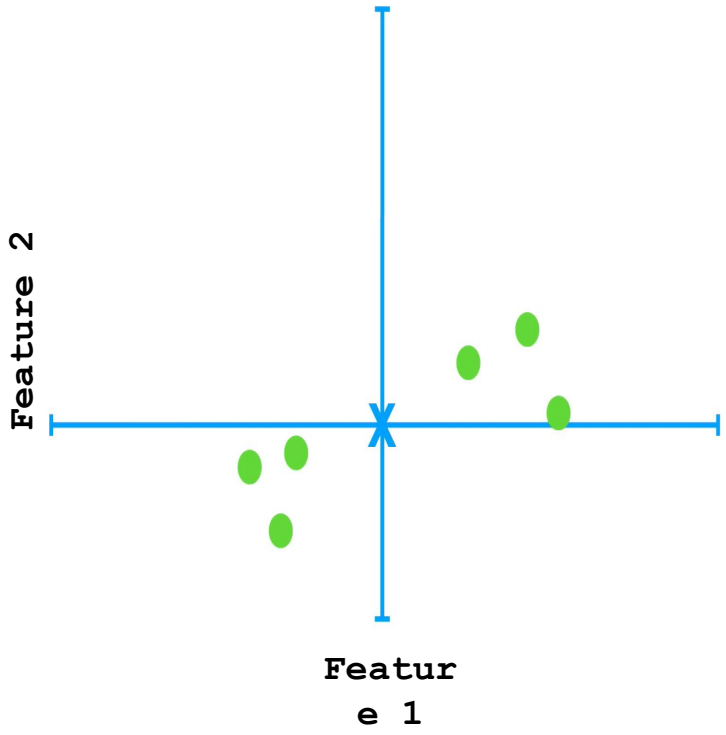
- Plot the data
- Calculate avg Feature 1

| | Feature 1 | Feature 2 |
|---|---|---|
| Observation 1 | 10 | 6 |
| Observation 2 | 11 | 4 |
| Observation 3 | 8 | 5 |
| Observation 4 | 3 | 3 |
| Observation 5 | 1 | 2.8 |
| Observation 6 | 2 | 1 |



# PCA steps

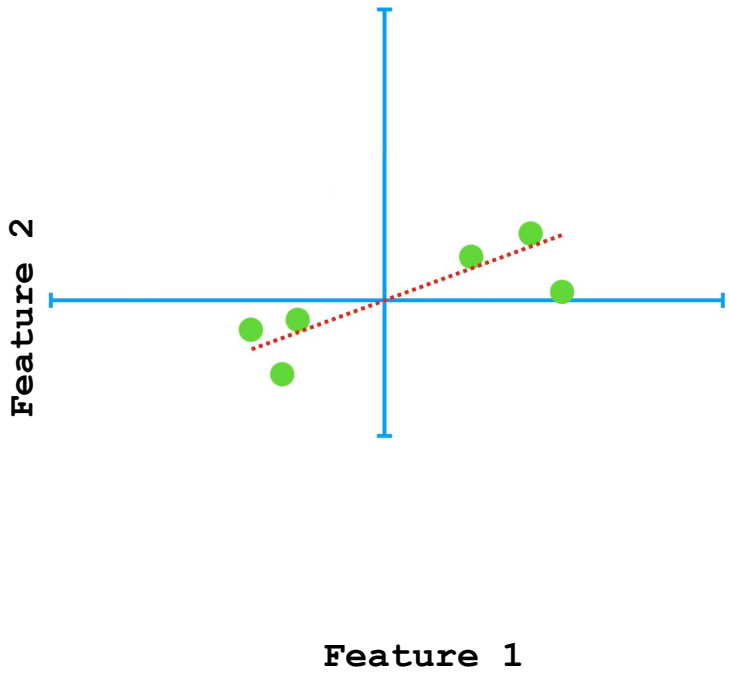
| | Feature 1 | Feature 2 |
|---|---|---|
| Observation 1 | 10 | 6 |
| Observation 2 | 11 | 4 |
| Observation 3 | 8 | 5 |
| Observation 4 | 3 | 3 |
| Observation 5 | 1 | 2.8 |
| Observation 6 | 2 | 1 |




- Plot the data
- Calculate avg Feature 1
- Calculate avg Feature 2
- Calculate center of the data
- Shift the data so the origin is in the center of the graph

# PCA steps

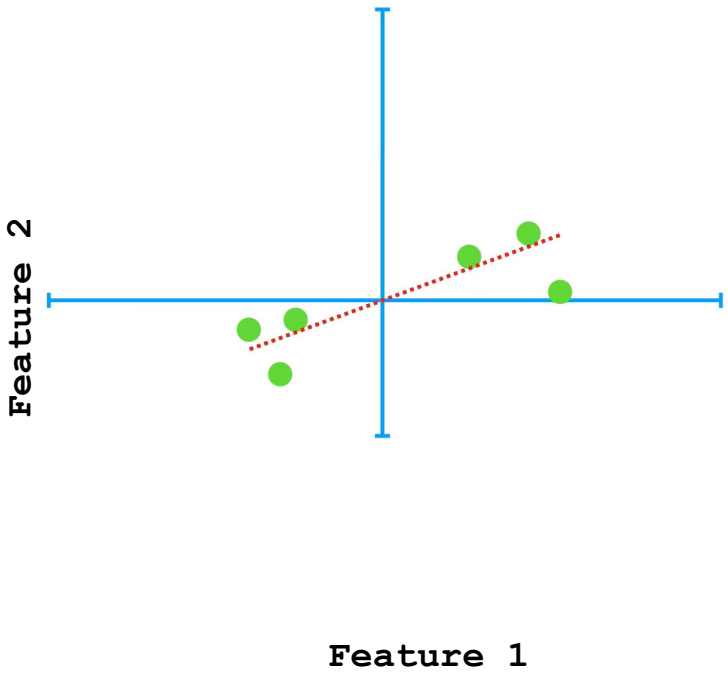
| | Feature 1 | Feature 2 |
|---|---|---|
| Observation 1 | 10 | 6 |
| Observation 2 | 11 | 4 |
| Observation 3 | 8 | 5 |
| Observation 4 | 3 | 3 |
| Observation 5 | 1 | 2.8 |
| Observation 6 | 2 | 1 |



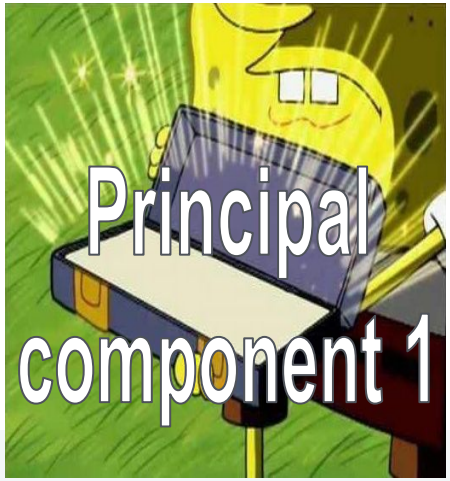

- Plot the data
- Calculate avg Feature 1
- Calculate avg Feature 2
- Calculate center of the data
- Shift the data so the origin is in the center of the graph
- Find the best fit line

# PCA steps

| | Feature 1 | Feature 2 |
|---|---|---|
| Observation 1 | 10 | 6 |
| Observation 2 | 11 | 4 |
| Observation 3 | 8 | 5 |
| Observation 4 | 3 | 3 |
| Observation 5 | 1 | 2.8 |
| Observation 6 | 2 | 1 |

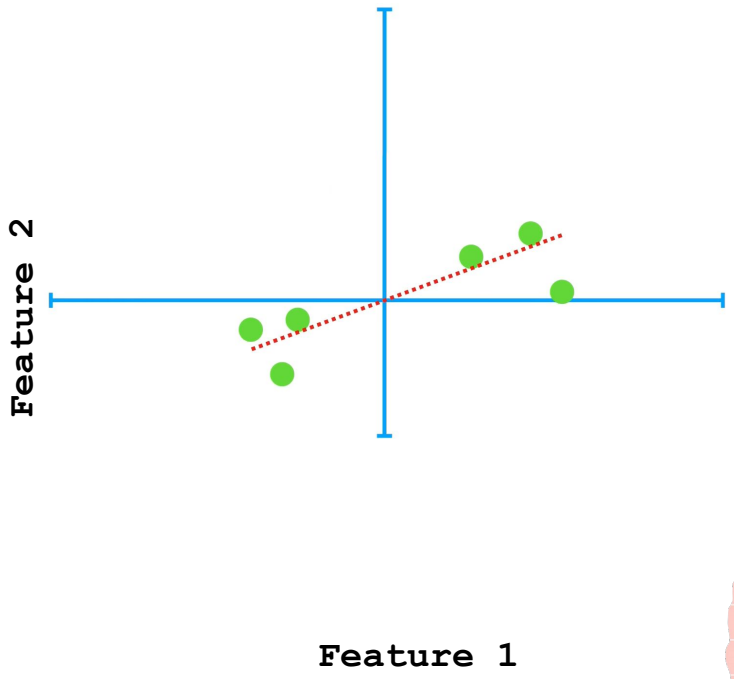



- Plot the data
- Calculate avg Feature 1
- Calculate avg Feature 2
- Calculate center of the data
- Shift the data so the origin is in the center of the graph
- Find the best fit line



# PCA steps

| | Feature 1 | Feature 2 |
|---|---|---|
| Observation 1 | 10 | 6 |
| Observation 2 | 11 | 4 |
| Observation 3 | 8 | 5 |
| Observation 4 | 3 | 3 |
| Observation 5 | 1 | 2.8 |
| Observation 6 | 2 | 1 |

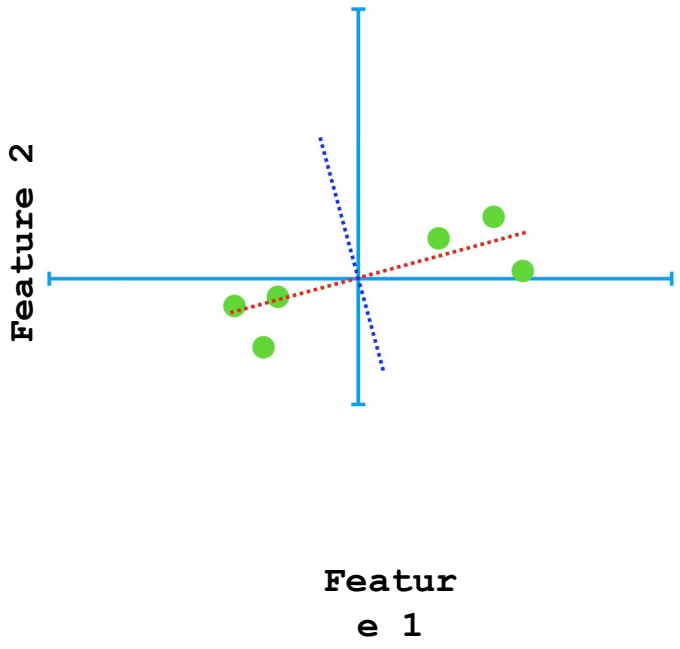



- Plot the data
- Calculate avg Feature 1
- Calculate avg Feature 2
- Calculate center of the data
- Shift the data so the origin is in the center of the graph
- Find the best fit line

- PC1 has a slope of 0.25
- Ratio tells you that f1 is more important to describe how spread out the data is
- PC1 is a linear combination of variables

# PCA steps

| | Feature 1 | Feature 2 |
|---|---|---|
| Observation 1 | 10 | 6 |
| Observation 2 | 11 | 4 |
| Observation 3 | 8 | 5 |
| Observation 4 | 3 | 3 |
| Observation 5 | 1 | 2.8 |
| Observation 6 | 2 | 1 |

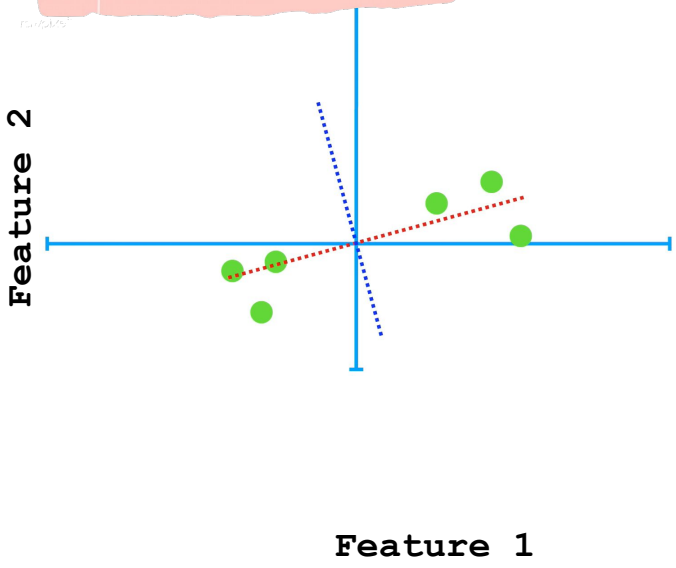



- Plot the data
- Calculate avg Feature 1
- Calculate avg Feature 2
- Calculate center of the data
- Shift the data so the origin is in the center of the graph
- Find the best fit line
- PC2 is simply the orthogonal line to PC1

# PCA steps

| | Feature 1 | Feature 2 |
|---|---|---|
| Observation 1 | 10 | 6 |
| Observation 2 | 11 | 4 |
| Observation 3 | 8 | 5 |
| Observation 4 | 3 | 3 |
| Observation 5 | 1 | 2.8 |
| Observation 6 | 2 | 1 |

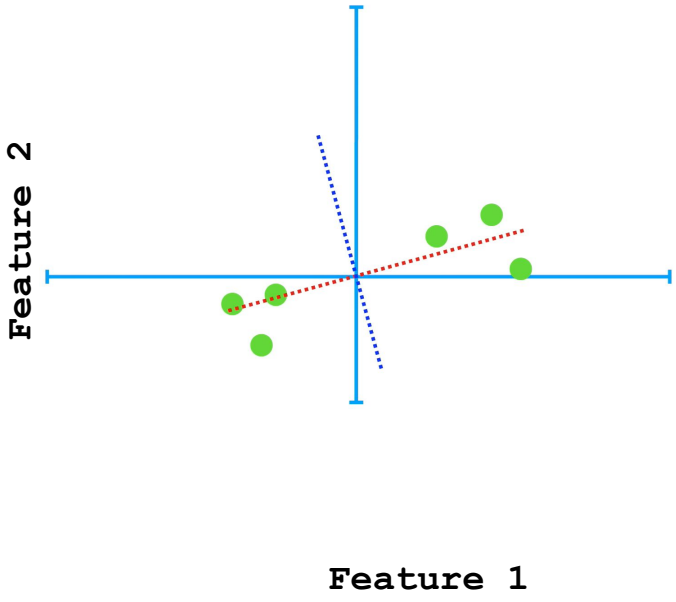
These principal components are orthogonal to each other (uncorrelated), and they reflect the maximum variance in the data




- Plot the data
- Calculate avg Feature 1
- Calculate avg Feature 2
- Calculate center of the data
- Shift the data so the origin is in the center of the graph
- Find the best fit line
- PC2 is simply the orthogonal line to PC1

# PCA steps

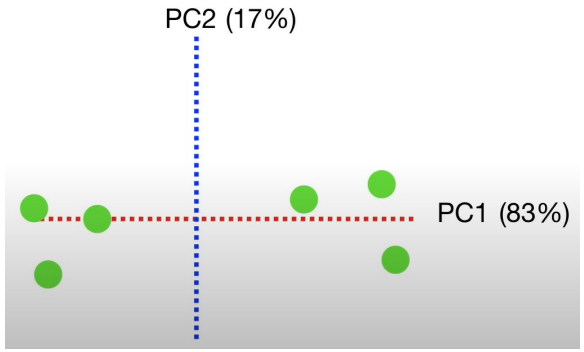
| | Feature 1 | Feature 2 |
|---|---|---|
| Observation 1 | 10 | 6 |
| Observation 2 | 11 | 4 |
| Observation 3 | 8 | 5 |
| Observation 4 | 3 | 3 |
| Observation 5 | 1 | 2.8 |
| Observation 6 | 2 | 1 |




- Plot the data
- Calculate avg Feature 1
- Calculate avg Feature 2
- Calculate center of the data
- Shift the data so the origin is in the center of the graph
- Find the best fit line
- PC2 is simply the orthogonal line to PC1
- You can compute a PC for each feature you have

# How much variation each principal component (PC) accounts for

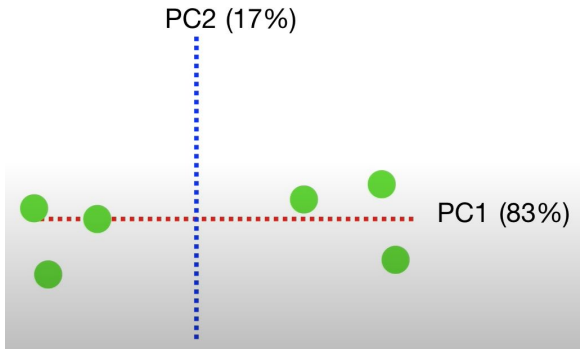
| | Feature 1 | Feature 2 |
|---|---|---|
| Observation 1 | 10 | 6 |
| Observation 2 | 11 | 4 |
| Observation 3 | 8 | 5 |
| Observation 4 | 3 | 3 |
| Observation 5 | 1 | 2.8 |
| Observation 6 | 2 | 1 |



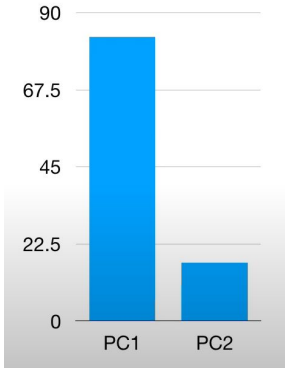

- Get the covariance matrix of your data
- Compute eigenvalues
- Total variance (sum all the eigenvalues)
- Compute proportion of variance

# How much variation each principal component (PC) accounts for

| | Feature 1 | Feature 2 |
|---|---|---|
| Observation 1 | 10 | 6 |
| Observation 2 | 11 | 4 |
| Observation 3 | 8 | 5 |
| Observation 4 | 3 | 3 |
| Observation 5 | 1 | 2.8 |
| Observation 6 | 2 | 1 |




- Get the covariance matrix of your data
- Compute eigenvalues
- Total variance (sum all the eigenvalues)
- Compute proportion of




Scree plot

# References

- StatQuest

# Code

- PCA
- Application of SVD: Recommender Systems. We will learn this in Mod 19

# AGENDA

- ✅ Assignment: Practical Applications I Feedback QA
- ✅ Required activities for Module 6
- ✅ Content review Module 6: Data Clustering and PCA
- Questions

# QUESTIONS?

# AGENDA

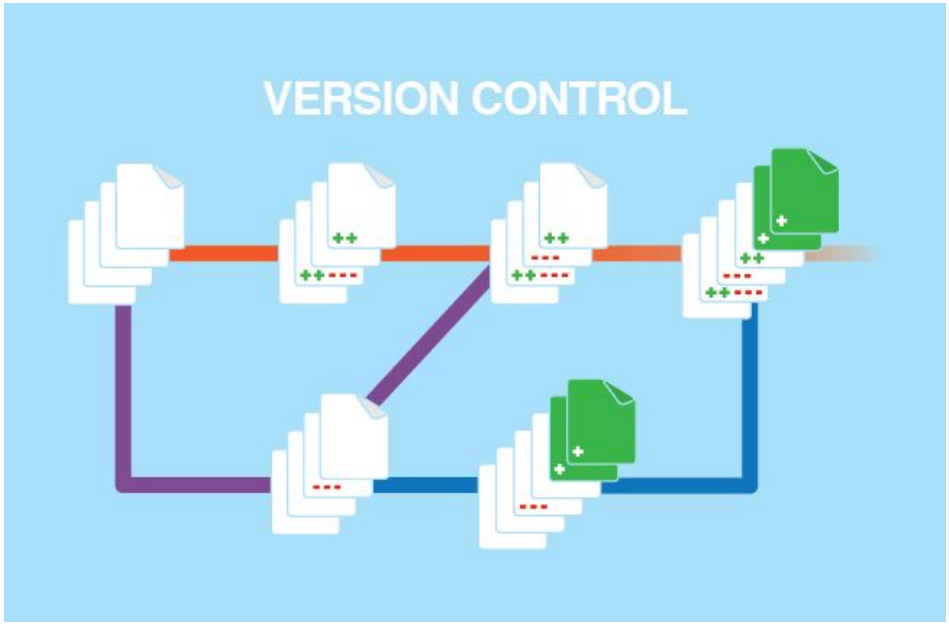
- ✅ Assignment: Practical Applications I Feedback QA
- ✅ Required activities for Module 6
- ✅ Content review Module 6: Data Clustering and PCA
- ✅ Questions

# APPENDIX

# Assignment: Practical Applications I Feedback

- Version Control
  - Git
  - GitHub

# The importance of Version Control




- System to track changes in files over time during software development
- Allows multiple people to work on the same project without stepping on each other's toes
- Facilitates easy rollback, collaboration, and parallel work

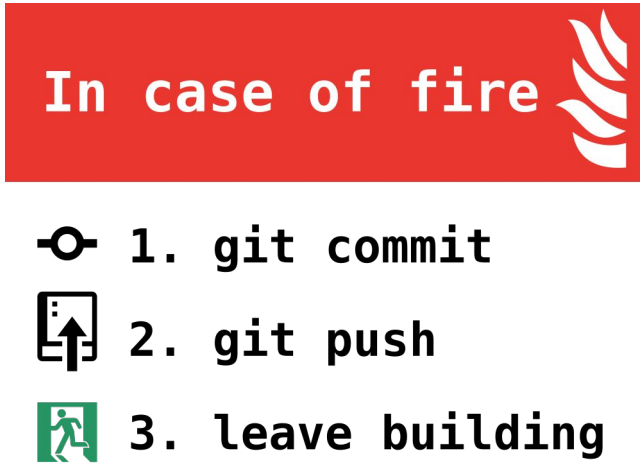
# Git


- Git is the most popular Version Control system nowadays
- Specifically, it's a Distributed Version Control System (DVCS), meaning every working copy of the codebase is a complete repository with full history
- A command-line tool (though there are GUI versions available) that manages source code history
- Can be used independently of any online platform

# GitHub


- GitHub is a platform that provides hosting for software development version control using Git. It allows multiple people to work on projects simultaneously, without interfering with each other's work
- It's the world's leading software development platform with millions of developers hosting and reviewing code, managing projects, and building software
- Offers a user-friendly interface and additional features like pull requests, issue tracking, and integrations with other developer tools
- While it's built around Git, using GitHub also means using Git, but the reverse isn't necessarily true. One can use Git without ever touching GitHub

# GitHub – Importance in Data Science




- **Collaboration**: Data scientists often work on teams and need a platform to collaborate, share code, and maintain versions of their scripts, notebooks, and data
- **Reproducibility**: GitHub ensures that there's a record of code changes, allowing for transparency and reproducibility in data science tasks
- **Portfolio Building**: For budding data scientists, having a GitHub profile can serve as a portfolio of their projects, demonstrating their coding and analytical skills to potential employers

# GitHub – Basic terminology

- **Repository (Repo):** A directory or storage space where your project lives
- **Commit:** A saved change to your repo
- **Branch:** A parallel version of a repository
- **Pull Request (PR):** Proposing your changes and requesting that someone review and pull in your contribution
- **Merge:** Merging your changes back to the main (master) branch
- **Fork:** A personal copy of another user's repository that lives on your account
- **Clone:** A copy of a repo that exists on your local computer

# GitHub Pages

- GitHub Pages is a static site hosting service offered by GitHub. It allows users to transform their GitHub repositories into websites
- By default, the URL of your GitHub Pages site will be in the format `username.github.io/repository-name` . However, you can also set up a custom domain if you want a more professional or personalized URL
- GitHub Pages is free for public repositories, but there are some limitations
- Documentation: https://docs.github.com/en/pages/getting-started-with-github-pages/creating-a-github-pages-site
- Examples: https://github.com/collections/github-pages-examples

# GitHub – Safety and Etiquette

- **Do not upload sensitive information:** Never commit passwords, API keys, or any other sensitive information to public repos
- **READMEs and Licensing:** Create clear README files for every repo and the understanding of software licensing
- **Respectful Collaboration:** Be kind, understanding, and patient when collaborating on shared projects

# GitHub – Resources

- Command Line: MAC vs Windows
  https://tracer.gitbook.io/manual/support/command-line-mac-vs.-windows

- Is a Mac or Windows PC Better for Programming?
  https://medium.com/geekculture/is-a-mac-or-windows-pc-better-for-programming-d5556bf06f1#:~:text=The%20Operating%20System%3A%20macOS%20vs,benefit%20of%20macOS%20is%20security

- Installing Git
  https://git-scm.com/downloads

# GitHub – Getting started!

- Sign in or create an account at: https://github.com/
- Configure Git with your username and email (important for commit messages):

```
git config --global user.name "Your Name"
git config --global user.email "youremail@example.com"
```

- Create a new repository

```
git clone https://github.com/username/repository-name.git
```

- Make Changes and Commit

```
git add .
git commit -m "A descriptive message about the changes."
git push origin main
```

- Fetches Changes

```
git pull
```