INTRO TO MACHINE LEARNING

DSI SUDS SCHOLAR BOOTCAMP 2025
SLIDES BY NAKUL UPADHYA

Slides and Labs:
[HTTPS://GITHUB.COM/UPADHYAN/SUDS-2025-BOOTCAMP-FRIDAY](https://github.com/upadhyan/suds-2025-bootcamp-friday)

ABOUT ME

Nakul Upadhya

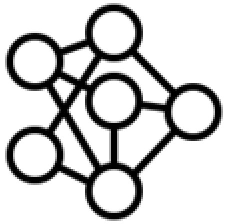
Nakul.Upadhya@mail.utoronto.ca

The Optimization and Machine Learning (OptiMaL) Lab

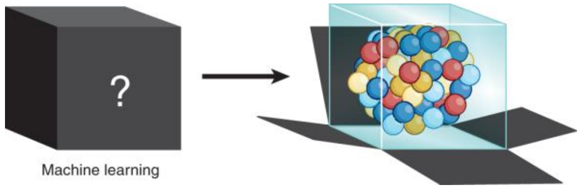
https://optimal.mie.utoronto.ca/

Research Areas

Deep Learning

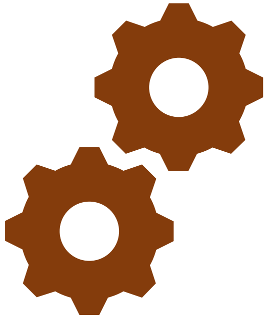


XAI



Applications

Manufacturing



Healthcare

LEARNING OUTCOMES

1. Grasp basic ML terminology
2. Understand the difference between Supervised and Unsupervised Learning
3. Construct your own ML Workflow

Slides and Labs:
[HTTPS://GITHUB.COM/UPADHYAN/SUDS-2025-BOOTCAMP-FRIDAY](https://github.com/upadhyan/suds-2025-bootcamp-friday)

PRELIMINARIES

WHAT IS MACHINE LEARNING?

Study of Algorithms that: [1]

- Improve their **performance**
- At some **task**
- With **experience**

[1] Tom Mitchell, CMU 10-601 slides

WHAT IS MODEL?

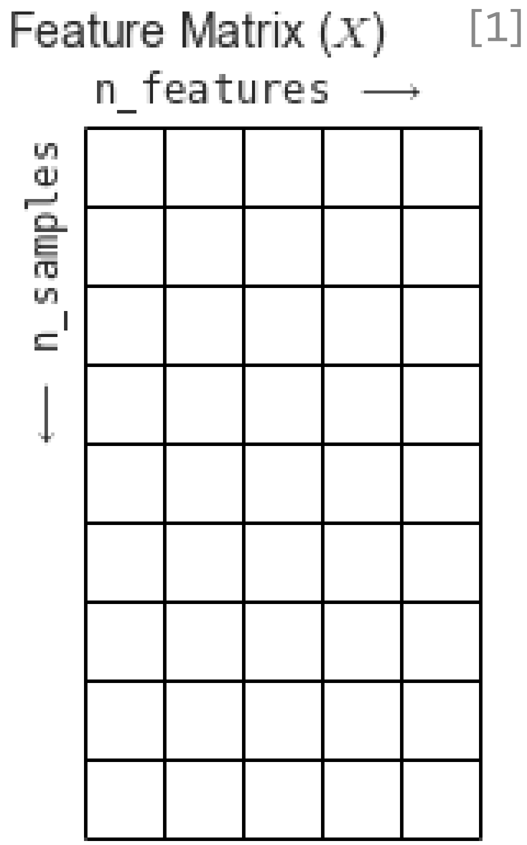
Study of Algorithms that: [1]

- Improve their **performance**
- At some **task**
- With **experience**

We call a trained algorithm a “Model”

[1] Tom Mitchell, CMU 10-601 slides

DATA AKA EXPERIENCE



- Sample: A datapoint
- Feature: an attribute of the samples

ML Algorithms learn relations between **features** across many **samples** to accomplish a task.

[1] Python Data Science Notebook – Introduction to Sci-Kit Learn – Jake Vanderplas

ML TASKS

Supervised

Uncover relations between the features and a **prediction target**

- Regression
- Classification

Semi-supervised

Self-supervised

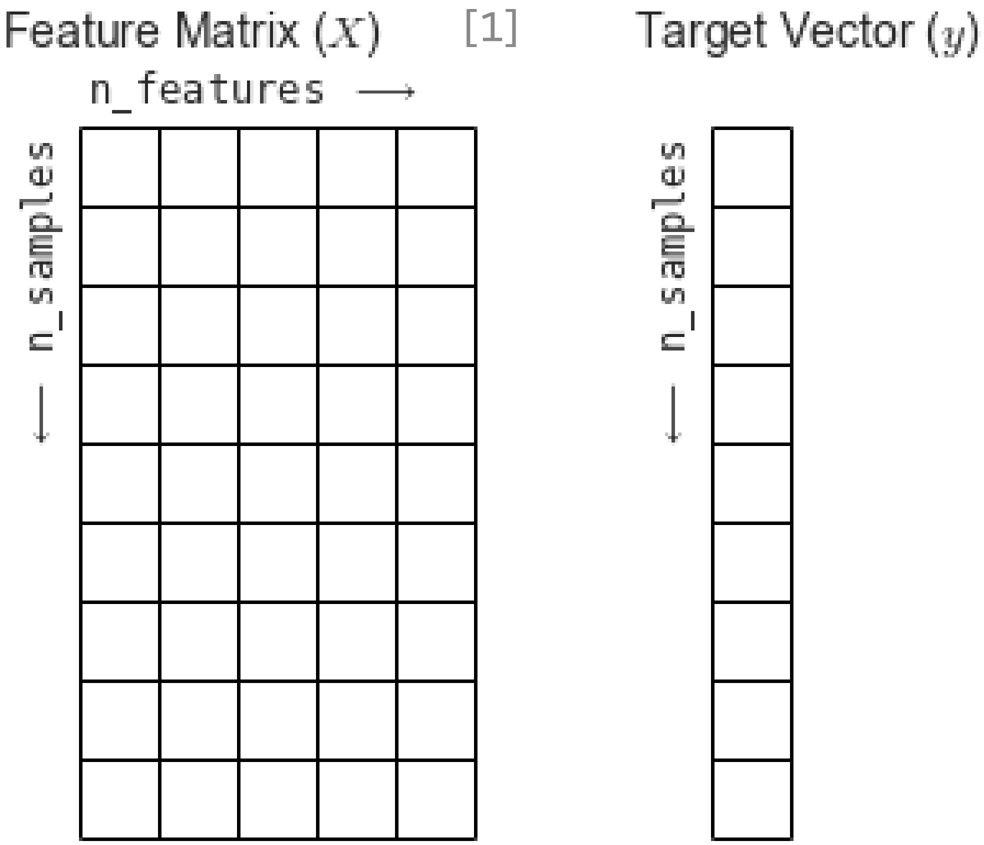
Unsupervised

Uncover hidden patterns **within the feature matrix**

- Clustering
- Dimensionality Reduction

SUPERVISED LEARNING

TERMINOLOGY



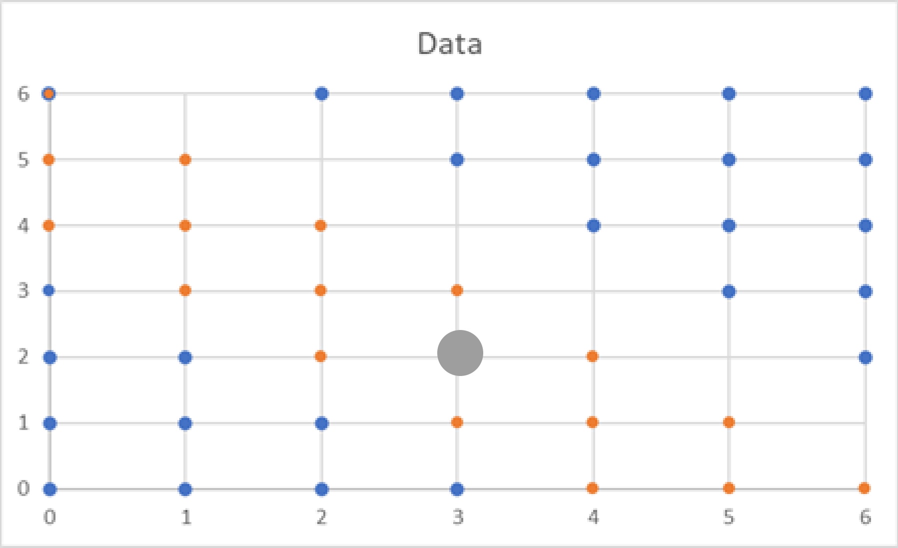
- Supervised Learning = Prediction
- Target: True Values

[1] Python Data Science Notebook – Introduction to Sci-Kit Learn – Jake Vanderplas

CLASSIFICATION VS. REGRESSION

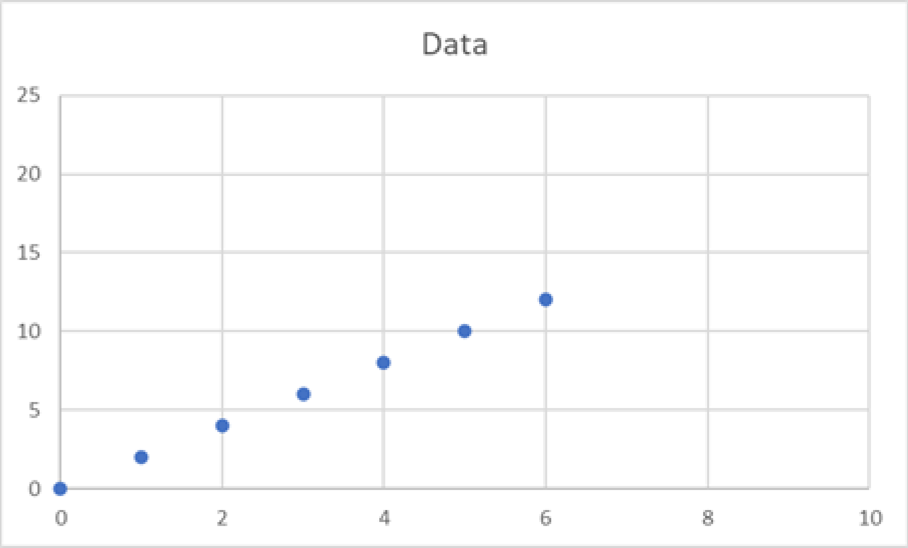
Classification

Categorical Target



Regression

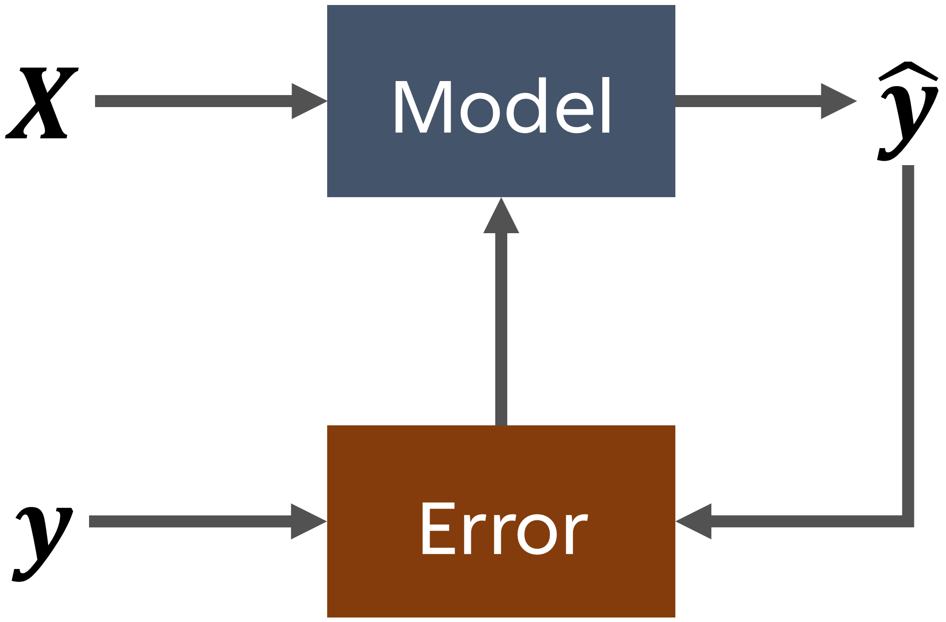
Numerical Target



CLASSIFICATION VS. REGRESSION

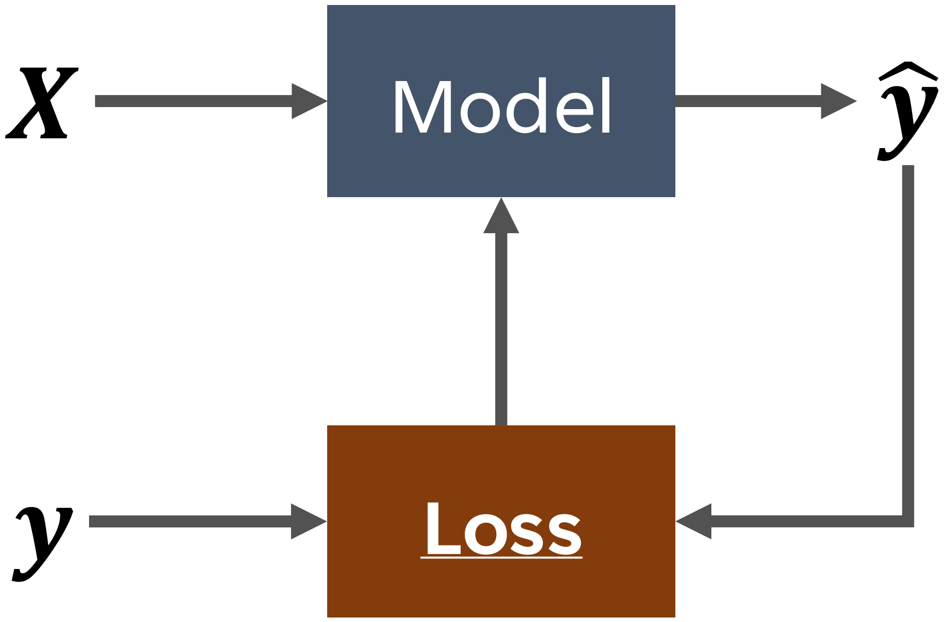
Classification	Regression
Categorical Target	Numerical Target
Blue vs. Orange	Age
Cancer vs. No Cancer	Income
Cat, Dog, or Bird	Sales

TRAINING A MODEL



- Make a prediction
- Calculate the error
- Update model based on error
- Repeat

TRAINING A MODEL



- Make a prediction
- Calculate the **loss**
- Update model based on **loss**
- Repeat

REGRESSION ERROR

Mean Squared Error

$$\frac{1}{n}\sum(y_i - \hat{y}_i)^2$$

Mean Absolute Error

$$\frac{1}{n}\sum|y_i - \hat{y}_i|$$

CLASSIFICATION ERROR

Misclassification

$$\frac{Incorrect}{Total}$$

Cross-Entropy

$$-\frac{1}{n}\sum\sum y_{i,c}\log\hat{p}_{i,c}$$

Impurity

$$1-\sum p_c(1-p_c)$$

EXAMPLE: MULTIPLE LINEAR REGRESSION

$$\hat{y} = f(x) = \theta_0 + \theta_1 x_1 + \cdots + \theta_n x_n = \Theta \cdot x$$

Minimize $\frac{1}{n}\sum(y_i - \Theta \cdot x)^2$

By Updating Θ

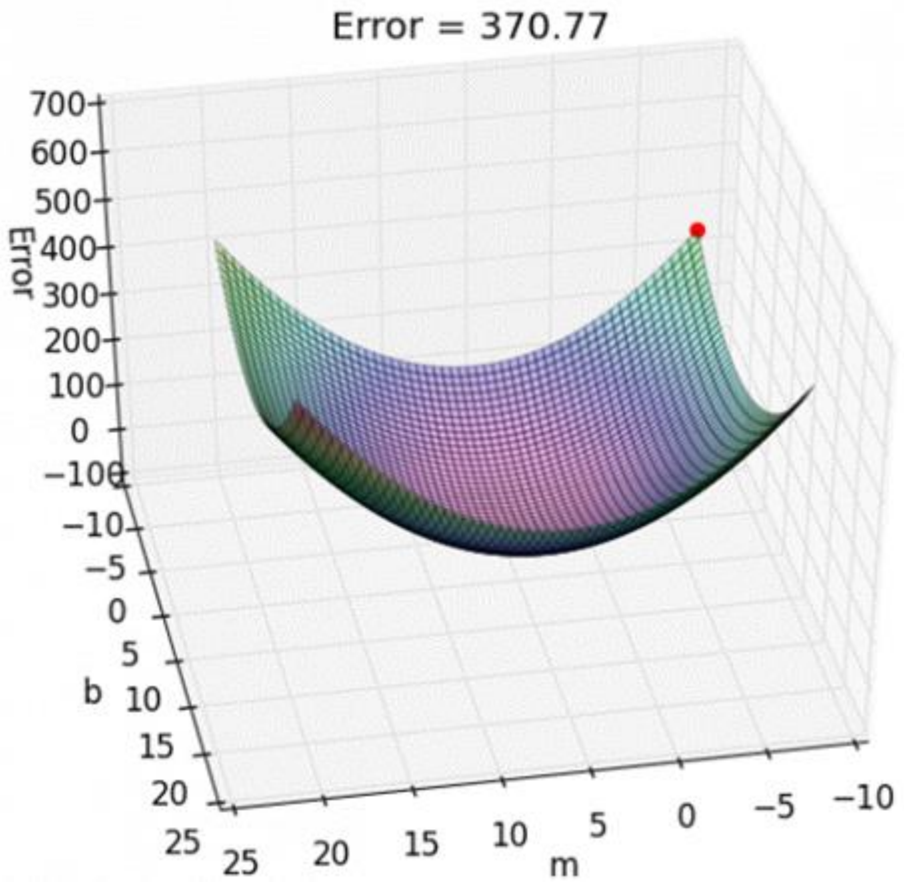
EXAMPLE: MULTIPLE LINEAR REGRESSION

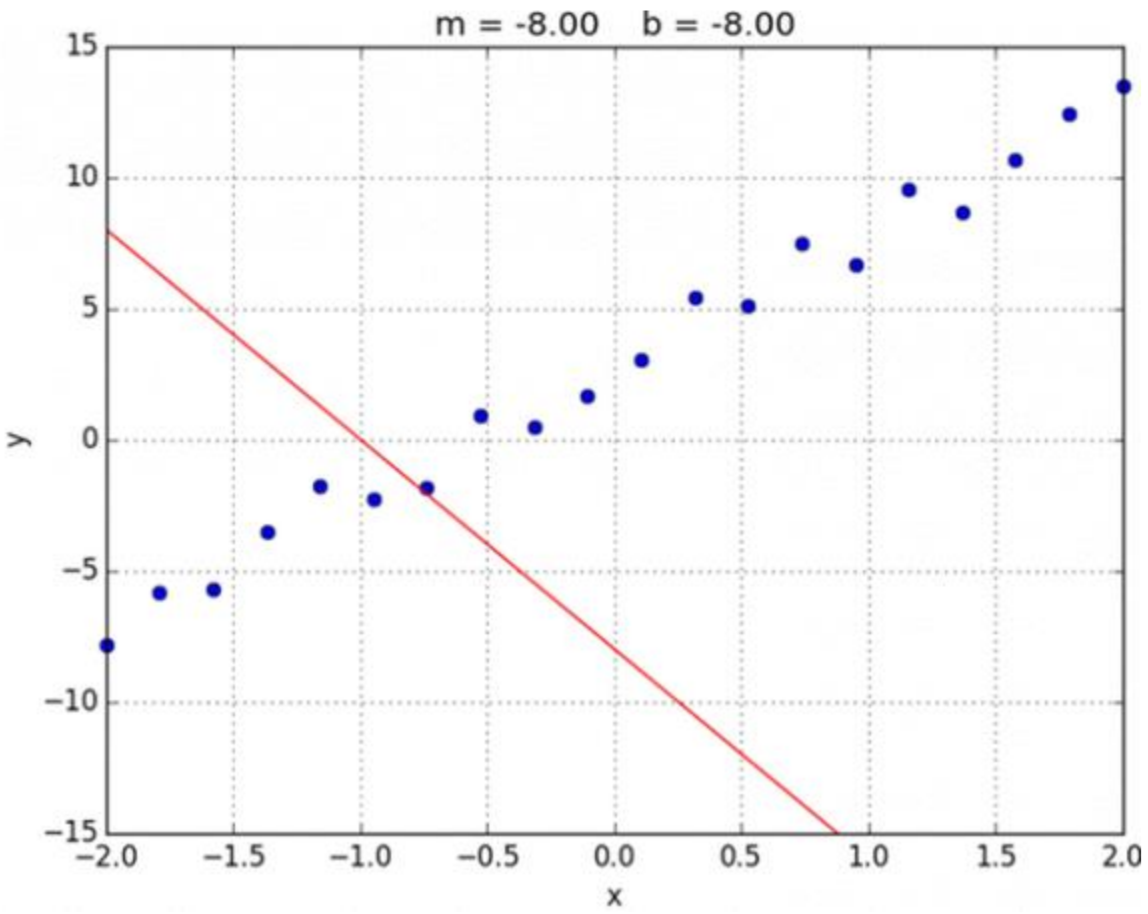
$$\hat{y} = f(x) = \theta_0 + \ \theta_1 x_1 + \cdots + \theta_n x_n = \Theta \cdot x$$

$$\nabla L = \frac{\partial}{\partial \Theta} \sum (y_i \ - \Theta \cdot x)^2$$ Gradient = Direction of Increase

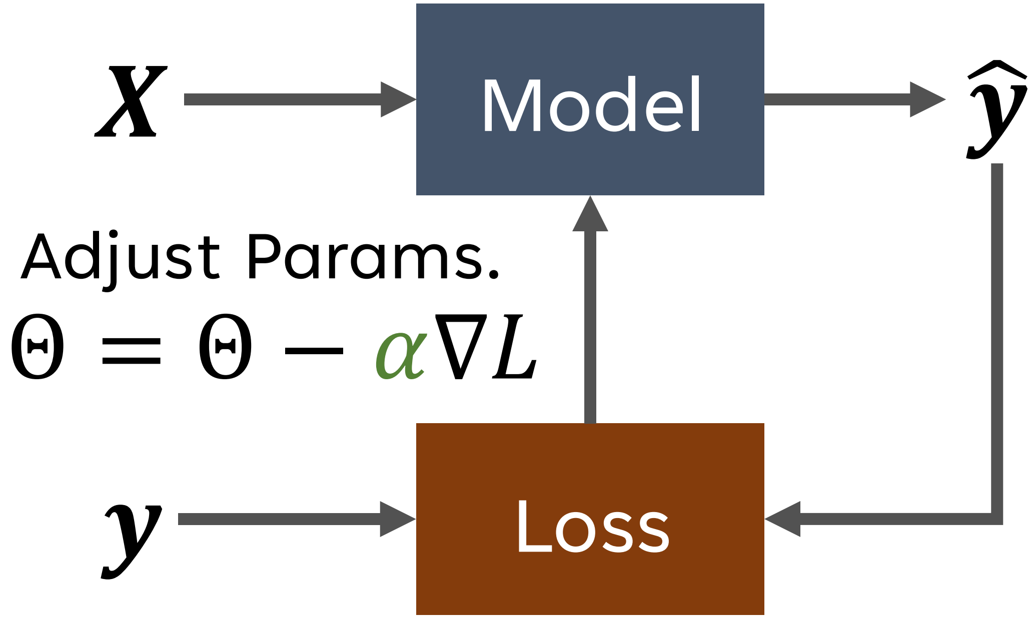
$$\Theta = \Theta - \alpha \nabla L$$ Take steps in opposite direction

EXAMPLE: LINEAR REGRESSION

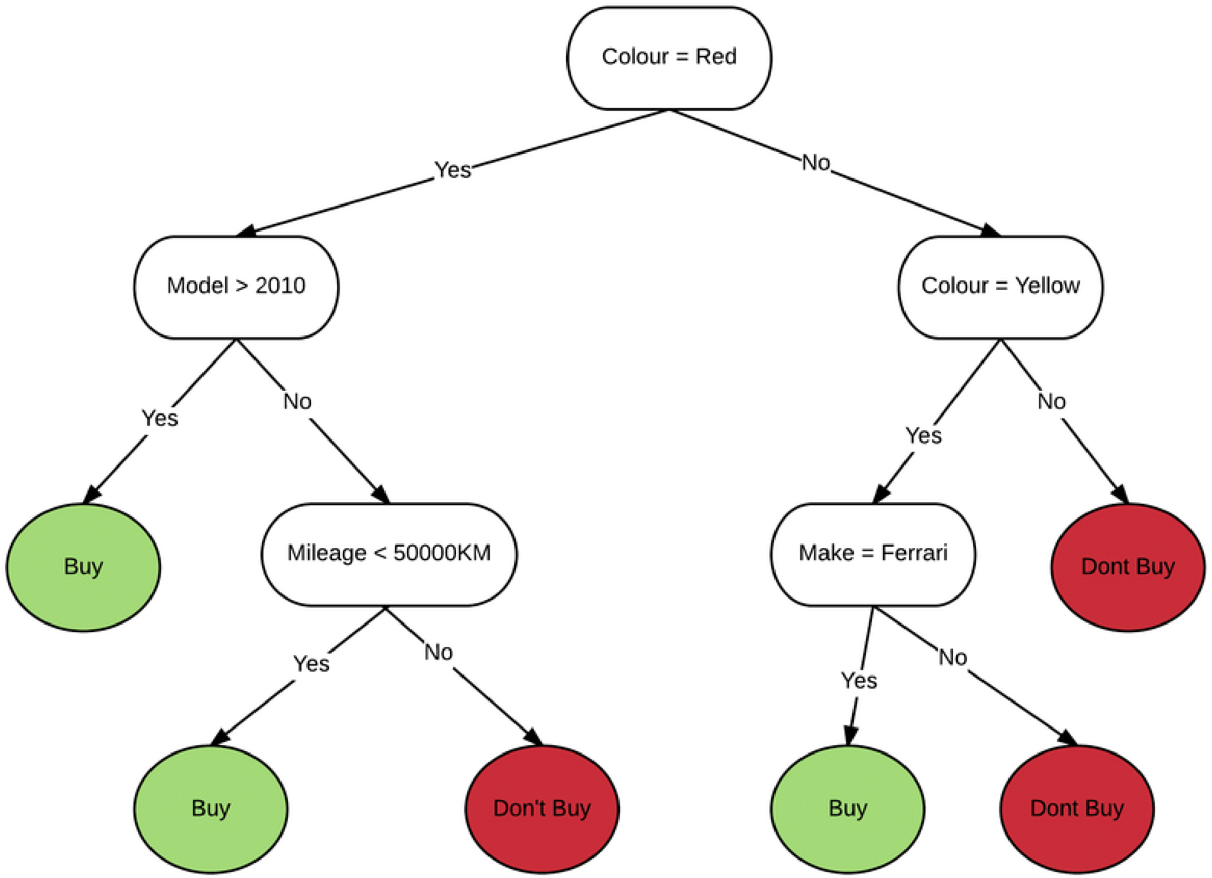




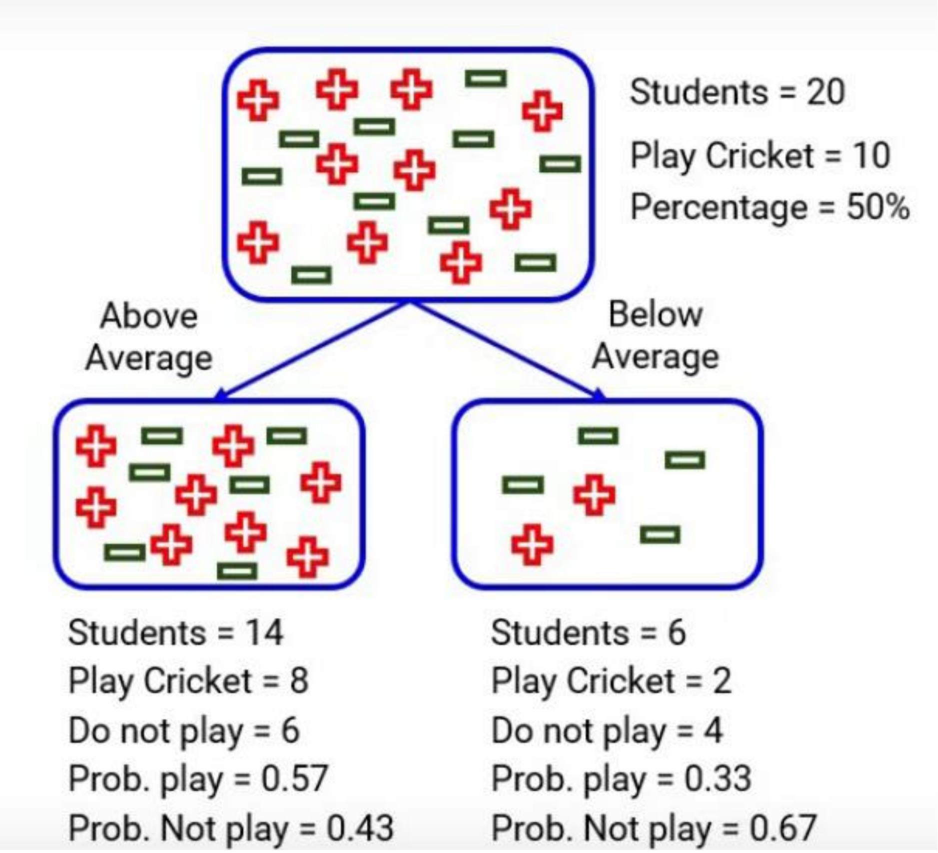
EXAMPLE: MULTIPLE LINEAR REGRESSION



EXAMPLE: DECISION TREE CLASSIFIER



EXAMPLE: DECISION TREE CLASSIFIER

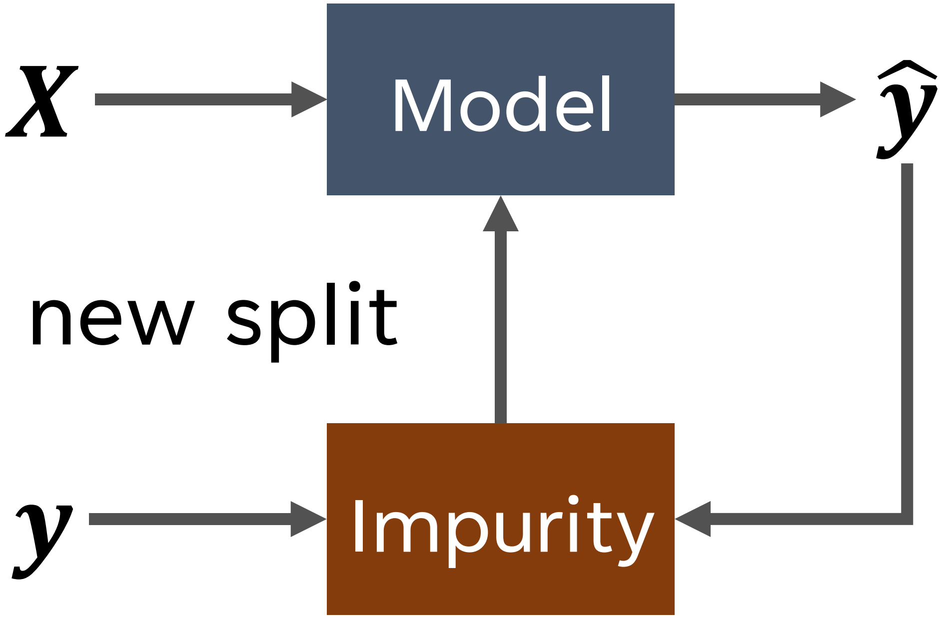


Impurity

$$1 - \sum p_c (1 - p_c)$$

https://www.analyticsvidhya.com/blog/2021/03/how-to-select-best-split-in-decision-trees-gini-impurity/

EXAMPLE: DECISION TREE CLASSIFIER

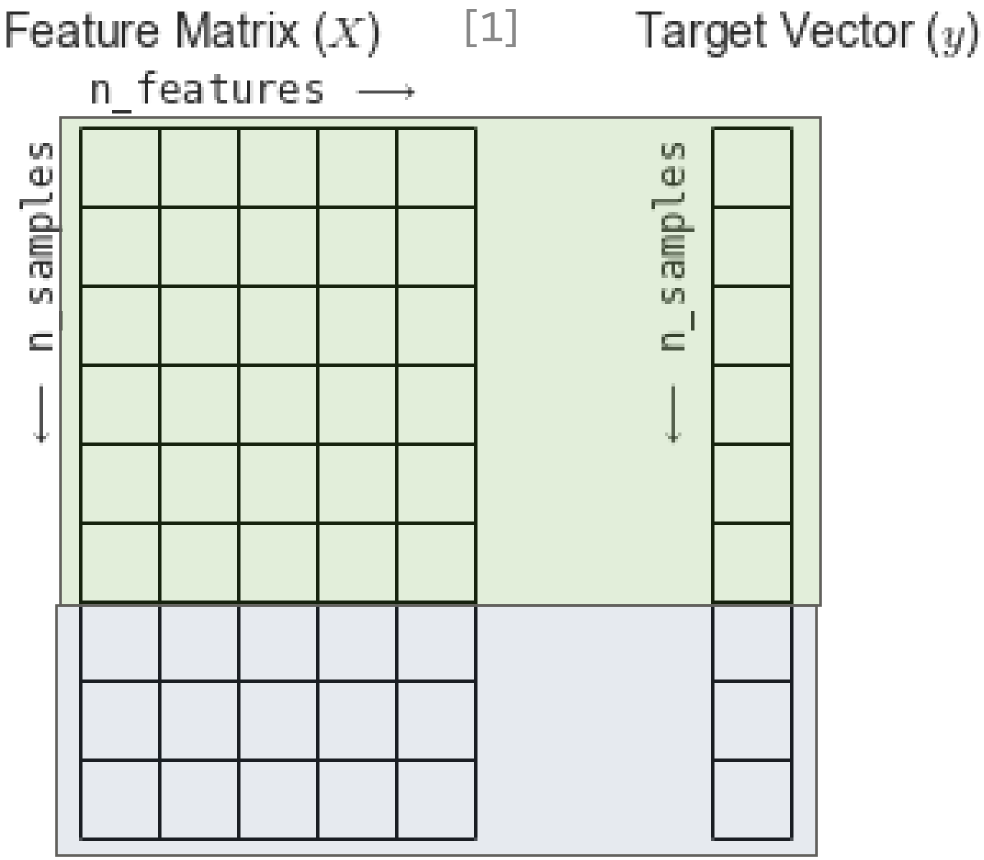


EVALUATING A TRAINED MODEL

How do we accurately judge how good a model is?

Evaluating error during training is not a good judge.

EVALUATING A MODEL

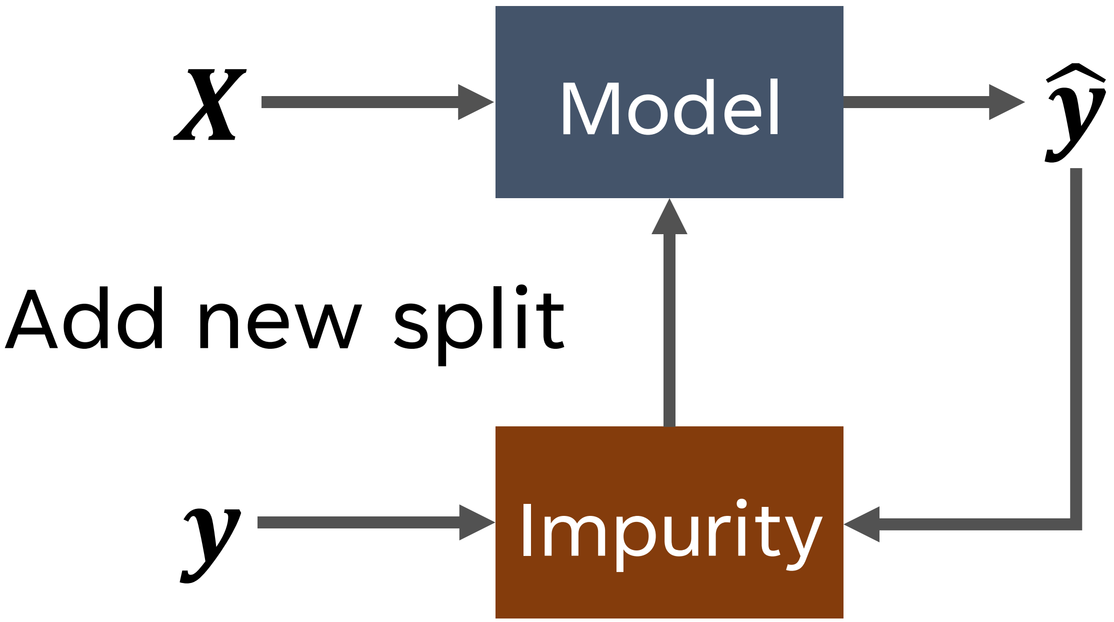


- Split the data up
- Training Set: Used to develop and train model
- Test Set: Used to evaluate model

[1] Python Data Science Notebook – Introduction to Sci-Kit Learn – Jake Vanderplas

PROBLEM: HYPERPARAMETERS

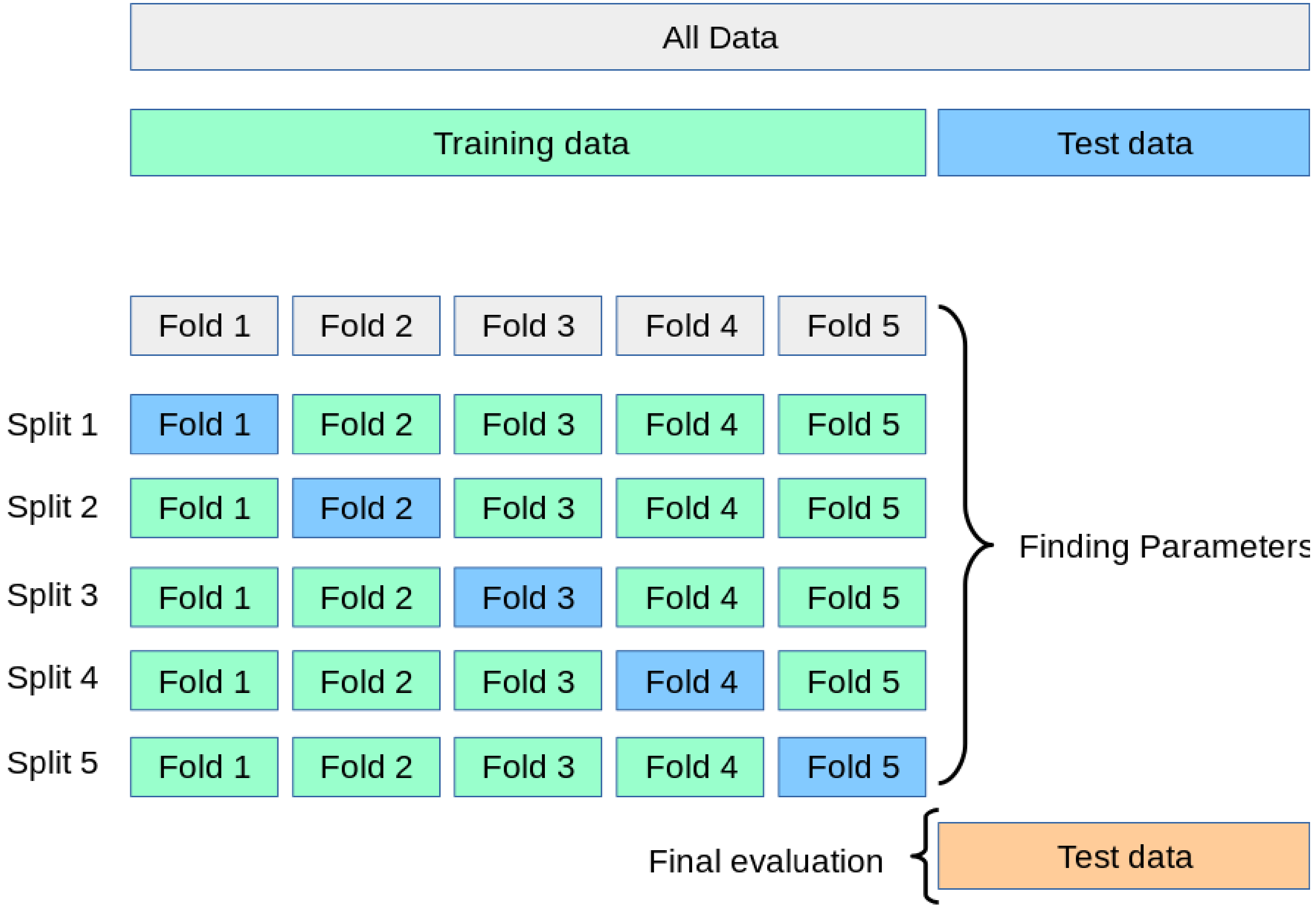
Recall training a Decision Tree



How many splits do we make?

How many points do we want to be in each split?

K-FOLD CROSS VALIDATION



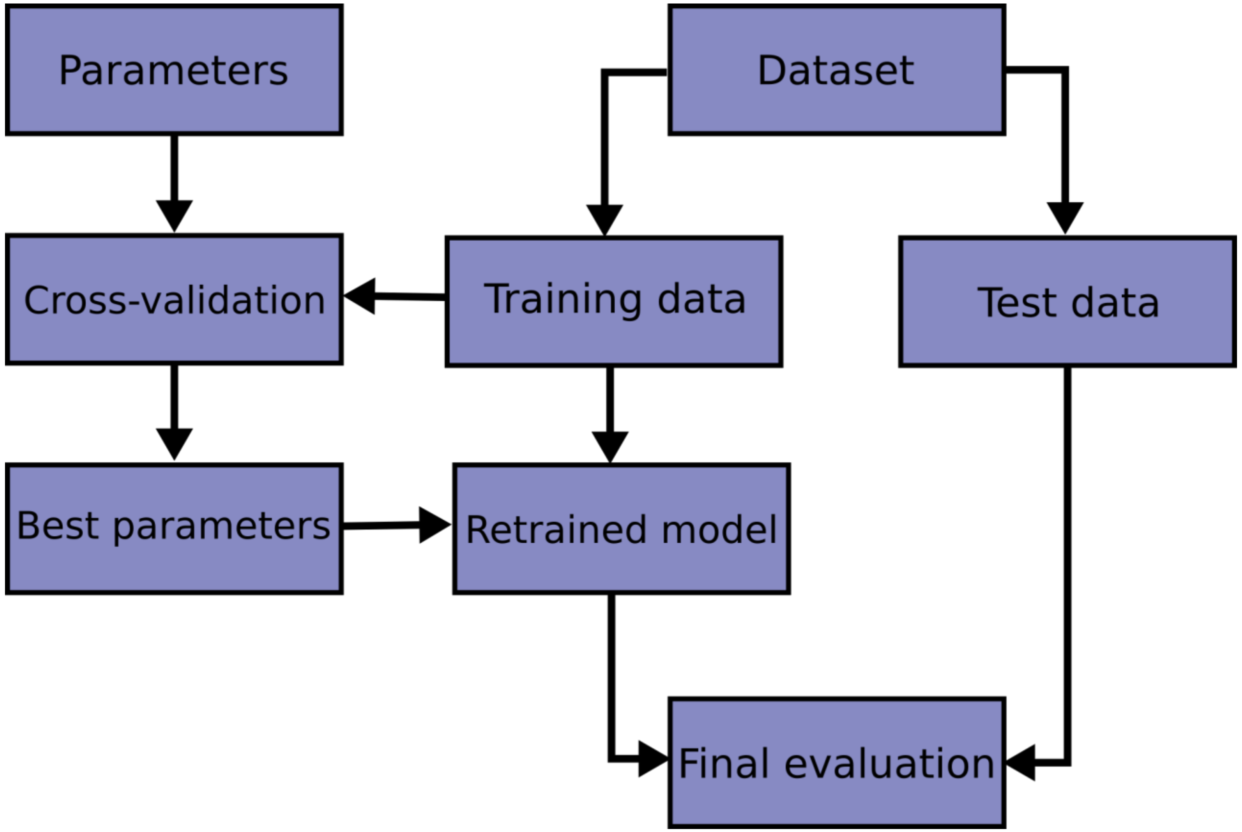
Train on all folds but one

Evaluate on last fold

Repeat with a different split

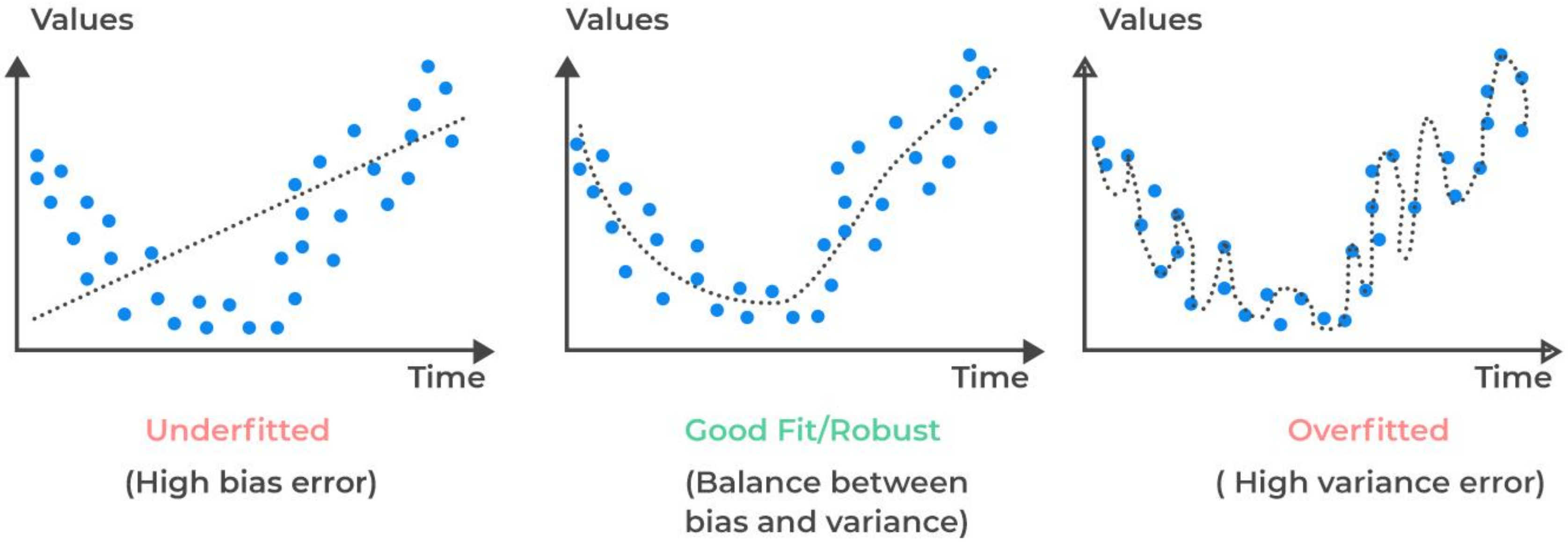
https://scikit-learn.org/stable/modules/cross_validation.html

SUPERVISED LEARNING PIPELINE



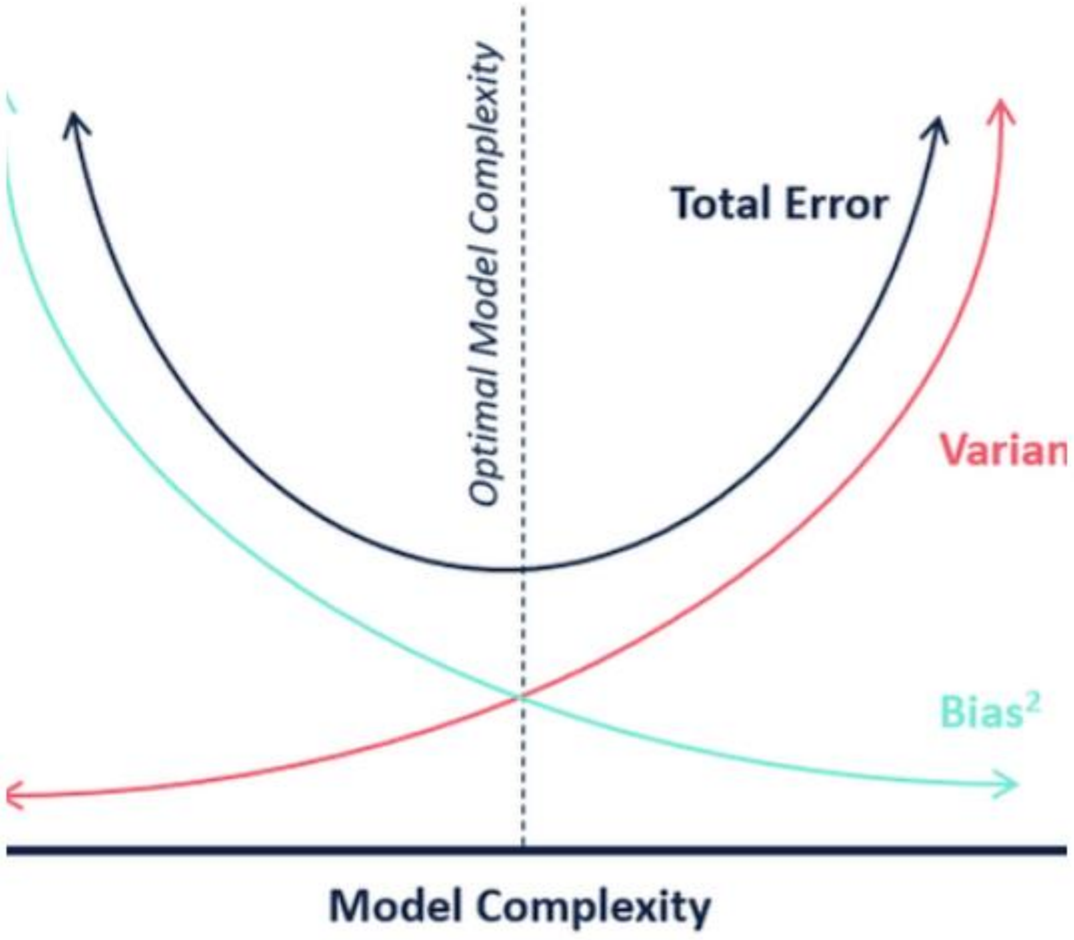
https://scikit-learn.org/stable/modules/cross_validation.html

UNDER/OVER FITTING



https://analystprep.com/study-notes/cfa-level-2/quantitative-method/overfitting-methods-addressing/

BIAS-VARIANCE TRADEOFF



https://www.analyticsvidhya.com/blog/2020/08/bias-and-variance-tradeoff-machine-learning/

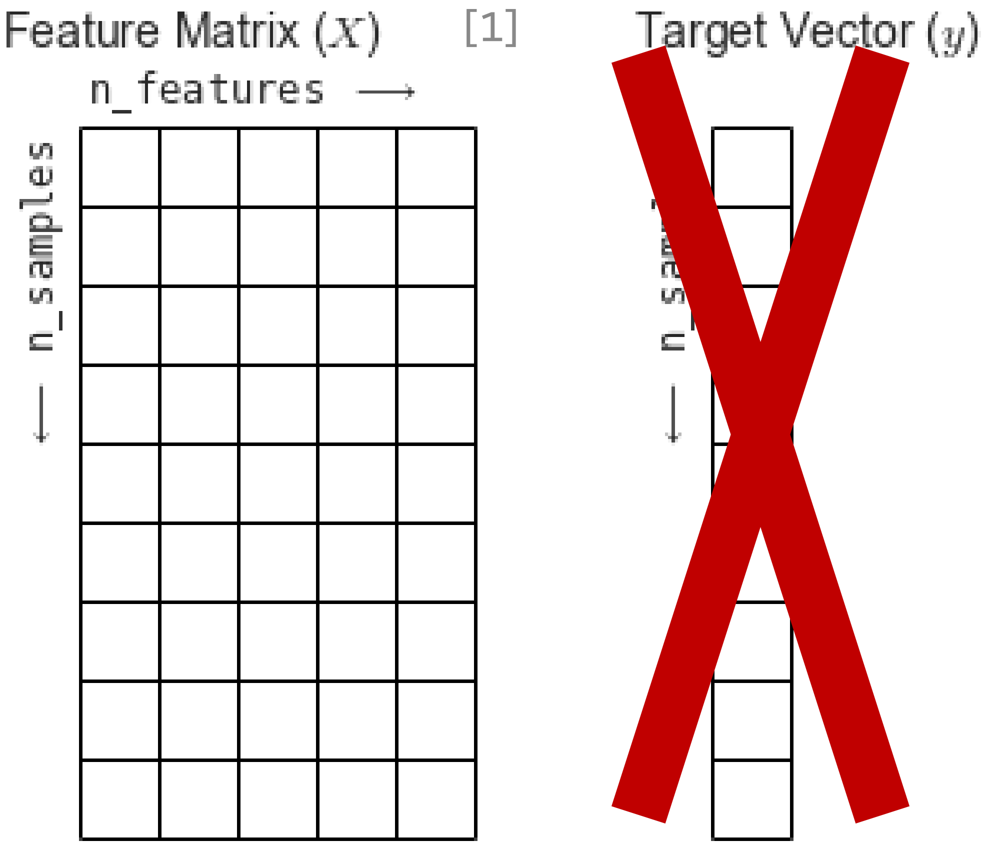
WORKSHOP
HTTPS://GITHUB.COM/UPADHYAN/SUDS-2025-BOOTCAMP-FRIDAY

UNSUPERVISED LEARNING

THOUGHT EXERCISE

What is the correct grouping of these pictures?

UNSUPERVISED SETTING

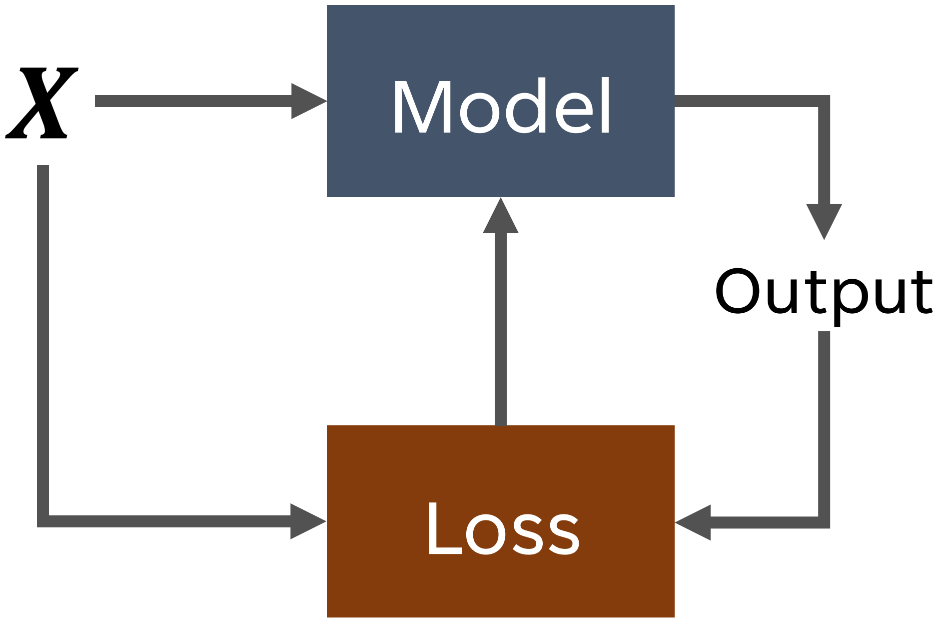


There are no "targets"

Unsupervised Learning = Finding patterns in the features

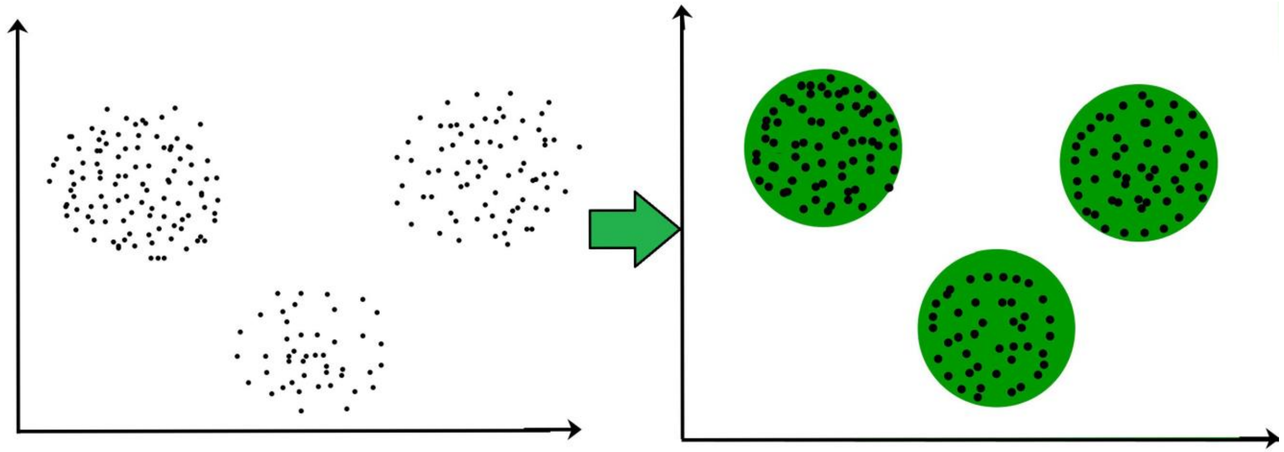
[1] Python Data Science Notebook – Introduction to Sci-Kit Learn – Jake Vanderplas

TRAINING A MODEL



- Loss is defined only by input points and model output
- ***No True Labels***

CLUSTERING



Find Groups in the data

Creating teams based on personality scores and skills

Identifying customer profile groups

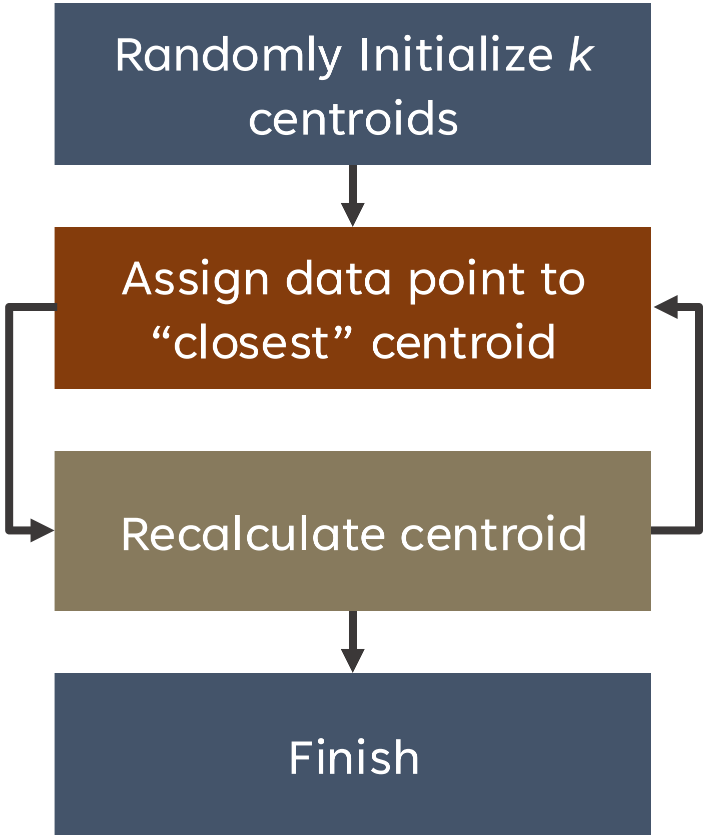
EXAMPLE: K-MEANS CLUSTERING

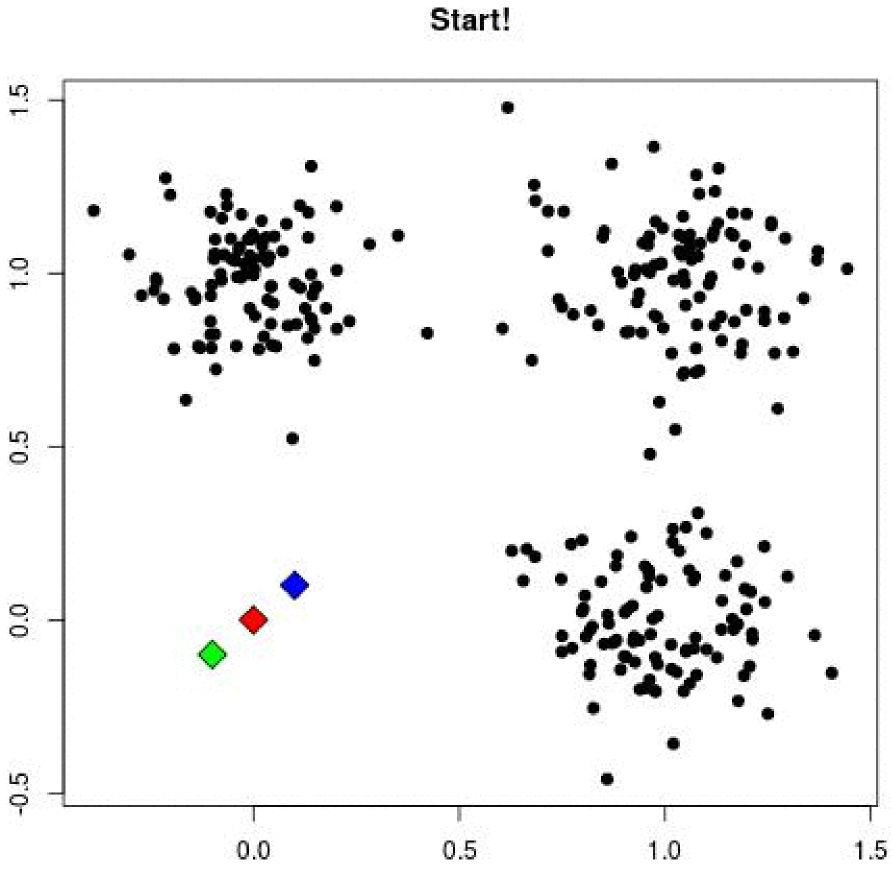
Goal: Group the datapoints into K Clusters

$$\text{Minimize } \frac{1}{n}\sum\sum w_{i,k}\|x_i - z_k\|^2$$

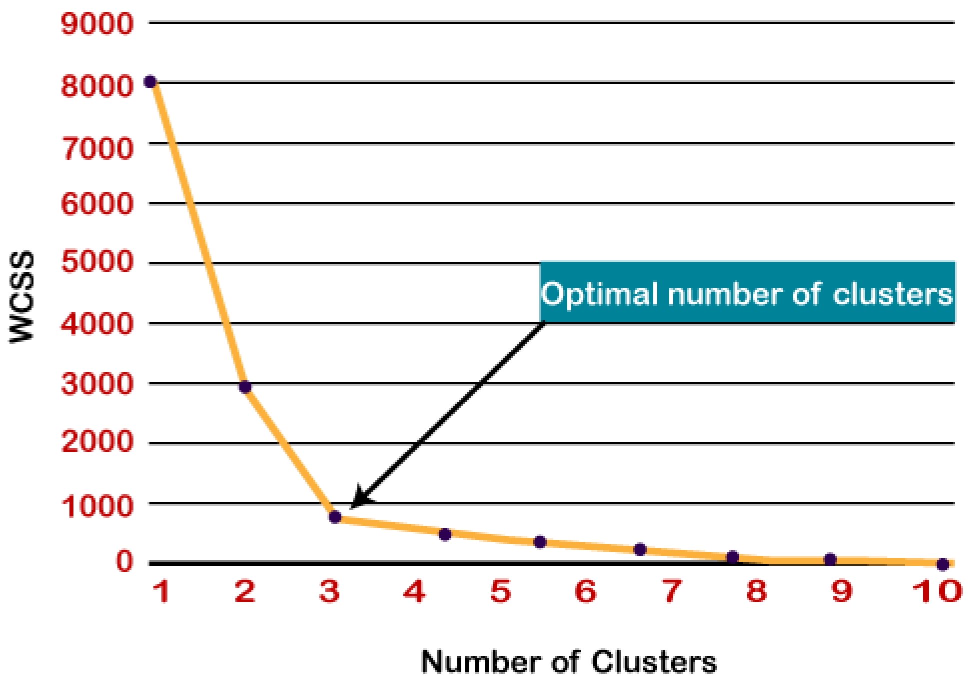
- $w_{i,k}$: is point *i* in cluster *k*
- z_k: Average of all points in cluster *k* (centroid)

EXAMPLE: K-MEANS CLUSTERING



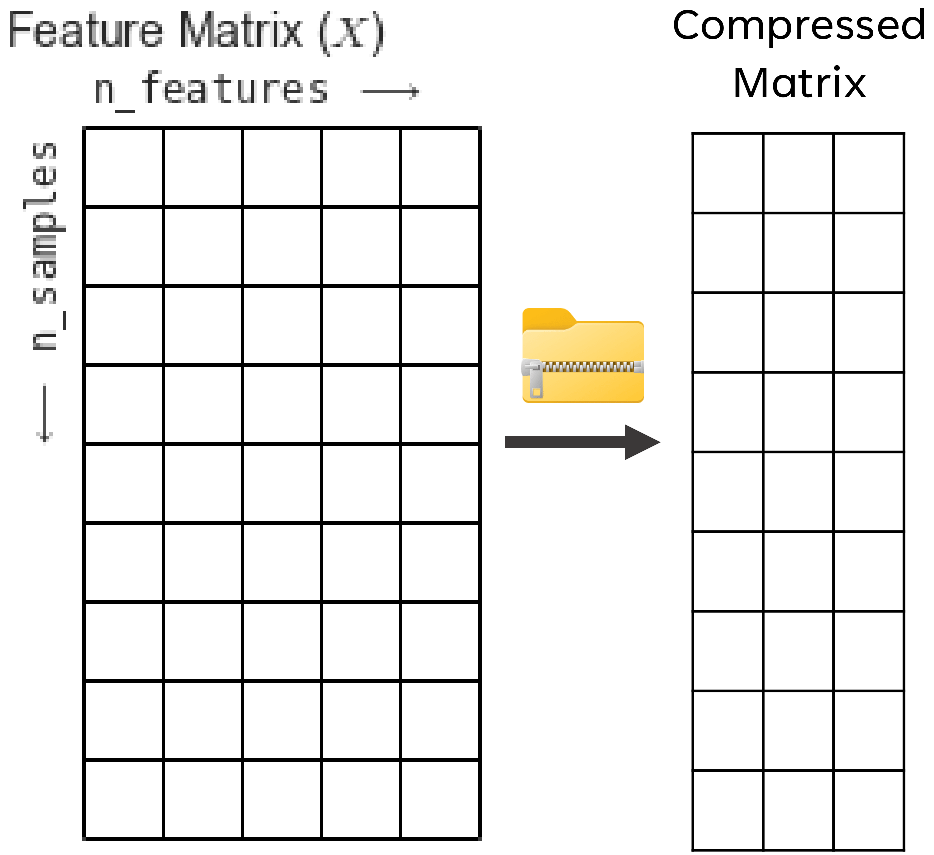


EXAMPLE: K-MEANS CLUSTERING



- Test different number of clusters and plot inertia
- Find the “elbow”

DIMENSIONALITY REDUCTION

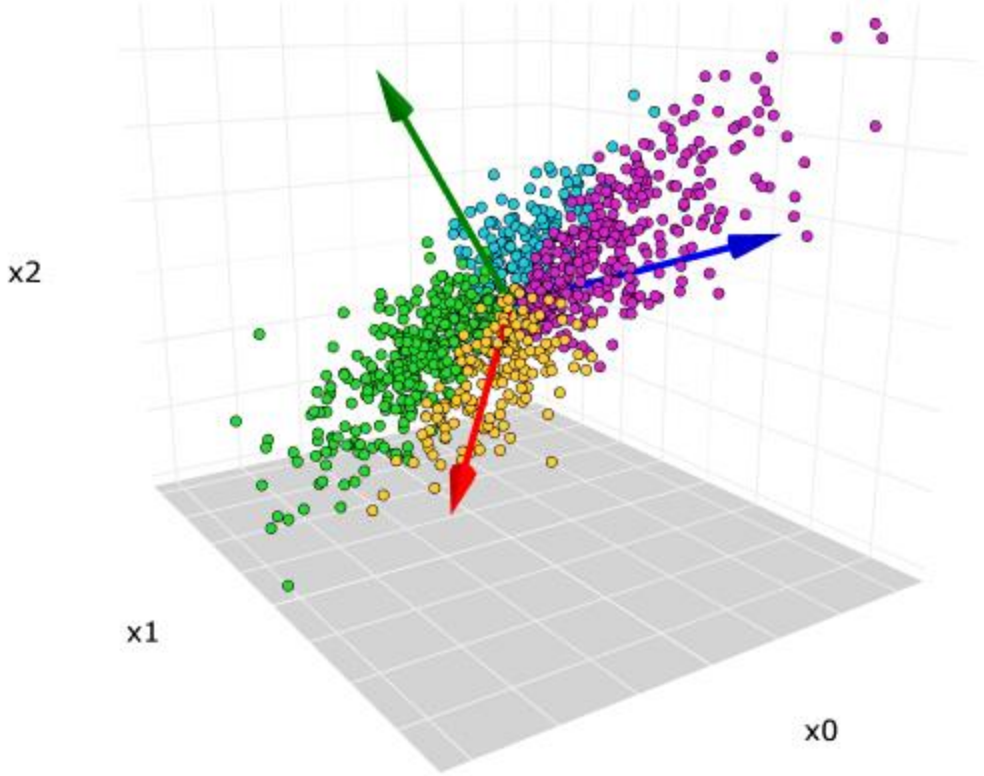


Find Groups in the data

Creating teams based on personality scores and skills

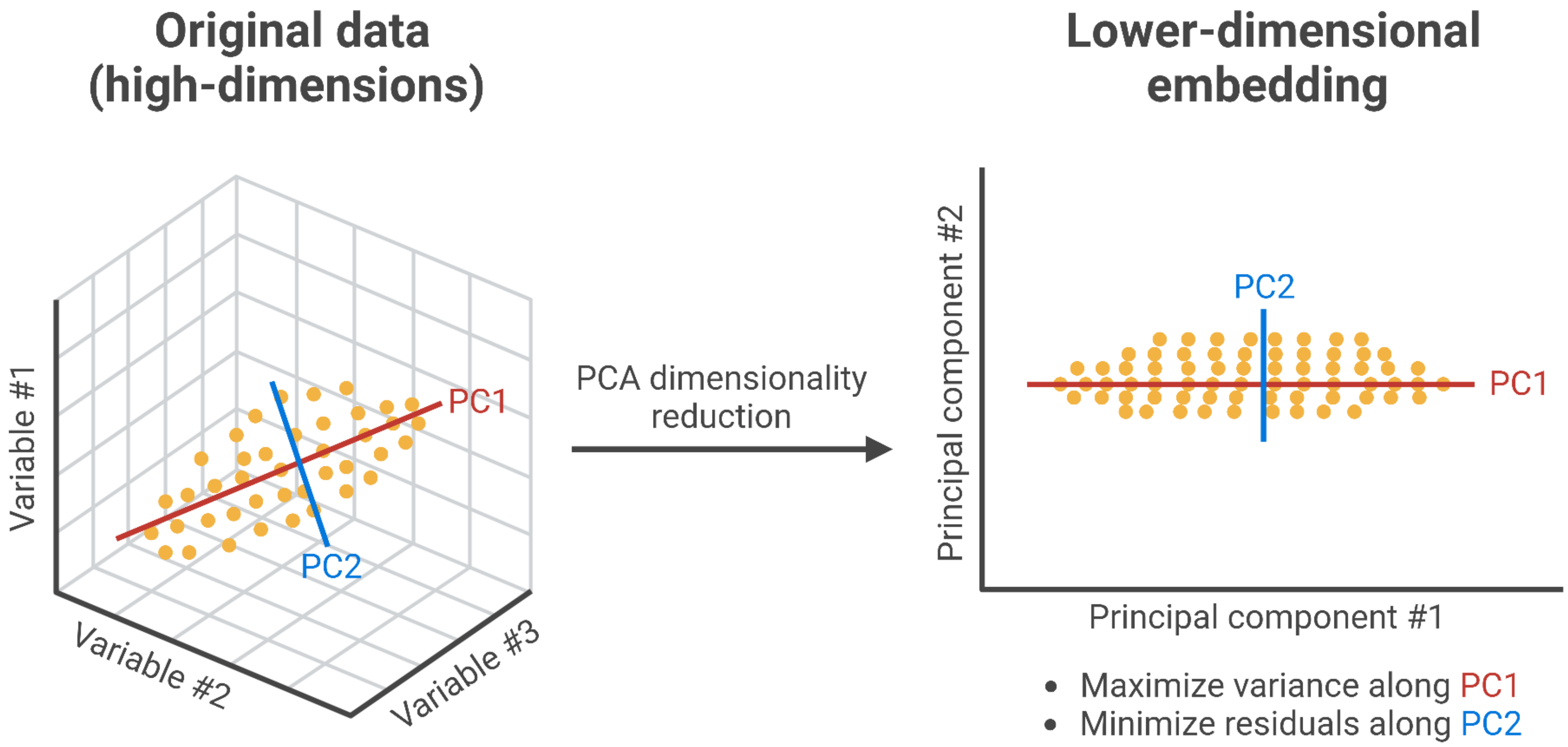
Identifying customer profile groups

EXAMPLE: PCA



- Find the “components” that capture the most variance in the data
- # of Components <= # of Features

EXAMPLE: PCA


Original data
(high-dimensions)
Variable #1
Variable #2
Variable #3
PC1
PC2
PCA dimensionality reduction
Lower-dimensional embedding
Principal component #2
Principal component #1
PC2
PC1
• Maximize variance along PC1
• Minimize residuals along PC2

WORKSHOP
HTTPS://GITHUB.COM/UPADHYAN/SUDS-2025-BOOTCAMP-FRIDAY