

# Unsupervised Machine Learning with Python

# Section 1.1: Introduction

# What is Machine Learning?

Definition (adapted from Wikipedia page on Machine Learning)

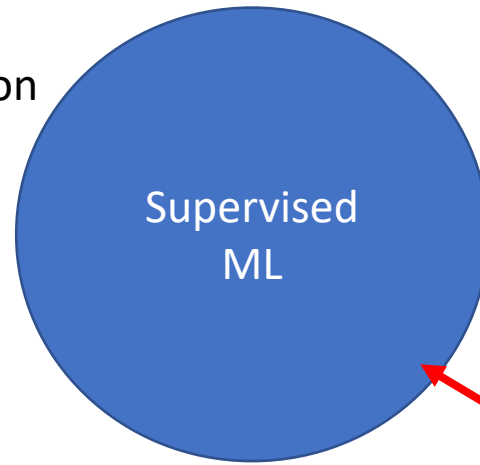
- Machine Learning is the study of algorithms and statistical models that computer systems use to perform a specific task without using explicit instructions, relying on patterns and inference instead

# Types of Machine Learning

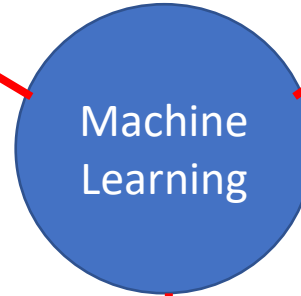
Fit function to data  
Use function for prediction

Applications:

- House Price prediction
- Image Classification
- Spam Filtering
- Language Translation



Supervised  
ML



Machine  
Learning



Unsupervised  
ML

Find patterns in data

Applications:

- Customer Segmentation
- Document Clustering
- Image Grouping
- Anomaly/fraud detection

Find strategy to maximize cumulative reward

Applications:

- Industrial Control
- Robotics
- Game Playing (chess, video games, etc)



Reinforcement  
ML

# Unsupervised Machine Learning

Definition (see Wikipedia page on Unsupervised Learning)

- Unsupervised learning is a type of machine learning that looks for previously undetected patterns in a dataset with no pre-existing labels and with a minimum of human supervision.

# Algorithms Covered in this Course

Course covers two broad categories of Unsupervised ML Algorithms:

Clustering:

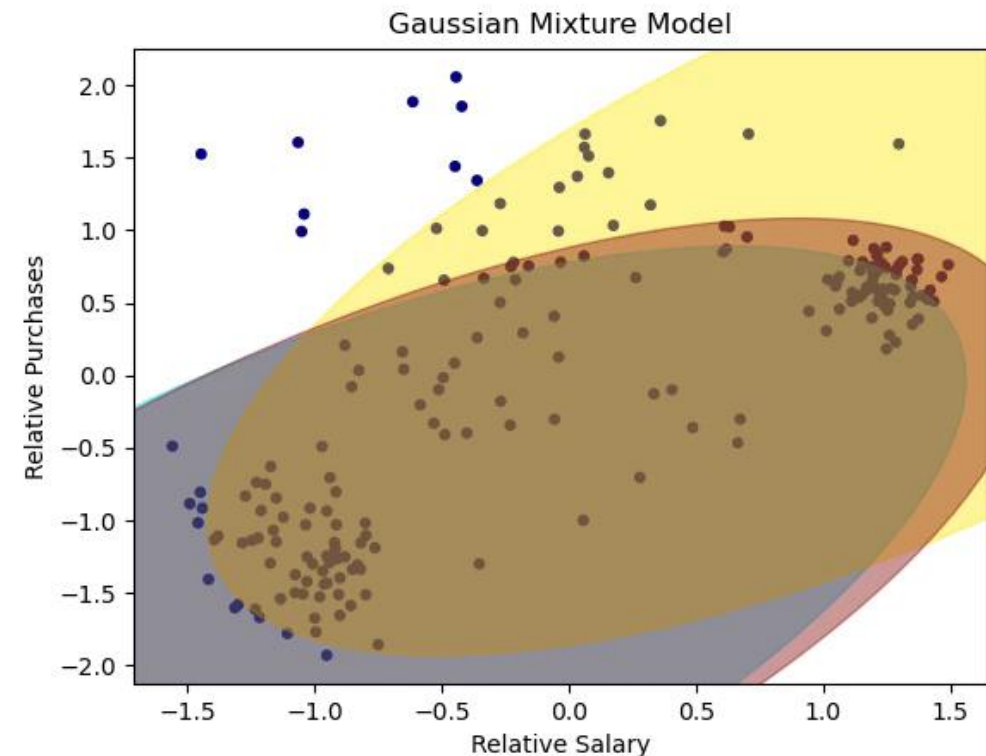
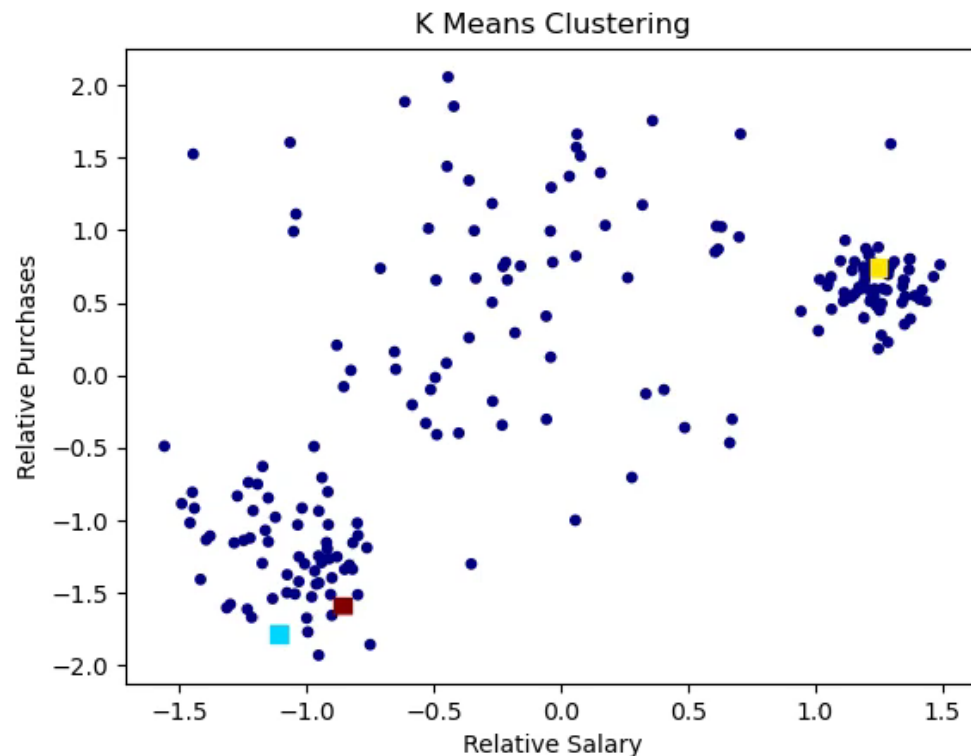
- Hierarchical, DBSCAN, K Means, Gaussian Mixture Model

Dimension Reduction:

- Principal Component Analysis
  - Reduce dimensions of data to speed up calculations

# Clustering Algorithms

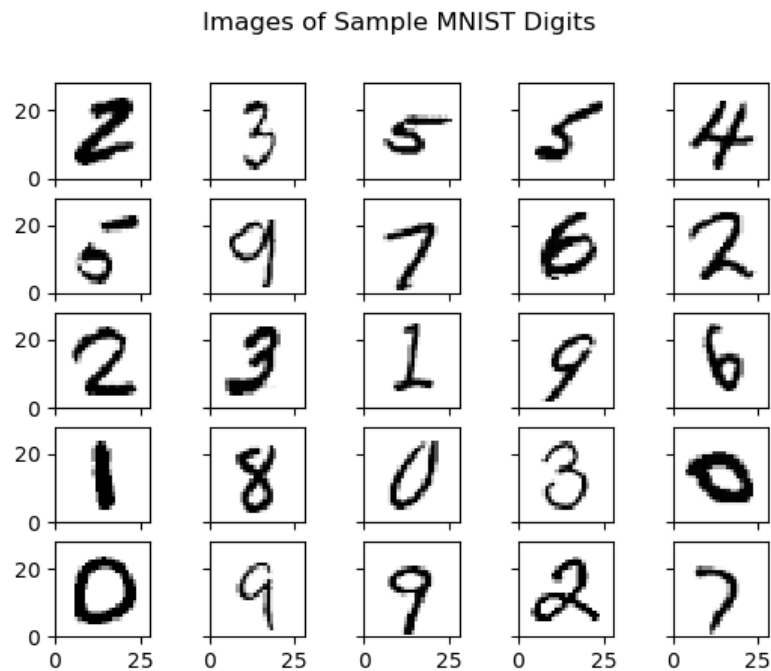
- Example: Find clusters in customer data
- Purpose: create specialized marketing campaigns for each cluster



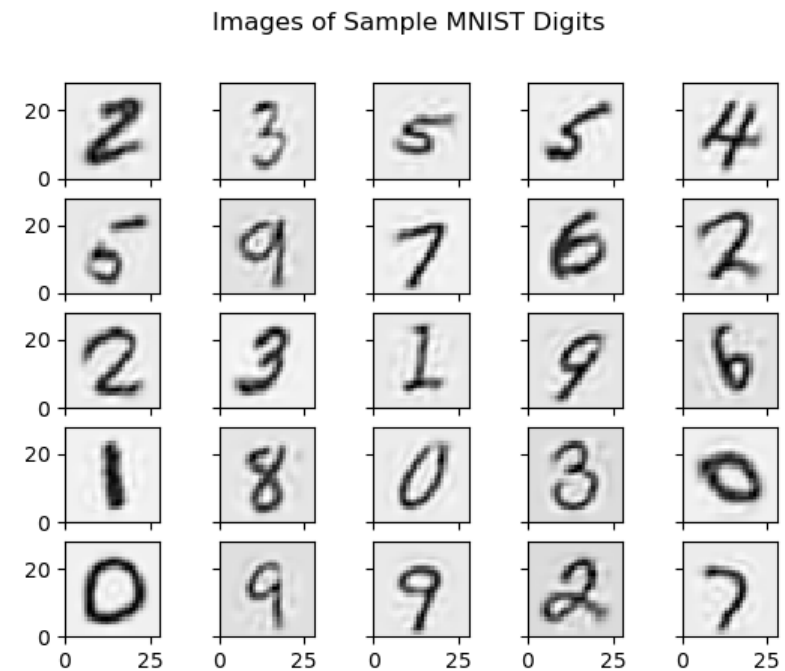
# Dimension Reduction Algorithm

- Example: Reduce dimension of images (while retaining essential features)
- Purpose: use to speed up calculations

Original: 784 dimensions

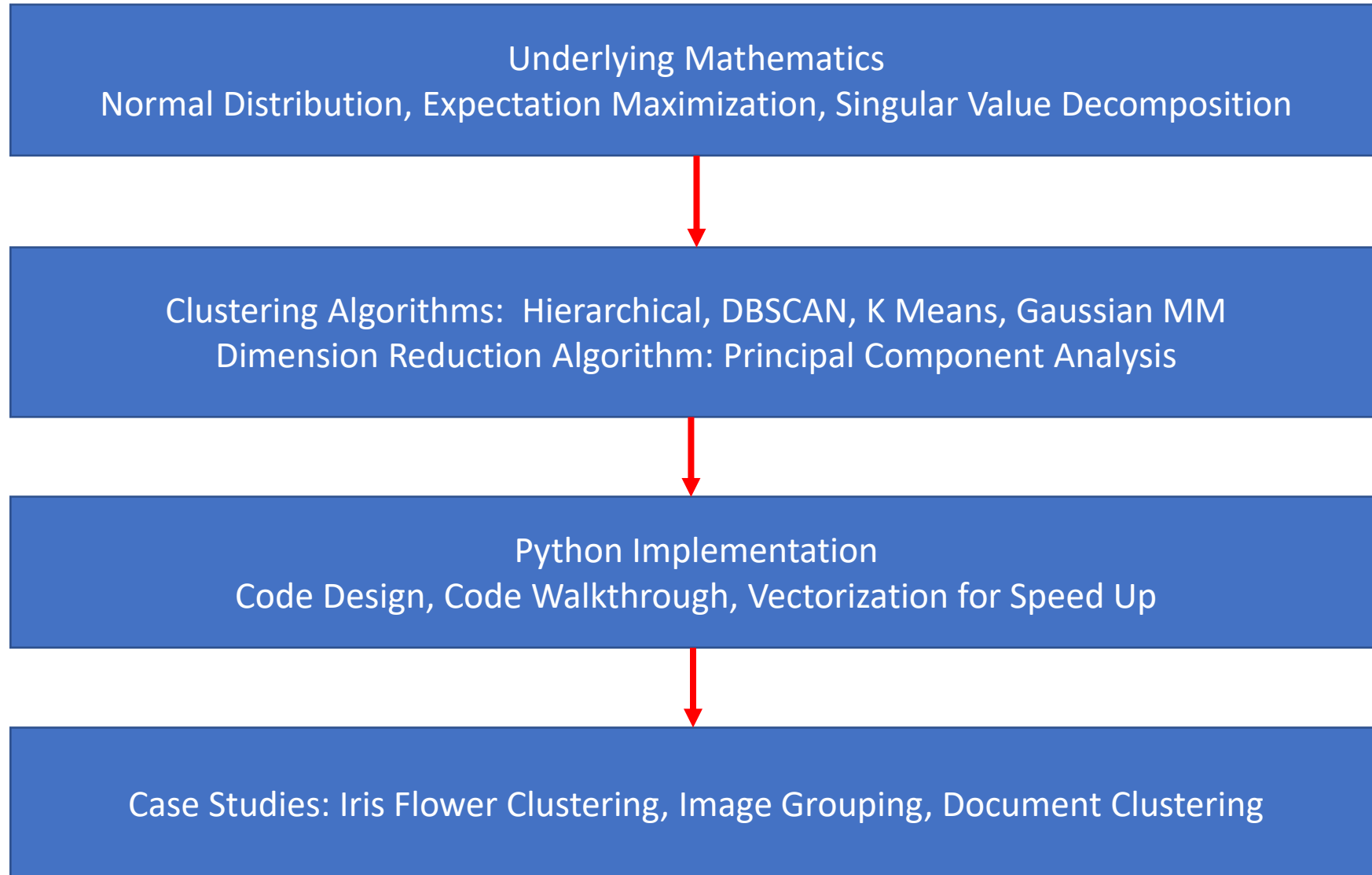


Reconstructed: using 78 principal components/dimensions





# What is in this Course?



# Course Approach

1. Video Lectures
2. Jupyter Notebook Demos
  - Present examples of python functionality of packages used in course
3. Code Design and Walkthrough
  - Explain code design and python implementation of algorithms
4. Exercises with Solutions
  - Give students opportunities for additional practice

# Course Outline:

- Section 01: Introduction
- Section 02: Python Demos
- Section 03: Review of Mathematical Concepts
- Section 04: Hierarchical Clustering
- Section 05: DBSCAN
- Section 06: K Means Clustering
- Section 07: Gaussian Mixture Model
- Section 08: Comparison of Clustering Algorithms
- Section 09: Dimension Reduction
- Section 10: Case Studies
- Section 11: Summary and Thank You

# Unsupervised Machine Learning with Python

# Section 1.2: About this Course

# Audience for this Course

This course is suitable for

1. Students interested in machine learning/data science without any previous experience
2. Students who have knowledge of the subject and would like a refresher and/or gain a more detailed understanding of the math, algorithms, and code development

# Course Prerequisites

## Linear Algebra

- Should be familiar with vectors, transpose, matrices, matrix multiplication, inverses, determinants, linear spaces

## Probability and Statistics

- Should be familiar with basic probability and statistics, including mean, covariance, normal distributions

## Python Programming

- Should be able to write Python 3 programs and run them in Jupyter notebooks and in the command window
- We will review functionality from relevant Python packages in Section 2

## Multivariable Calculus (Optional)

- Students familiar with multivariable calculus will be able to follow the derivation of Gaussian Mixture Model approach presented in a separate PDF document

# How to Get the Most from this Course

Learning is not a spectator activity – active participation is required!!!

1. Take notes as you go through the material
2. Do the coding
  - Design videos describe code structure
  - Walkthrough videos show the code implementation
  - Students can do design and code development without watching these videos or can review videos first, and then do the implementation
3. Do the exercises
  - Solutions provided
4. Ask questions on forums



# Why Code from Scratch?

Why code from scratch when there are many Python packages for Unsupervised ML?

- It is my fundamental belief that to truly understand what is going on in an algorithm one must code from scratch!

# Unsupervised Machine Learning with Python

# Section 1.3: Course Resources and Set Up

# Course Github Site

<https://github.com/satishchandrareddy/UnsupervisedML/>

- Code
  - Programs developed in this course for Unsupervised ML
  - Data for case studies
- Examples
  - Jupyter notebook demos
- Exercises
  - PDF file of exercises and solutions
  - Solution files (Jupyter notebooks and python program files)
- Presentations
  - PDF files of presentations
- Resources
  - UnsupervisedML\_Resources.pdf: references and links to additional resources
  - UnsupervisedML\_GMM.pdf: derivation for Expectation Maximization for Gaussian Mixture Model

# Resources File

File: UnsupervisedML/Resources/UnsupervisedML\_Resources.pdf

## Unsupervised Machine Learning with Python

Useful resources

Section 1: Introduction

Course Github site:

<https://github.com/satishchandrareddy/UnsupervisedML>

Wikipedia page for Machine Learning

[https://en.wikipedia.org/wiki/Machine\\_learning](https://en.wikipedia.org/wiki/Machine_learning)

Wikipedia page for Unsupervised Learning:

[https://en.wikipedia.org/wiki/Unsupervised\\_learning](https://en.wikipedia.org/wiki/Unsupervised_learning)

# Set Up

- Instructor will use Windows 10 machine
  - Course material not specific to Windows – can use MacOS or Linux
- All code examples written in Python
- Course will run programs using
  - Jupyter Notebook: demos
  - Command Window: main course codes
- Should have text editor compatible with Python for writing program:
  - Examples: Atom, Sublime, Notepad++, etc

# Packages and Versions used by Instructor

Component	Version	Description/Comments
Python	3.8.3	
NumPy	1.18.5	Package for scientific computing. In this course, the numpy array is the principal container used to hold data
Matplotlib	3.2.2	Package for plotting and animation
pandas	1.0.5	Package containing data structures and data analysis tools. We will use its functionality to load data from csv file and to plot
scikit-learn	0.23.1	Package for supervised and unsupervised learning. We will use its functionality for generating datasets and processing text
IPython	7.16.1	Package of tools for interactive use of Python. We will use its functionality for animation in Jupyter notebooks.
wordcloud	1.8.1	Package for generating word clouds (useful for viewing document clustering results)
copy, pathlib, time		These packages are part of the python release

# Anaconda Platform

If you don't have Python on your machine, suggest installing Anaconda

- Anaconda Platform is distribution of Python for scientific computing
- We will use Anaconda prompt window and Jupyter notebooks to run programs
- Anaconda installation comes with packages: numpy, matplotlib, pandas, sklearn, IPython (we only need to install wordcloud)
- You are free to use any python platform, as long as you can install required packages, run python in the command window, and use Jupyter notebooks.



# Package Installation

- Install wordcloud manually
  - Open Anaconda prompt window
  - Make sure you are in base environment
  - Issue command: `conda install wordcloud`
- For simplicity, I am using the base environment
  - I have not created a separate conda environment on my machine for this course. You are free to create an environment, if you like

# 1.3 DEMO: Course Resources and Set Up

1) Download and unzip resources from Course Github Site

<https://github.com/satishchandrareddy/UnsupervisedML/>

2) Installation of Anaconda

<https://www.anaconda.com>

3) Installation of wordcloud:

- In Anaconda Prompt Window (base environment) issue command: `conda install wordcloud`

4) Test in Anaconda Prompt Window

- Open Anaconda Prompt window
- Change to directory `UnsupervisedML/Code/Programs`
- Issue command: `python testcommandwindow.py`

5) Test in Jupyter Notebook

- Open Jupyter Notebook
- Change to directory `UnsupervisedML/Examples/Section01`
- Open `JupyterNotebookTest.ipynb`