MIS4311
Machine Learning Applications

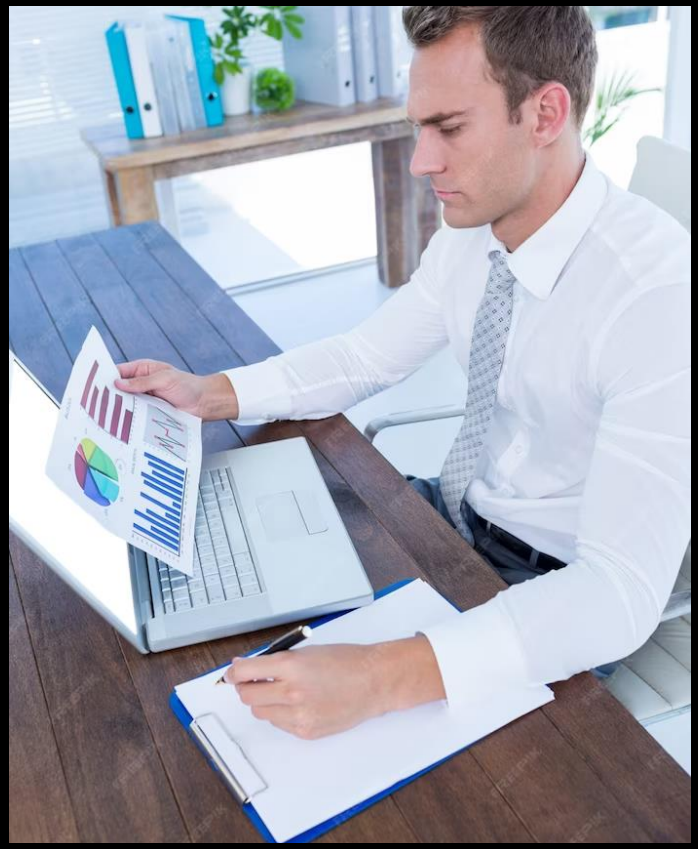
Fall 2025

Lecture #2

Introduction

In this class, we will cover the foundations and applications of machine learning.

We will explore **supervised**, **unsupervised**, and **reinforcement** learning, as well as common **algorithms** used in machine learning.



What is Machine Learning?

Machine learning is a type of artificial intelligence that allows computers to learn from data without being explicitly programmed. In other words, it is a way for computers to learn and improve from experience.
Machine learning is used in a variety of applications, from **image recognition** to **natural language processing**. By using machine learning, we can automate tasks and make predictions based on data.

Supervised Learning

Supervised learning is a type of machine learning where the input and output data is already labeled.

The goal is to learn a mapping function from the input to the output data. This is commonly used in **classification** and **regression** problems.

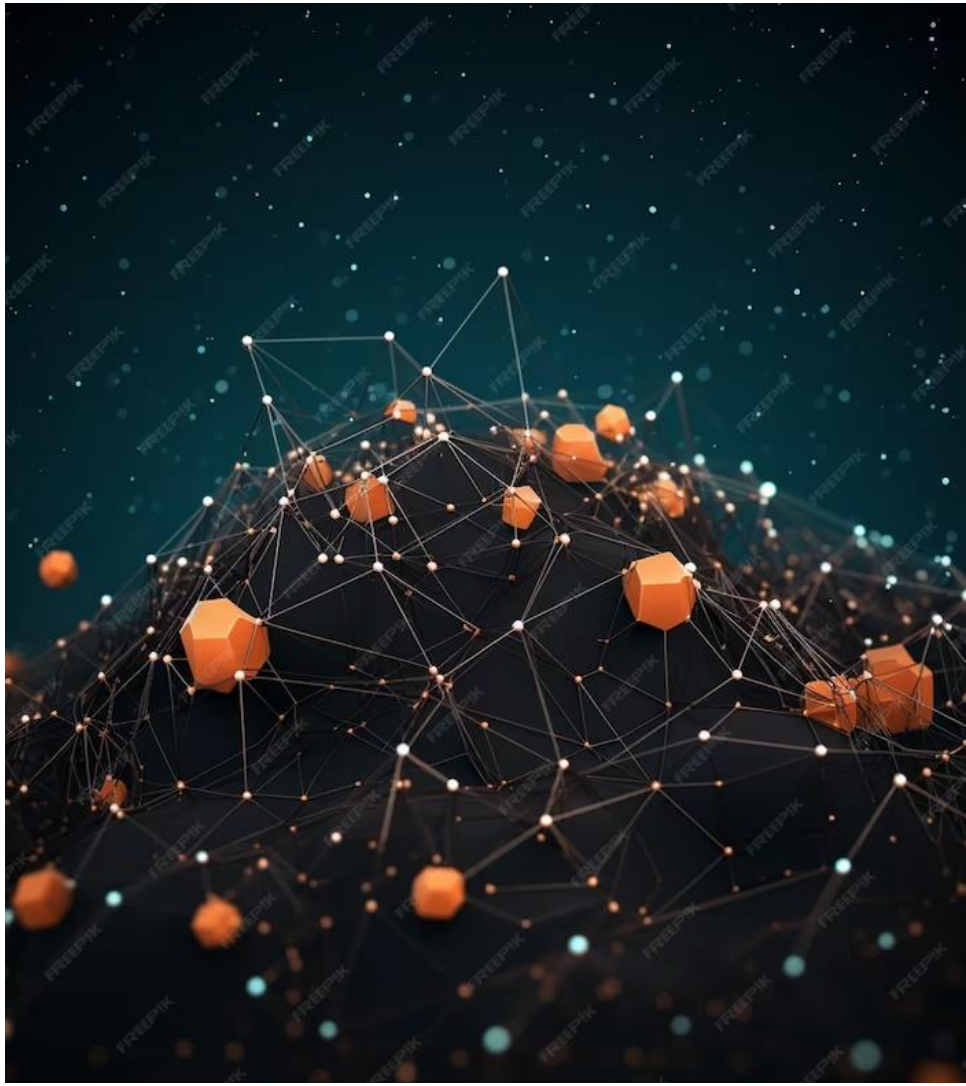
Some popular algorithms for supervised learning include **decision trees** and **random forests**.

Unsupervised Learning

Unsupervised learning is a type of machine learning where the input data is not labeled.

The goal is to find patterns and structure in the data. This is commonly used in **clustering** and **dimensionality reduction** problems.

Some popular algorithms for unsupervised learning include **k-means** and **principal component analysis**.



Common Machine Learning Algorithms

There are many algorithms used in machine learning, but some of the most common include **linear regression**, **logistic regression**, **support vector machines**, **neural networks**, and **decision trees**.

Each algorithm has its own strengths and weaknesses, and choosing the right one depends on the problem you are trying to solve.

Applications of Machine Learning

Machine learning is used in a variety of applications, from **recommendation systems** to **fraud detection**.

It is used in **healthcare** to predict diseases and in **finance** to predict stock prices.

Machine learning is also used in **autonomous vehicles** to navigate roads and in **speech recognition** to understand human language.

Reinforcement Learning

Reinforcement learning is a type of machine learning where an agent learns to interact with an environment by performing actions and receiving rewards or punishments.
The goal is to learn a policy that maximizes the cumulative reward over time.
This is commonly used in **game playing** and **robotics**. Some popular algorithms for reinforcement learning include **Q-learning** and **deep reinforcement learning**.

Types of Machine Learning

Supervised machine learning algorithms:

***Classification**

***Regression**

This is **the most commonly used** machine learning algorithm. It is called supervised because the process of algorithm learning from the **training dataset** can be thought of **as a teacher supervising** the learning process.

Suppose we have **input** variables **x** and an **output** variable **y** and we applied an algorithm to learn **the mapping function** from the input to output such as:

$$y = f(x)$$

Supervised machine learning algorithms

*** Classification:**
A problem is called classification problem when we have the **categorized output** such as “black”, “teaching”, “non-teaching”, etc.

*** Regression**:
A problem is called regression problem when we have **the real value output** such as “distance”, “kilogram”, etc.

Decision tree, random forest, KNN, logistic regression are the examples of supervised machine learning algorithms.

Unsupervised machine learning algorithms

In unsupervised learning there will be no correct answer and no teacher for the guidance. Algorithms help to discover interesting patterns in data.

- **Clustering:** In clustering problems, we need to discover the inherent (distinctive) groupings in the data.

  For example, grouping customers by their purchasing behavior.

- **Association:** A problem is called association problem, if it requires discovering the rules that describe large portions of our data.

  For example, finding the customers who buy both x and y.

Reinforcement machine learning algorithms

These kinds of machine learning algorithms are used very less.

These algorithms train the systems to make specific decisions.

Basically, the machine is exposed to an environment where it trains itself continually using the trial and error method.

These algorithms learn from past experience and tries to capture the best possible knowledge to make accurate decisions.

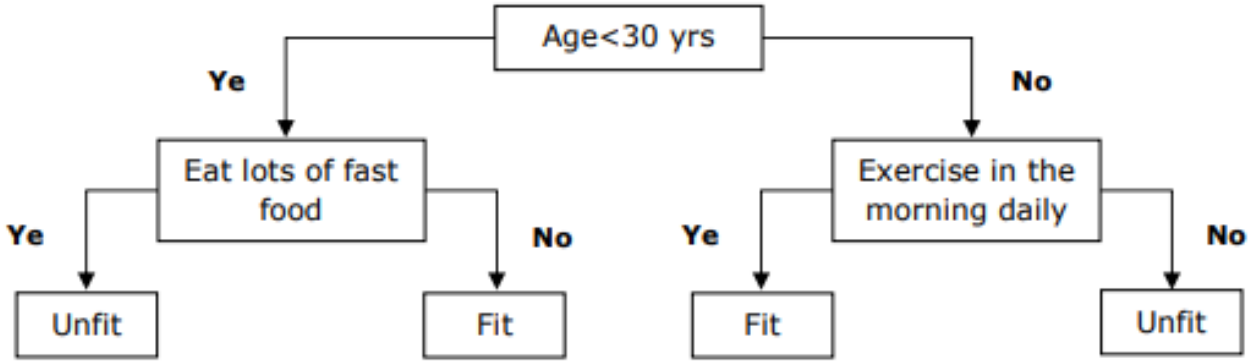
Markov Decision Process is an example of reinforcement machine learning algorithms.

Most Common Machine Learning Algorithms

Linear Regression: It is one of the most well-known algorithms in statistics and machine learning.

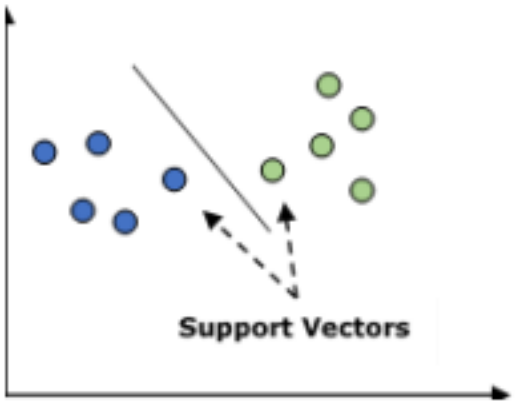
Logistic Regression: It is a classification algorithm and also known as **logit** regression. Basically, it predicts the probability hence its output lies in **between 0 and 1**.

Decision Tree: Basically it is a classifier expressed as recursive partition **based on the independent variables**. Decision tree has nodes which form the rooted tree.



Most Common Machine Learning Algorithms

Support Vector Machine (SVM): It is mainly used for classification problems. The main concept of SVM is to plot **each data item as a point in n-dimensional space** with the value of each feature being the value of a particular coordinate. Here **n** would be the features we would have.



Naïve Bayes: The logic behind this classification technique is to use Bayes theorem for building classifiers. The assumption is that **the predictors are independent**. In simple words, it assumes that the **presence of a particular feature in a class is unrelated to the presence of any other feature**. The Naïve Bayes model is easy to build and particularly useful **for large data sets**.

Most Common Machine Learning Algorithms

K-Nearest Neighbors (KNN):
It is widely used to solve classification problems. The main concept of this algorithm is that **it used to store all the available cases** and classifies new cases by majority votes of its k-neighbors.

K-Means Clustering:
The main logic of K-Means clustering algorithm (unsupervised) is to classify the data set through a number of clusters.

Random Forest:
It is a supervised classification algorithm. Basically it is **the collection of decision trees** (i.e., forest) or ensemble (union) of the decision trees. The basic concept of random forest is that each tree gives a classification and the forest **chooses the best classifications from them.**

Introduction to AI with Python concept

Download and install **Anaconda** from the link:

https://www.anaconda.com/products/distribution

Why Python?

1) Open source

2) Syntax as simple as English

3) a large, active, collaborative community

4) Extensive packages for almost any task

python

```
print('Hello, World!')
```

java

```
public class Hello{
        public static void main(String
argv[]){
        system.out.println('Hello,
World!');
        }
}
```

C language

```
#include <stdio.h>
int main(){
        printf('Hello, World!');
        return 0;
}
```

Python IDE

Integrated Development Environment (Text Editor)

PyCharm

Eclipse

Spyder

Variables and Data Types

Variable names must begin with either a letter or an underscore.

Note: they can contain numbers, they must not start with one

strings

integers

floating points

booleans

dynamic typing

```
information1 = None
number = 0  # number is an integer variable
roll_width = 1.4 /* roll_width is a floating point variable */
price_per_metre = 5
filename = 'data.txt'
trace =False
sentence = 'this is a whole lot of nothing'
total_price = roll_width * price_per_metre
```

Variables and Data Types

Once Python has decided what type a variable is, it will flag up a **TypeError** if you try to perform an inappropriate operation on that data.

```
'''
Problem: Try to add different data types.
User must be able to initialize and assign
variables.
'''

b = 3
c = 'word'
trace = False
d = b + c
Traceback (most recent call last):
File '<stdin>', line 1, in <module>
TypeError: unsupported operand type(s) for +:
'int' and 'str'
```

Get value and Display it

Get a value from the user and display it.

\n → newline

\t → Tab (space)

\v → Vertical tab

```
'''
Problem: Get a value from the user and
display it.
Target User: Noushin Hajarolasvadi
Interface: Pycharm
‘’’
name=‘Mehmet’
name = input('Enter your name: ')
print('Your name is:', name)
print('and it has\t', len(name), '\t
charachters')
```

Arithmetic in Python

Addition	**+**	**35 + 22**	**57**
		'Py' + 'thon'	**'Python'**
Subtraction	**-**	**35 - 22**	**13**
Multiplication	*****	**3 * 2**	**6**
		'Py' * 2	**'PyPy'**
Division	**/**	**3.0 / 2**	**1.5**
		3 / 2	**1**
Exponentiation	******	**2 ** 0.5**	**1.41421356…**

Operators

Symbol	Operation
**	Power (exponent)
%	modulus
//	integer division
/	division
*	multiplication
-	subtraction
+	addition

```
a = 9
b = 10.5

print('sum is:', a + b)
print('difference is:', a - b)
print('multiplication is:', a * b)
print('division is:', a / b)
print('power is:', a ** b)
```

Operators

Logical operators:

(expression1) **and** (expression2)

(expression1) **or** (expression2)

not (expression1)

```
a = 2
b = 8
c = 4
d = 6
```

false true

```
print((a >= b) and (c < d))
```

false true

```
print((a >= b) or (c < d))
```

false

```
print(not a)
```

Your First Python Program

- Python is “case-sensitive”
  - print “Game Over”
  - Print“Game Over”
  - PRINT “Game Over”

Program Documentation

- Comment lines provide documentation about your program
  - Anything after the "#" symbol is a comment
  - Ignored by the computer

```
# I am a Programmer
# First Python Program
# March 1, 2021
''' some comments '''
```

Variables

```
x = 4                   # x is of type int
y = ”mehmet”            # y is of type str


x = 3                   # x is of type int
x = ”gizem"             # x is now of type str

x = str(6)      # x will be ‘6'
y = int(4)      # y will be 4
z = float(2)    # z will be 2.0
```

Use ”type()” function to find type of any variable

Assignment operators

=	x = 5	x = 5
+=	x += 3	x = x + 3
-=	x -= 3	x = x - 3
*=	x *= 3	x = x * 3
/=	x /= 3	x = x / 3
%=	x %= 3	x = x % 3
//=	x //= 3	x = x // 3
**=	x **= 3	x = x ** 3

Variables

```
z = 4                    # Python is case-sensitive
Z = ”mehmet”             # Z will not overwrite z

x, y, z = "Orange", "Banana", "Cherry"
x = y = z = "Orange"

fruits = ["apple", "banana", "cherry"]
x, y, z = fruits         #unpacking a list
```

Python Casting

```
x = float(1)          # x will be 1.0
y = float(2.8)        # y will be 2.8
z = float("3")        # z will be 3.0
w = float("4.2")      # w will be 4.2
```

```
x = str("s1")     # x will be 's1'
y = str(2)        # y will be '2'
z = str(3.0)      # z will be '3.0'
```

String Methods

capitalize() Converts the first character to upper case

lower() Converts a string into lower case

upper() Converts a string into upper case

title() Converts the first character of each word to upper case

```
#named indexes:
txt1 = "My name is {fname}, I'm {age}".format(fname = "John", age = 36)
#numbered indexes:
txt2 = "My name is {0}, I'm {1}".format("John",36)
#empty placeholders:
txt3 = "My name is {}, I'm {}".format("John",36)
```

String Methods

name=”ahmet”

0 1 2 3 4

print(name[1]) → h

print (len (name)) → 5

```
>>> name= “Erkay Savas"
>>> print (name[6:11])      → Savas
>>> print (name[6:10])      → Sava
>>> print (name[1:10:2])    → ra aa
>>> print (name[9:3:-1])    → avaS y
>>> print (name[:])         → Erkay Savas
>>> print (name[::-1])      → savaS yakrE
```

Next Week

- Python application tools

Thank you for your participation ☺