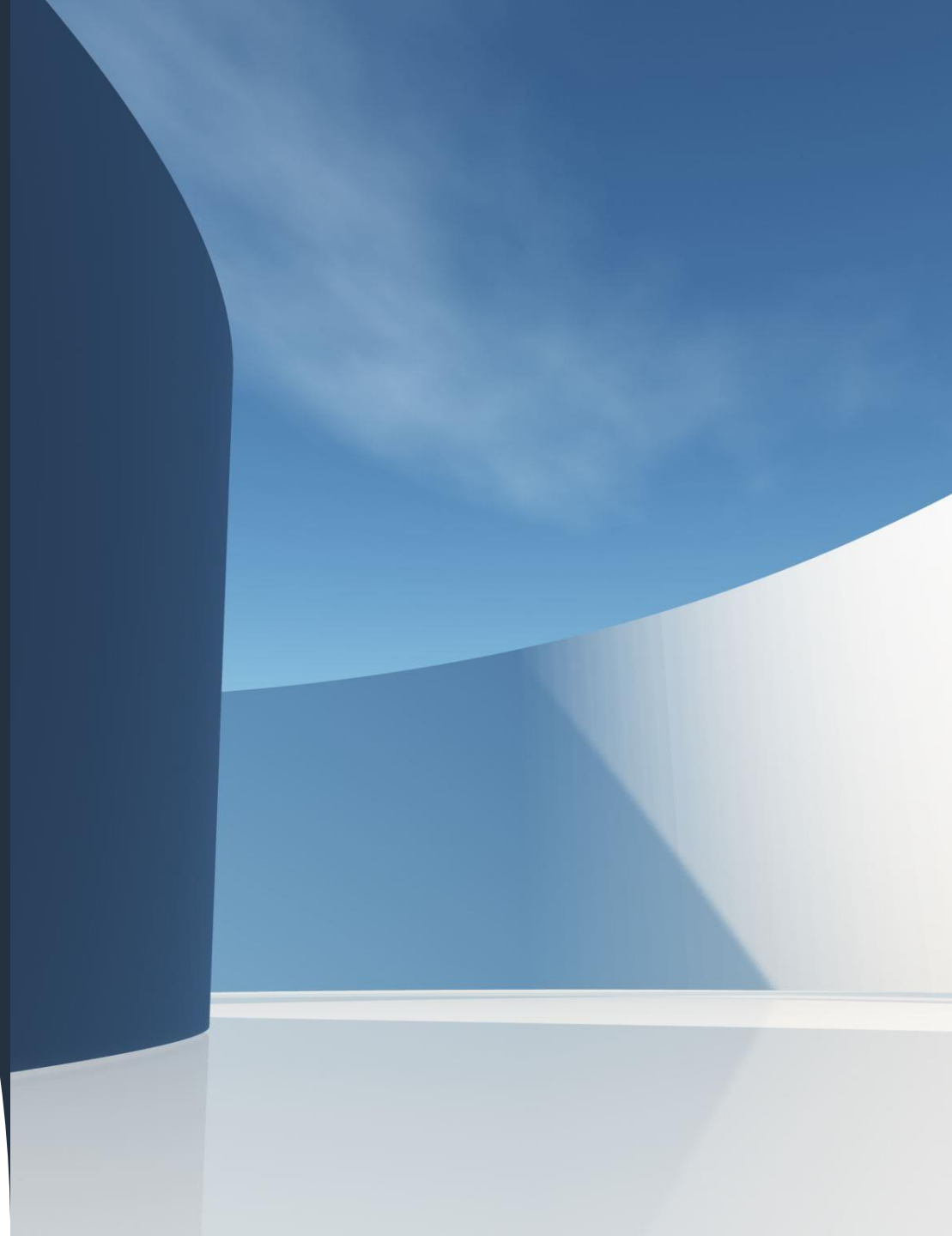
Artificial Intelligence with Python

Module 3 - AI



ARTIFICIAL INTELLIGENCE

Nowadays, Artificial Intelligence ("AI") is one of the buzz-words that is being talked about in news, social media, entertainment and elsewhere. It claims to revolutionize the world by making machines intelligent. At the very least, it is all around us in our everyday lives.

But what is AI really?

- "Artificial" means "man-made" and "intelligence" means ability to think, reason,...
- One definition of AI would be "...**a branch of computer science dedicated to creating intelligent machines that work and react like humans".** But we will soon see that it is much more than computer science. As a matter fact, AI uses ideas and techniques from
- mathematics
- statistics
- social sciences
- biology

Before moving on let's take a look at some of the key characteristics of human intelligence. Humans are able to

- use available information to make decisions
- communicate with other people
- identify patterns (in data)
- remember what people have said or done
- adapt to new situations

AI attempts to mimick these human behaviour (and more) to solve complex problems. In doing so it

- is more accurate than human
- faster than human
- does not get tired
- does not let emotions influence the outcome (this may or may not be a good thing)
- outputs consistent results and decisions

But AI has also disadvantages

- can't think outside of the box
- no emotions
- increases (our) dependency on machines
- not very original (creativity, arts, music, paintings,...)

- Large portion of AI is machine learning where the system is trained with data and it can make predictions based on this data. Deep Learning is a subset of machine learning and it uses neural networks to solve very complicated problems in pattern recognition etc.

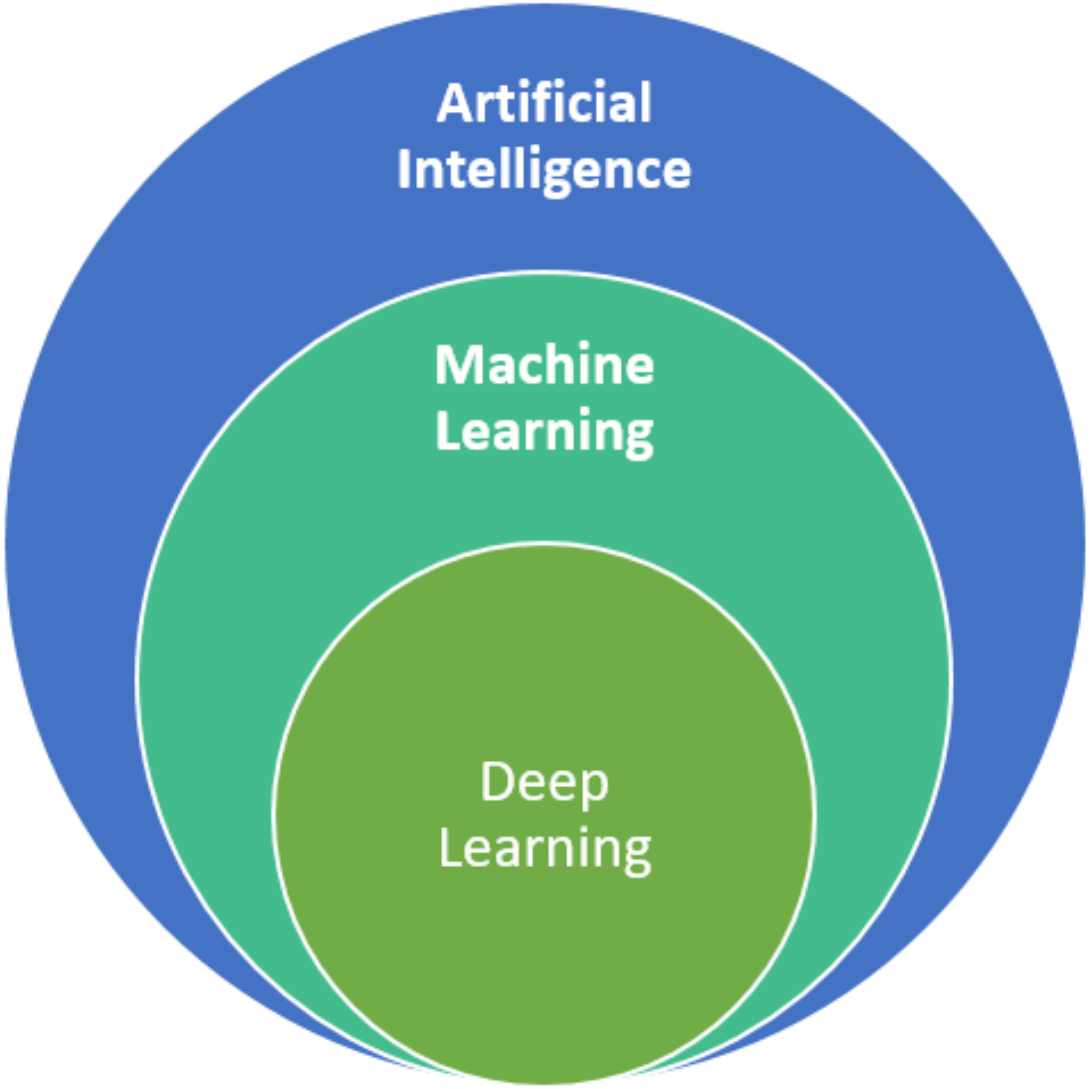


Figure 1: artificial intelligence, machine leaning and deep learning Source: Nadia BERCHANE (M2 IESCI, 2018)

AI can be divided into categories based on capabilities that AI can perform. These are

1. Narrow AI (or weak AI)
    1. able to perform dedicated tasks with intelligence.
    2. cannot perform outside of its specific domain
    3. will probably fail if instructed to work on other tasks
    4. currently the most widely available form of AI.
    5. Examples: self-driving cars, speech recognition,...

2. General AI (or strong AI)

1. able to perform any intellectual task with efficiency comparable to humans.
2. able to think on its own (like humans)
3. does not exist currently (as far as we know) but is under heavy research and development

3. Super AI

1. able to exceed to human intelligence
2. would be able to perform any task better than any human
3. still a hypothetical concept

Another classification of AI is based on functionality:

1. Reactive machines

1. machines that are purely reactive
2. cannot form memory or use past experience
3. only look at current situation and react only on it
4. cannot infer from data to evaluate future actions
5. we program them and they run the program
6. Examples: coffee maker, washing machine,

Another classification of AI is based on functionality:

2. Limited memory (current)
    1. use past experience and present data to make decisions
    2. does not evolve new ideas
    3. reprograms itself continually based on training
    4. short-term memory
    5. Examples: autonomous vehicles (they remember recent speed/distance of nearby cars, speed limit,... for navigation).

Another classification of AI is based on functionality:

3. Theory of mind

1. can socialize and understand human emotions, beliefs, thoughts,... and how they affect decision making
2. able to adjust their behaviour
3. yet to be built (although debatable), but under intense research

Another classification of AI is based on functionality:

4. Self-awareness

1. future of AI
2. super intelligent machines
3. conscious machines with their own sentiments and self-awareness, knowledge of internal state
4. able to react like human beings and predict people's feelings and act accordingly
5. does not exist yet, hypothetical

Applications of AI

1. Healthcare; AI can help doctors to diagnose diseases and give advice to patients on proper treatments
2. Gaming
3. Finance; automation, algorithm trading, banks' loan decisions,...
4. Transportation; route planning, autonomous buses, autopilots in planes,...
5. Social media, news
6. Automotive industry; self-driving cars, driver assists, computer vision to detect road signs
7. Robotics; industrial robots, humanoid robots, drones,...
8. Entertainment; Netflix, Spotify etc. suggest titles based on user's past choices
9. Education; automated grading, chatbots as teacher's assistants, virtual tutors in the future??

More concrete examples from real-life:

- personal assistants
- search engine results refining (Google photos tagging, query completion)
- spam detection
- text translation
- chatbots
- e-commerce recommendations

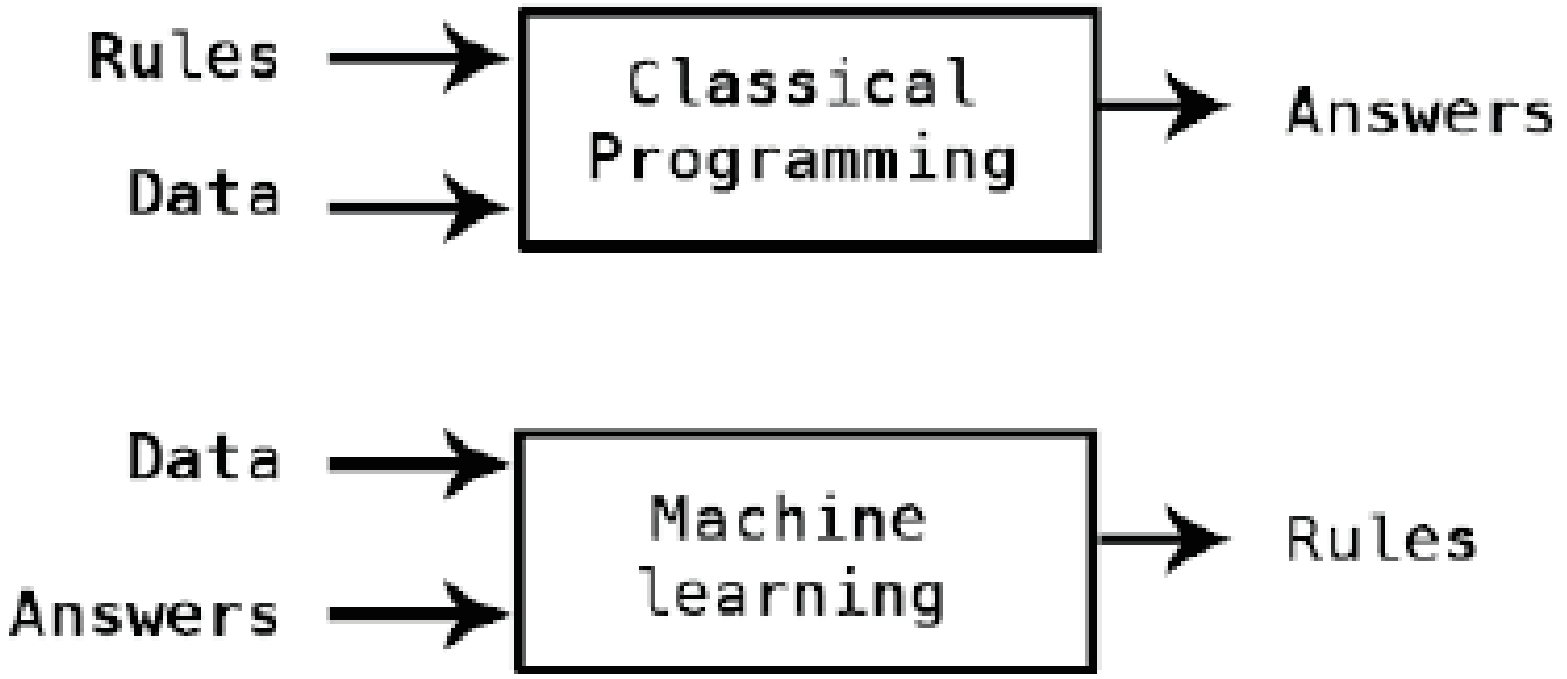
Machine learning basic concepts

Machine learning is a subset of AI in which computers learn from past data more or less automatically. Machine learning techniques are based on mathematical models which enable the computer make predictions based on historical data and some other problem-specific information.

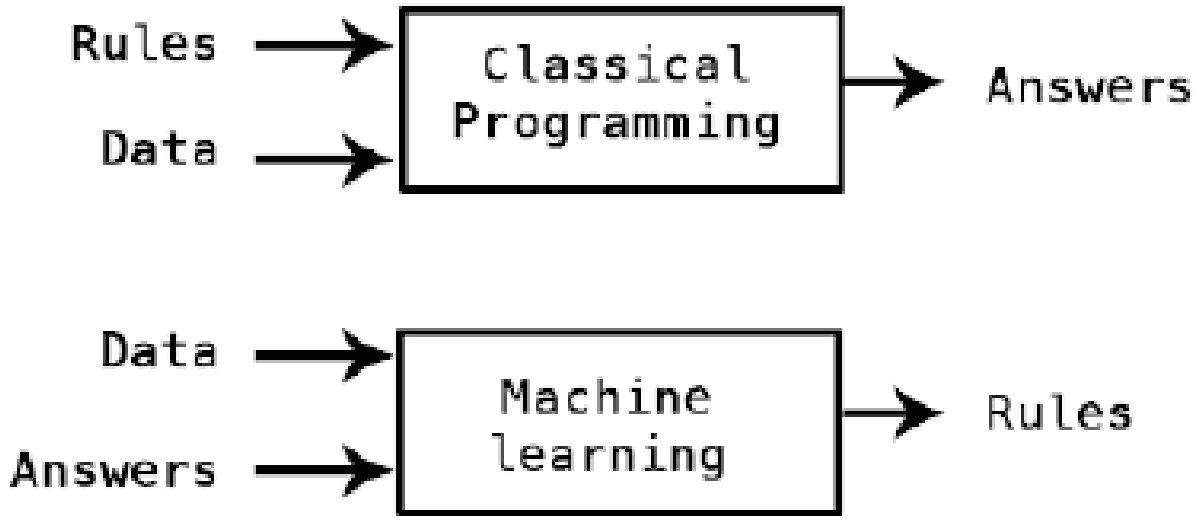
- Machine learning can be used in
- speech recognition
- image recognition
- text recognition
- email filtering
- recommendation systems

- Consider a baby who is trying out new tastes by experimentation. She will soon learn that play-doh tastes bad and warm milk tastes good. In a similar fashion, a machine must combine past information to learn from this information and be able to behave differently in the future.

Machine learning makes heavy use of computer science. The difference between machine learning and traditional programming is illustrated below.



In classical programming paradigm the programmer writes a series of rules (if-statements, loops,.) and provides data to the program. The program provides answers (results) based on these rules and data. In machine learning the situation is partly reversed. The computer program is provided with data and answers and its task is to create rules. These rules can be applied to new data to make predictions.



A machine learning system is said to be trained rather than programmed explicitly. Training means that computer is given a set of examples relevant to task at hand and it is to find statistical patterns in these examples in order to output the rules.

In general, machine learning performs better when more data is fed to the system. Luckily the amount of data has risen dramatically during the last 10-20 years due to widespread use of Internet to the point where we now talk about Big Data. At the same time computing performance and amount of memory has also increased steadily. These are the driving forces of current success of AI and machine learning.

Machine learning types

Machine learning is typically divided into three types (or classes).

1. supervised learning
    1. "learning with a teacher"
    2. system is provided with labeled data (both input and output given as training data)
    3. system is able to predict future outcomes based on past data
    4. Techniques: Regression, Classification
    5. Example: Image classification (training data is a set of images with labels associated with them)

1. unsupervised learning
    1. "learning on your own"
    2. system is able to identify hidden patterns (or trends) from the input data (no output data or labels given)
    3. data is organized according to patterns with similarities and anomalies becoming more evident
    4. Techniques: Clustering, Association, Anomaly Detection
    5. Example: buyer grouping based on purchase history and browsing history

•

1. reinforcement learning
    1. no training data given
    2. agent interacts with the environment
    3. learns based on feedback (either reward or punishment) received for its last action
    4. tries to maximize total reward
    5. uses neural networks
    6. Example: game AI (chess, pong, etc.)

Machine learning workflow

- A full development of machine learning system is divided into smaller steps for simplicity. Strictly speaking, the first step is to formulate the problem as accurately as possible. In addition, it is crucial to come up with any additional information that can help better understand the problem and eventually solve it too.

- The basic steps in machine learning are

1. data gathering
2. data preparation
3. data analysis (descriptive statistics)
4. model training
5. model testing
6. deployment for making predictions

- It is generally considered that steps 1-3 are most laborous in terms of human work. Collecting the data and putting it into correct form and making sure it is valid often takes the most time in a ML workflow.
- In training phase the model's parameters are tuned to optimal values. After training it is crucial to test the model with data it has never seen before. In testing the model is used to predict outcomes based on input and the model's predictions are compared to known outputs (if available). During testing also metrics are computed to quantify the performance of the model.

Datasets for machine learning

- Since machine learning relies heavily on data it is crucial to have access to it. But first, let us discuss what we mean here by a dataset.
- A dataset is a collection of data that is organized in some structural format. Most often the data is in tabular format such as below. This data comes from iris.csv file

In the context of Python we often use the module 'pandas' which treats datasets as data frames. A data frame is more general than an array since arrays (in numpy) can contain only one type of data, e.g. int64 integers. In data frames columns have names (see above) and rows have numbers. Moreover, different columns may contain different types of data (numeric, text, categorical etc.).
Let us illustrate the use of pandas in data analysis. As always, one needs to import the module first. It is standard convention to give it an abbreviation pd:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa

import pandas as pd

Next, we read some data from a CSV file into a Pandas dataframe

```
df = pd.read_csv("iris.csv")
```

The functions head(), tail() and describe() are useful functions to get a glimpse of the data

```
print(df.head())
print(df.tail())
print(df.describe())
```

The data types, indices, column names and actual data values can be inspected using

```
print(df.dtypes)
print(df.index)
print(df.columns)
print(df.values)
```

The data can be sorted by a column using the column name

```
df2 = df.sort_values('sepal_width',ascending=False) # does not sort in-place
print(df2)
```

Pandas provides powerful methods to slice data frames. We start with slicing columns by name

```
print(df[['sepal_width']]) # slice one column by name
print(df[['sepal_width','sepal_length']]) # slice two columns by name
```

Slicing by rows is done as follows:

```
print(df[2:4]) # slice rows by index, exclusive
```

Slicing by rows and columns at the same time uses the functions loc() or iloc(). For example,

```
print(df.loc[2:4,['petal_width','petal_length']]) # slice rows by index and columns by name
print(df.iloc[2:4,[0,1]]) # slice row and columns by index
```

Slicing can be combined with assignment operator to assign new values to existing data records.

Filtering of the data can be done using logical conditions or isin() function. For example,

```
print(df[df.sepal_width>3]) # slicing with logical condition
print(df[df['species'].isin(["Iris-setosa"])])
```

New columns can be added to a data frame by giving it a name and using assignment operator as

```
df['sepal_area'] = df.sepal_length*df.sepal_width
print(df)
df['zeros'] = 0.0
print(df)
```

To remove columns you may use the drop() function with axis parameter set to 1:

```
df = df.drop(['zeros'],axis=1)
print("df after drop",df)
```

For renaming columns there are two possibilities. Either rename one column (which can be done in-place) or give all columns new names

```
df.rename(columns = {'sepal_area':'sep_ar'},inplace=True)
print(df.head())
df.columns = ['col1','col2','col3','col4','col5','col6']
print(df.head())
```

To add a row to a data frame takes an intermediate step with the function Series(). For example,

```
to_append = [7.0,4.0,5.5,6.6,"Iris-setosa",28.0]
a_series = pd.Series(to_append, index = df.columns)
df = df. _ append(a_series, ignore_index=True)
print(df) OR
to_append = [7.0,9.0,9.0,7.0,"Hybrid"]
df.loc[len(df)] = to_append
print(df)
```

Looping of the data frame can be done in a single for-loop as

```
for ind, row in df.iterrows():
print(ind,row['col2'])
```

Note that here row represents an entire row at index 'ind' and usual slicing can be done per row.

Finally, data frames are easy to save to CSV files using

```
df.to_csv("iris_new.csv")
```

THANKS