University of Sulaimani
College of Science
Computer Department
4th Stage

Data Science Management

Introduction to Data Science

Class 1

Theoretical and practical lectures

Assist. Prof. Dr. Miran Taha Abdullah
2025-2026

The agenda for this lecture includes:

What is Data Science?

Why Data Science?

What Does a Data Scientist Do?

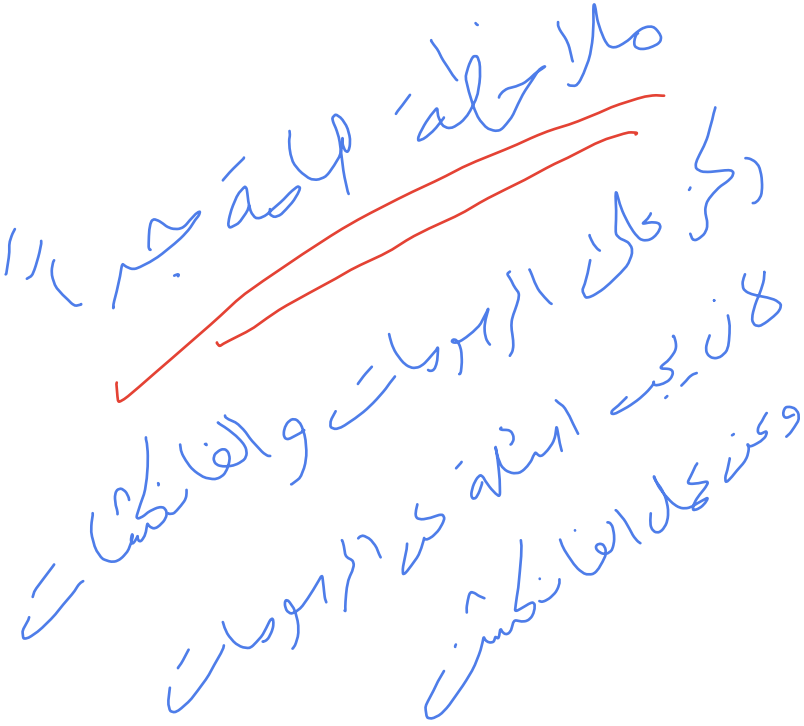
Why Become a Data Scientist

Skills Required to Become a Data Scientist

Data Scientist vs Data Analyst

Average Salary of a Data Scientist

Data Scientist Roadmap



Introduction to Data Science

- Data science is the field of study that uses:

scientific methods, Processes, algorithms, and systems

to **extract knowledge** and **insights** from **structured** and **unstructured** data.

The scientific method is the process of *objectively* establishing facts by testing and experimenting. It's a systematic way to ensure that findings are based on evidence rather than assumptions or bias

Processes : a series of actions or steps taken in order to achieve a particular end.

Algorithm : step-by-step set of instructions used to solve a specific problem or perform a task

A **system** is an organized collection of components or elements that interact and function together to achieve a common objective

- **Data** is a collection of information.
- One purpose of Data Science is to structure data, making it interpretable and easy to work with.
- Data can be categorized into two groups:

- **Unstructured Data**
- Unstructured data is not organized. We must organize the data for analysis purposes.

Structured Data

Structured data is organized and easier to work with.

Unstructured data

Structured data

Introduction to Data Science

- Data Science combines various disciplines such as **statistics**, **machine learning**, **data analysis**, and **visualization** to **uncover hidden patterns**, **trends**, and **correlations in data**.

- Data science plays a crucial role in **decision-making**, **forecasting**, and **problem-solving** across industries, driving innovation and enabling organizations to make data-driven decisions

What is Data Science?

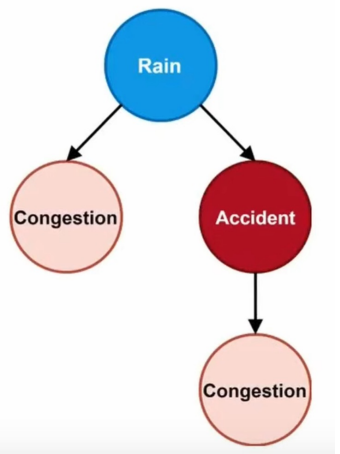
- Data Science is about **data gathering**, **analysis** and **decision-making**.
- Data Science is **about finding patterns in data**, through **analysis**, and make future **predictions**.

By applying Data Science, companies can achieve: ← اتى في الأمتحان

- Better decisions (should we choose A or B)
- Predictive analysis (what will happen next?)
- Pattern discoveries (find pattern, or maybe hidden information in the data)

Better decisions

Predictive analysis



Pattern discoveries

Where is Data Science Needed?

Examples of where Data Science is needed:

- For route planning: To discover the best routes to ship.
- To foresee delays for flight/ship/train etc. (through predictive analysis).
- To create promotional offers.
- To find the best suited time to deliver goods.
- To forecast the next years revenue for a company.
- To analyze health benefit of training.
- To predict who will win elections.

Data science's lifecycle

1- Capture: This stage involves gathering raw structured and unstructured data.

i.e. Data Acquisition, Data Entry, Signal Reception, Data Extraction.

2- Maintain: This stage covers taking the raw data and putting it in a form that can be used.

i.e. Data Warehousing, Data Cleansing, Data Staging, Data Processing, Data Architecture.

3- Process: Data scientists take the prepared data and examine its patterns, ranges, and biases to determine how useful it will be in predictive analysis.

i.e. Data Mining, Clustering/Classification, Data Modelling, Data Summarization.

Data science's lifecycle (cont.)

4- Analyse: This stage involves performing the various analyses on the data.

i.e. Exploratory/Confirmatory, Predictive Analysis, Regression, Text Mining, Qualitative Analysis.

5- Communicate: In this final step, analysts prepare the analyses in easily readable forms such as charts, graphs, and reports.

i.e. Data Reporting, Data Visualization, Business Intelligence, Decision Making.

How Does a Data Scientist Operate?

1. **Ask the right questions** - To understand the business problem.
2. **Explore and collect data** - From database, web logs, customer feedback, etc.
3. **Extract the data** - Transform the data to a standardized format.
4. **Clean the data** - Remove erroneous values from the data.
5. **Find and replace missing values** - Check for missing values and replace them with a suitable value (e.g. an average value).
6. **Normalize data** - Scale the values in a practical range (e.g. 140 cm is smaller than 1,8 m. However, the number 140 is larger than 1,8. - so scaling is important).
7. **Analyze data, find patterns and make future predictions**.
8. **Represent the result** - Present the result with useful insights in a way the "company" can understand.

Types of Data Science

1. Data Scientist
2. Data Analyst
3. Machine learning expert
4. Data engineer
5. Data Architect
6. Data Administrator
7. Business Analyst
8. Business Intelligence Manager

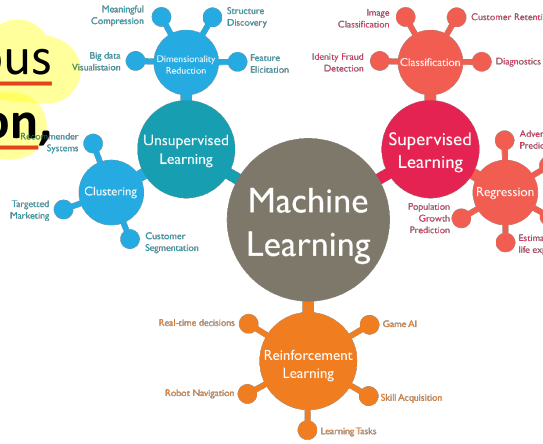
Types of Data Science (cont.)

1. Data Analyst:

Data analyst is an individual, who performs mining of huge amount of data, models the data, looks for patterns, relationship, trends, and so on. They come up with visualization and reporting for analyzing the data for decision making and problem-solving process.

2. Machine Learning Expert:

The machine learning expert is the one who works with various machine learning algorithms used in data science such as **regression, clustering, classification, decision tree, random forest**, etc.



3. Data Engineer:

A data engineer works with massive amount of data and responsible for building and maintaining the data architecture of a data science project. Data engineer also works for the creation of data set processes used in modelling, mining, acquisition, and verification.

Data Engineers
They own the data realms.
They build systems.
They make the data usable.

4. Data Scientist:

A data scientist is a professional who works with an enormous amount of data to come up with compelling business insights through the deployment of various tools, techniques, methodologies, algorithms, etc.

Data Scientists
They make insights out of data and build models that allow you to automate business processes or decisions.

Prerequisite for Data Science (How Does a Data Scientist Work?)

Non-Technical Prerequisite

Curiosity: To learn data science, one must have curiosities. When you have curiosity and ask various questions, then you can understand the business problem easily.

Critical Thinking: It is also required for a data scientist so that you can find multiple new ways to solve the problem with efficiency.

Communication skills: Communication skills are most important for a data scientist because after solving a business problem, you need to communicate it with the team

Technical Prerequisite

- **Machine learning:** To understand data science, one needs to understand the concept of machine learning. Data science uses machine learning algorithms to solve various problems.
- **Mathematical modelling:** Mathematical modelling is required to make fast mathematical calculations and predictions from the available data.
- **Statistics:** Basic understanding of statistics is required, such as mean, median, or standard deviation. It is needed to extract knowledge and obtain better results from the data.
- **Computer programming:** For data science, knowledge of at least one programming language is required. R, Python, Spark are some required computer programming languages for data science.
- **Databases:** The depth understanding of Databases such as SQL, is essential for data science to get the data and to work with data.

Why python?

- High-level interpreted language
- Very clear syntax => easy-to-read
- Open source
- Platform independent
- Large number of libraries
- Binding to all standard GUI kits (TkInter, Qt, ..)
- Powerful automated testing tools
- Easily integrated with C/C++, and Fortran
- Actively used and extended by scientists

Python Language

- Clear syntax -> easy to understand
- Indentation for blocks
- Use of namespace, exception handling
- Automatic doc.: doc strings & pydoc
- Dynamic typing
- Supports both procedural & OO approach
- Easy code testing
- Garbage collecting
- Functional programming, iterators,…

Operator	Description	Syntax
+	Addition: adds two operands	x + y
–	Subtraction: subtracts two operands	x – y
*	Multiplication: multiplies two operands	x * y
/	Division (float): divides the first operand by the second	x / y
//	Division (floor): divides the first operand by the second	x // y
%	Modulus: returns the remainder when first operand is divided by the second	x % y
**	Power : Returns first raised to power second	x ** y

Comparison Operators

Operator	Description	Syntax
>	Greater than: True if the left operand is greater than the right	x > y
<	Less than: True if the left operand is less than the right	x < y
==	Equal to: True if both operands are equal	x == y
!=	Not equal to – True if operands are not equal	x != y
>=	Greater than or equal to True if the left operand is greater than or equal to the right	x >= y
<=	Less than or equal to True if the left operand is less than or equal to the right	x <= y

Logical Operators

Logical operators perform **Logical AND**, **Logical OR**, and **Logical NOT** operations. It is used to **combine conditional statements**.

Operator	Description	Syntax
and	Logical AND: True if both the operands are true	x and y
or	Logical OR: True if either of the operands is true	x or y
not	Logical NOT: True if the operand is false	not x

Bitwise Operators

Bitwise operators act on **bits and perform the bit-by-bit** operations. These are used to operate on **binary numbers**.

Operator	Description	Syntax
&	Bitwise AND	x & y
|	Bitwise OR	x | y
~	Bitwise NOT	~x
^	Bitwise XOR	x ^ y
>>	Bitwise right shift	x>>
<<	Bitwise left shift	x<<

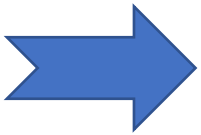
In Python how to convert a number to Binary

- Answer: it's simple method, use the bin() function to convert from a decimal value to its corresponding binary value.

Use bin() Function to Convert Int to Binary in Python
Use format Function to Convert Int to Binary in Python
Use the str.format() Method to Convert Int to Binary in Python

- >>> bin(1)→ 0b**1**
- >>>bin(2)→ 0b**10**
- >>>bin(3)→ 0b**11**
- >>>bin(4)→ 0b**100**

format(***no.***, 'b')



```
>>> format(1,'b')
'1'
>>> format(2,'b')
'10'
>>> format(3,'b')
'11'
>>> format(4,'b')
'100'
```

Format types convertion

Formatting Type	Role
=	Places the sign to the leftmost position
b	Converts the value into equivalent binary
o	Converts value to octal format
x	Converts value to Hex format
d	Converts the given value to decimal
E	Scientific format, with an E in Uppercase
X	Converts value to Hex format in upper case

```
# Examples of Bitwise operators
a = 10
b = 4

# Print bitwise AND operation
print(a & b)

# Print bitwise OR operation
print(a | b)

# Print bitwise NOT operation
print(~a)

# print bitwise XOR operation
print(a ^ b)

# print bitwise right shift operation
print(a >> 2)

# print bitwise left shift operation
print(a << 2)
```


Output

```
0
14
-11
14
2
40
```

```
>>> format(10,'b')
'1010'
>>> format(4,'b')
'100'
```

Assignment Operators

Assignment operators are used to assigning values to the variables.

Operator	Description	Syntax
=	Assign value of right side of expression to left side operand	x = y + z
+=	Add AND: Add right-side operand with left side operand and then assign to left operand	a+=b a=a+b
-=	Subtract AND: Subtract right operand from left operand and then assign to left operand	a-=b a=a-b
*=	Multiply AND: Multiply right operand with left operand and then assign to left operand	a*=b a=a*b
/=	Divide AND: Divide left operand with right operand and then assign to left operand	a/=b a=a/b
%=	Modulus AND: Takes modulus using left and right operands and assign the result to left operand	a%=b a=a%b
//=	Divide(floor) AND: Divide left operand with right operand and then assign the value(floor) to left operand	a//=b a=a//b
**=	Exponent AND: Calculate exponent(raise power) value using operands and assign value to left operand	a**=b a=a**b
&=	Performs Bitwise AND on operands and assign value to left operand	a&=b a=a&b
|=	Performs Bitwise OR on operands and assign value to left operand	a|=b a=a|b
^=	Performs Bitwise xOR on operands and assign value to left operand	a^=b a=a^b
>>=	Performs Bitwise right shift on operands and assign value to left operand	a>>=b a=a>>b
<<=	Performs Bitwise left shift on operands and assign value to left operand	a <<= b a= a << b

Identity Operators:

is and is not are the identity operators both are used to check if two values are located on the same part of the memory. Two variables that are equal do not imply that they are identical.

is True if the operands are identical
is not True if the operands are not identical

```
a = 10
b = 20
c = a

print(a is not b)
print(a is c)
```

Output
True
True

4. Working with Fractions

Fraction(numerator, denominator)

```
>>> from fractions import Fraction
>>> f = Fraction(3, 4)
>>> f
 Fraction(3,4)
```

```
>>> from fractions import *
>>> Fraction(0.75)
Fraction (3,4)
>>> Fraction ('3/4')
Fraction (3,4)
```

```
>>> Fraction(4) == Fraction(4,1)
True
```

5. Complex Numbers

- The numbers we've seen so far are the so-called real numbers. Python also supports complex numbers with the imaginary part identified by the letter j or J
- (as opposed to the letter i used in mathematical notation).

For example, the

- complex number 2 + 3i would be written in Python as 2 + 3j: >>> a = 2 + 3j
- >>> type(a)
- <class 'complex'> As you can see, when we use the type() function on a complex number, Python tells us that this is an object of type complex.

7. Handling Exceptions and Invalid Input

Three types of Errors

Complie time Error

Syntax Error
E.g.
Missing (:)
Wrong Spelling

Logical Error

Wrong output

2+2= 5X

Run time Error

Your code completely complied and no error.
6/0 = ?

```
A = 5
B = 2
Print(A/B)
#Statement : Normal -- Normal statement, it will not give an error
```

```
A = 5; B = 2 ; C = 0
Print (A/B)
Print(A/C)  #→ execution stop here, Critical Statement
```

Exception: description of the error

An exception is an **error** that happens during the execution of a program. Exceptions are known to non-programmers as instances that do not conform to a general rule. The name "exception" in computer science has this meaning as well: It implies that the problem (the exception) doesn't occur frequently, i.e. the exception is the "exception to the rule".

Exception handling is a construct in some programming languages to handle or deal with errors automatically. Many programming languages like C++, Objective-C, PHP, Java, Ruby, Python, and many others have built-in support for exception handling.

Error handling is generally resolved by saving the state of execution at the moment the error occurred and interrupting the normal flow of the program to execute a special function or piece of code, which is known as the exception handler. Depending on the kind of error ("division by zero", "file open error" and so on) which had occurred, the error handler can "fix" the problem and the programm can be continued afterwards with the previously saved data.

End of Class 1