

1. Why one should learn Data Science & Machine Learning?

- Data Science is a field that encompasses related to data cleansing, preparation, and analysis. Data science is an umbrella term in which many scientific methods apply. For example, mathematics, statistics, and many other tools scientists apply to data sets. Scientist applies the tools to extract knowledge from data.
- Machine learning is the shining star of the moment. With every industry looking to apply AI in their domain, studying machine learning opens world of opportunities to develop cutting edge machine learning applications in various verticals – such as cyber security, image recognition, medicine, or face recognition.

2. What is the Difference Between Data Science, Deep Learning & Artificial Intelligence?

- Data science is a cousin to statistics, which incorporates tools like statistical modelling, hypothesis testing, visualization methods, and machine learning algorithms. Artificial intelligence is a branch of computer science that sometimes uses machine learning as a tool to help computers and robots learn from their environment. You can think of a Venn Diagram with two mostly non-overlapping circles, data science and AI. The overlapping part is machine learning (and likely some Bayesian methods these days).

3. What are the applications of Data Science and Machine Learning in Businesses?

- Fraud and Risk Detection.
- Healthcare.
- Internet Search.
- Targeted Advertising.
- Website Recommendations.
- Advanced Image Recognition.
- Speech Recognition.
- Airline Route Planning.

4. Which programming language is better R or Python and Why?

- Which is easier to learn? R has a steep learning curve, and people without programming experience may find it overwhelming. Python is generally considered easier to pick up.
- Another advantage of Python is that it is a more general programming language: For those interested in doing more than statistics, this comes in handy for building a website or making sense of command-line tools. The way Python works reflects the way computer programmers think. R, on the other hand, reflects its origins in statistics. Many programmers find the design of R irritating, because it's so different to what they're used to, Groskopf says. For someone interested in becoming a general-purpose programmer, Python is a better choice.
- But for data analysis, the differences between R and Python are starting to break down, he says. Most of the common tasks once associated with one program or the other are now doable in both. They are similar enough, in fact, that if most of your colleagues are already using R or Python, you should probably just pick up that language.
- So, the great R-versus-Python debate is settled. If all you're doing is data analysis, it doesn't really matter which one you use.

5. Which is your favourite Data Science Model and why?

- I am a fan of simple yet flexible algorithms. If I had to choose one, I'd say my favourite algorithm is the Ensemble, which I consider my own "Master Algorithm". Whatever algorithm you start from, you can always use an ensemble to improve it. Ensembles won the Netflix Prize and routinely show their great performance, but they are also relatively easy to understand, optimize, and inspect.
- In case I am accused of cheating for choosing a "super algorithm", I will choose another one: Logistic Regression. LR is a very simple but efficient and flexible algorithm that can be used for many applications, notably classification, but also ranking.

6. What are Data Structures in R & Python and their Usage?

- Data Structures in R

- i. Vector: Vector is the most basic data structure in R programming language. It comes in two parts: Atomic vectors and Lists. They have three common properties:
 - ~Type function – what actually it is?
 - ~Length function – how many elements does it contain.
 - ~Attribute function – extra arbitrary metadata.
- ii. Matrix: A matrix is a two-dimensional rectangular data set and thus it can be created using vector input to the matrix function. In addition, a matrix is a collection of numbers arranged into a fixed number of rows and columns. Usually, the numbers are the real numbers. By using a matrix function, we can reproduce a memory representation of the matrix in R. Hence, the data elements must be of the same basic type.
- iii. Array: In R Programming, arrays are multi-dimensional Data structures. In an array, data is stored in the form of matrices, row, and as well as in columns. We can use the *matrix level, row index, and column index* to access the matrix elements. Arrays in R are the data objects which can store data in more than two dimensions. An array is created using the *array ()* function. We can use vectors as input. To create an array, we can use these values in the dim parameter. For example: In this following example, we will create an array in R of two 3×3 matrices each with 3 rows and 3 columns.
- iv. Lists: Lists are the objects which contain elements of different types – like strings, numbers, vectors and another list inside them. A list can also contain a matrix or a function as its elements. In other words, a list is a generic vector containing other objects. A list is created using the *list ()* function. For example: The variable x is containing copies of three vectors n, s, b and a numeric value 3.
- v. Data Frames: First of all, we are going to discuss where the concept of data frame came. The concept comes from the world of the statistical software used in empirical research. It generally refers to tabular data: a data structure representing the cases (rows), each of which consists of numbers of observation or measurement (columns). A data frame is used for storing data tables. It is a list of vectors of equal length.

- Data Structure in Python

- i. Lists: Lists in Python are one of the most versatile collection object types available. The other two types are dictionaries and tuples, but they are really more like variations of lists. Python lists do the work of most of the collection data structures found in other languages and since they are built-in, you don't have to worry about manually creating them.
- ii. Dictionary: In python, dictionary is similar to hash or maps in other languages. It consists of key value pairs. The value can be accessed by unique key in the dictionary. Keys are unique & immutable objects.
- iii. Tuple: Python tuples work exactly like Python lists except they are immutable, i.e. they can't be changed in place. They are normally written inside parentheses to distinguish them from lists (which use square brackets), but as you'll see, parentheses aren't always necessary. Since tuples are immutable, their length is fixed. To grow or shrink a tuple, a new tuple must be created.
- iv. Sets: Unordered collection of unique objects. Set operations such as union (*|*), intersection (*&*), difference (*-*) can be applied on a set. Sets are immutable i.e. once created further data can't be added to them. *()* are used to represent a set. Objects placed inside these brackets would be treated as a set.

7. What are Methods in Python?

- A method is a function that “belongs to” an object. (In Python, the term method is not unique to class instances: other object types can have methods as well. For example, list objects have methods called `append`, `insert`, `remove`, `sort`, and so on.