**Day 1 to Day 7: Python Basics**

* **Python basics all**
* **OOP Concepts**

**Read from Owns collections.**

**Day 8 to Day 17: PANDAS Library**

* Creating a data frame, reading data from a file, and writing a data frame to a file
* Indexing and Selection of data from a data frame
* Iteration and Sorting
* Aggregation and Group By
* Missing Values and handling of missing values
* Renaming and Replacing in Pandas
* Concatenating, Merging, and Joining in a data frame
* Summary Analysis, Cross Tabulation, and Pivot
* Date, Categorical and Sparse Data

**Day 18 to Day 22: NUMPY Library**

* Creation of an Array
* Indexing and Slicing
* Data Types
* Joining and splitting
* Searching and Sorting
* Filtering required data elements

**Day 23 to Day 25: Visualizations**

Now its time to spend some quality time understanding and using some of the key visualization libraries like ggplot, Plotly, and Seaborn. Use a sample dataset and try different visualization like Bar Chart, Line/Trend Chart, Box Plot, Scatter Plots, Heatmaps, Pie Chart, Histogram, Bubble Charts, and other interesting or interactive visualizations

**Day 26 to Day 35: Statistics, Implementation, and Use-cases:**

The next important topic to be covered is Statistics, explore the descriptive statistics techniques that are commonly used such as Mean, Median, Mode, Range Analysis, Standard Deviations, and Variances.

Then cover slightly deeper techniques such as identifying the Outliers in the Dataset and measuring the Margin of Error.

* F-test
* ANOVA
* Chi-Squared Test
* T-Test
* Z-Test

**Day 36 to Day 40: SQL for Data Analysis:**

* Selecting data from a table
* Joining data from different tables based on a key
* Performing Group by and Aggregation functions on data
* Use of case statements and filter conditions

**Day 41 to Day 50: Exploratory Data Analysis (EDA)**

In any Data Science project, about 80% of the time is spent in this activity so it is best to spend time learning this topic thoroughly. In order to learn the Exploratory Data Analysis, there isn’t a specific set of functionalities or topics to be covered but the dataset and the use-case would drive the analysis. Hence it would be preferred to use some sample dataset from competitions hosted in kaggle and learn to perform exploratory analysis.

Another method to learn exploratory data analysis is to write your questions about the dataset and try to find answers for them from the dataset. Like if I consider the most popular Titanic Dataset then try to find answers for questions like people of which gender/age/deck had a higher probability of dying and so on. Your ability to perform a thorough analysis would improve with time so be patient and learn slowly and confidently.

By now you have learned all the core skills required for a Data Scientist, now you are ready to learn Algorithms.

**What happened to Mathematics?**

Yes it is important to know about Linear Algebra and Calculus but I would prefer not to spend time learning mathematics concepts but as and when they are required you can refer and brush up your skills, High-School Level of mathematics would be sufficient. For example, let’s say you are learning about Gradient Descent then while learning the algorithm you can spend time learning about the mathematics behind it. Because if you start learning the important concepts in mathematics then it could be very time consuming and moreover by learning as and when required you would learn just enough required for the time but instead if you start learning all concepts in mathematics then you would be spending way more time and would be learning way more than what is required.

**Day 51 to Day 70: Supervised Learning and Project Implementation**

Spend the first 10 days in knowing some of the key algorithms in Supervised Learning, understand the math behind them, and in the next 10 days focus on learning by developing a project. Some of the Algorithms that should be covered in this period are,

* Linear Regression and Logistic Regression
* Decision Tree / Random Forest
* Support Vector Machine (SVM)

In the first 10 days, the focus should be on understanding the theory behind the algorithms you have chosen. Then spend some time understanding the scenarios where each of the algorithms would be more suitable as compared with others like Decision Trees are best when there are a lot of categorical attributes in the dataset.

Then pick a solved example in Kaggle, you will be able to find ample solved examples try to re-execute them but carefully understand each and every line in the code and understand the reason for them. By now you have got good theoretical knowledge as well as working knowledge from the solved examples.

As a final step, pick a project, and implement a supervised learning algorithm, start with data collection, exploratory analysis, feature engineering, model building, and model validation. There will definitely be a lot of questions and issues but when you complete the project you would have got a very good knowledge about the algorithm and the methodologies

**Day 71 to Day 90: Unsupervised Learning and Project Implementation**

Now its time to focus on unsupervised learning, similar to the method used in supervised learning spend the initial days in understanding the concepts behind the algorithms you have chosen in unsupervised learning, and then learn by implementing a project.

The algorithm that should be covered here are,

* Clustering Algorithm — Used to identify Clusters in the dataset
* Association Analysis — Used to identify patterns in the data
* Principal Components Analysis — Used to reduce the number of attributes
* Recommendation System — Used to identify similar users/products and to make recommendations

*In the initial days, the focus should be on* understanding each of the above algorithms and techniques also to understand the purpose of each of them and the scenarios where they can be used like principal components analysis generally used for dimensionality reduction when the dataset you are working is having a very large number of columns and you would want to reduce it but still retain the information from them and recommendation systems are popular in e-commerce where based on the purchase patterns of a customer other items that they would likely be interested in could be recommended to increase sales.

When you are comfortable with the theory and the scenarios where they can be used then it is time to pick a solved example and learn by reverse engineering them that is understanding each and every line of code and re-executing them.

As a final step now its time to pick a use-case and implement based on your learnings so far. On completing the project/use-case you would have learned a lot and you would have gained a much better understanding of these algorithms and that would remain with you forever.

**Day 91 to Day 100: Natural Language Processing Basics**

Make use of this time to focus on analysis and use-cases for unstructured / text data. Few things worth spending time here would include

* Learn to use API to fetch data from the public sources
* Perform a few basic sentiments analysis — Data from twitter API can be used to extract tweets of a particular hashtag and then the sentiment and the emotions behind those tweets can be computed
* Topic Modelling — This is useful when there are a large number of document and you want to group them into different categories that this method would come handy

Python Basics: 2 days

Pandas: 3 days

NUMPY: 1 days

Visualizations: 1 days

Statistics, implementation and use case: 4 days

SQL: 1 days

EDA: 3 days

Supervised learning and project implementation: 7 days

Unsupervised learning and project implementations: 7 days

Natural language processing: 3 days

====================================

Total day: 32 days

Complete data science in 1 month.

After this start to do advance projects. Projects of finance, real state etc.