LET THE DATA CONFESS

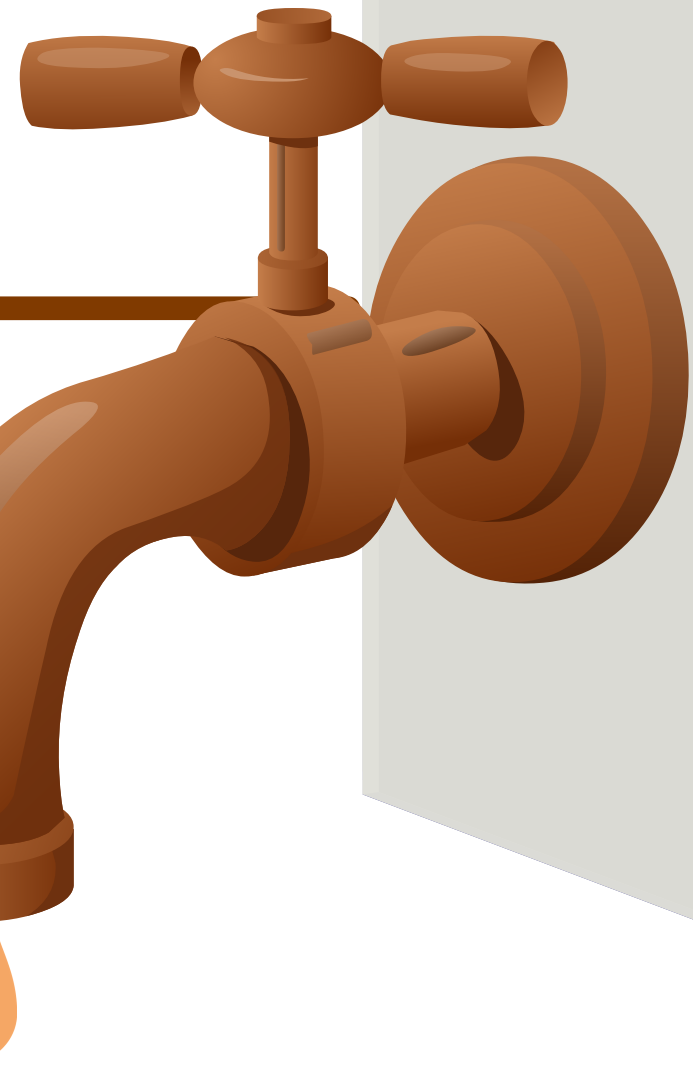
Understand | Learn | Code | Implement

Do you wanna know about

DATA LEAKAGE &

How does it affect Performance?

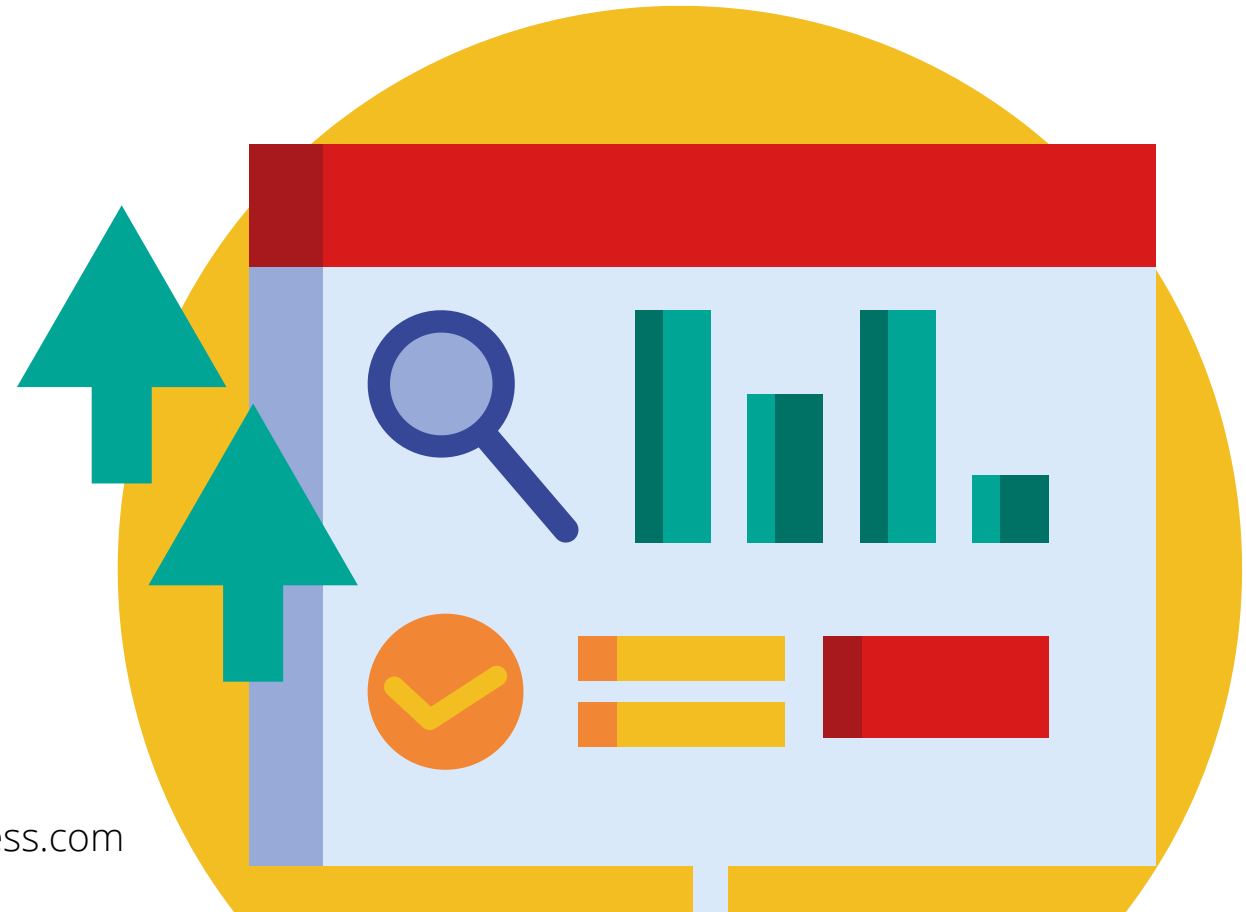
-letthedataconfess



What is Data Leakage ?

- It is the scenario where the machine learning model is already aware of some part of test data after training.

- This causes the problem of **overfitting**.

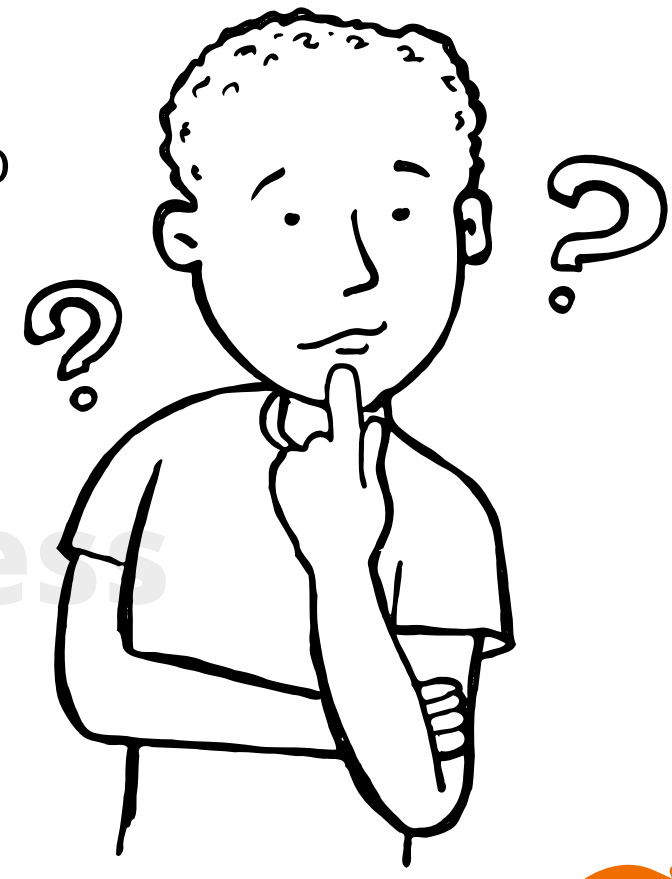


How does it exactly Happen?

- When you split your data into train and test set, some of your data present in the test set is also copied in the train set and vice versa.

- As a result of which when you train your model with this type of split it will give really good results on the train and test set.

- But when you deploy your model into production it will not perform well, because when a new type of data comes it won't be able to handle it.



When can it Happen ?

This generally happens when we use bad practices during the preprocessing of the data. Some of them are given below:

Post splitting:

If you initially perform all the preprocessing and then split the data into train and test, while splitting, the data is selected randomly so some of the preprocessed data from the test set go to the train set which causes data leakage as the model will be knowing some part of the test data.

Duplication of data:

If identical data points have occurred multiple times inside the dataset then it creates duplication of data. Now train and test will also contain some of these points causing data leakage.

Random split in time-series:

In time series our data is dependent on the previous one so A->B->C->D. In this case, while splitting, A, C may go to the train set while B, D may go to the test set. As they are interdependent, so even after splitting, A, B, C, D are aware of each other.

How to fix this?

- The main culprit behind this is the way we split our dataset and when. If we combine our train and test data, then perform the feature selection process and split it, data leakage may happen. To avoid this you can do the following-

- **Pre Splitting:**

  iIn this, instead of doing feature engineering process on the complete dataset, first split the data into train and test then go ahead and perform feature engineering.

- **Cross-validation:**

  Train and test split always vary the accuracy due to random selection of data, so it's always better to perform cross-validation because it will find the best split for train & test by performing multiple experiments of splitting. This also ensures that test and train data don't mix up.

- **Create a separate validation set:**

  It is always recommended to make a separate split called validation set which is different from the test set. This data should be split out in the early stage itself so that our train and test split should not contain any data from the validation split at all. And during testing, we can pass the validation set just to check if our model is performing well.

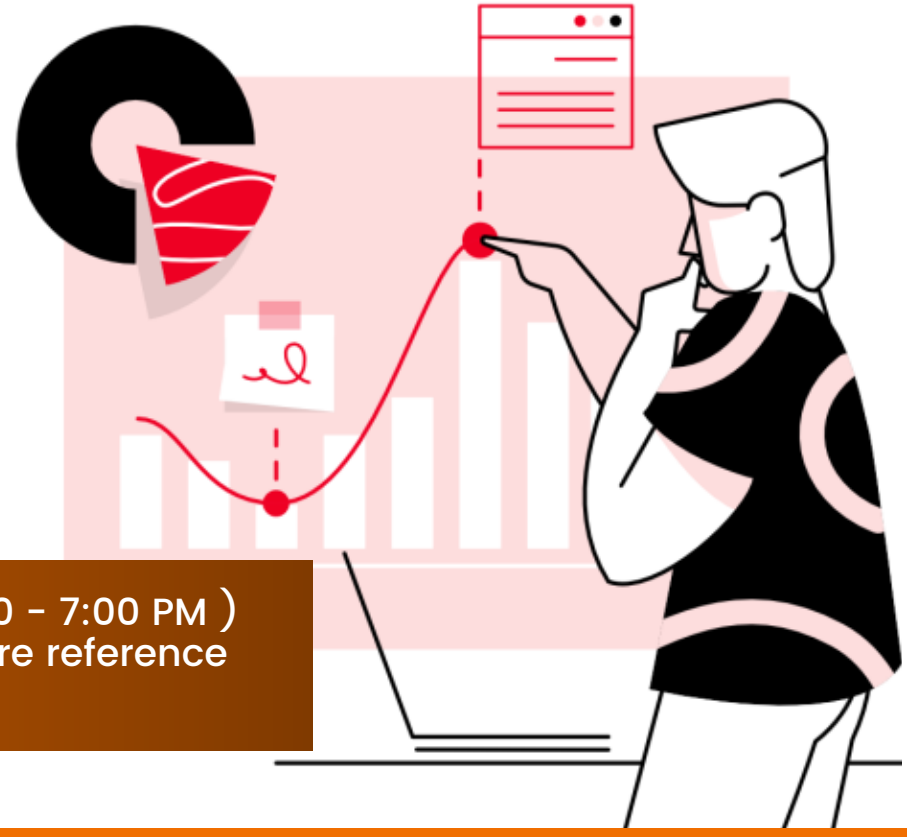
LET THE DATA CONFESS
Understand | Learn | Code | Implement

Exploratory Data Analysis Using Python

Let's learn how to make the data confess!

DATE
12th & 13th JUNE

4+1 HOURS
04.00 - 06.00 PM
Each day



Speaker
Arpita Gupta
Founder, Let The Data Confess PVT. LTD.

Bonus : 1. 1 hour doubt session on 2nd day (6:00 - 7:00 PM)
2. Content of the Workshop for your future reference
3. Certificate to every participants

FEE : 299/- ₹

www.letthedataconfess.com

LET THE DATA CONFESS

Understand | Learn | Code | Implement

www.letthedataconfess.com

LET THE DATA CONFESS

Understand | Learn | Code | Implement

Follow | Support