Data Preprocessing-Data Shuffling

Data Shuffling

Data shuffling is a technique used in data preprocessing to randomize the order of rows or observations in a dataset. This process helps to prevent any bias that may be present due to the ordering of data.

Why is Data Shuffling necessary?

1. **Prevents Bias**: When data is ordered by a particular attribute, it can introduce bias into algorithms that analyze the data. For instance, if we're analyzing sales data and our data is ordered chronologically, some models may perform better on earlier or later dates.
2. **Improves Model Performance**: Shuffling the data helps ensure that the model performance is not skewed due to the initial order of data.

Example Use Case

Suppose you have a dataset containing information about customer purchases:

Customer ID	Purchase Date	Product Purchased
1	2022-01-01	Product A
2	2022-01-02	Product B
3	2022-01-03	Product C

If you're building a predictive model to forecast sales, you might want to shuffle the data first:

```
 import pandas as pd

# Create a sample DataFrame
data = {
    "Customer ID": [1, 2, 3],
    "Purchase Date": ["2022-01-01", "2022-01-02", "2022-01-03"],
    "Product Purchased": ["A", "B", "C"]
}
df = pd.DataFrame(data)

print("Before Shuffling:")
print(df)

# Shuffle the data
df_shuffled = df.sample(frac=1).reset_index(drop=True)

print("\nAfter Shuffling:")
print(df_shuffled)
```

Before Shuffling

Customer ID	Purchase Date	Product Purchased
1	2022-01-01	A
2	2022-01-02	B
3	2022-01-03	C

After Shuffling

Customer ID	Purchase Date	Product Purchased
2	2022-01-03	C
3	2022-01-02	B
1	2022-01-01	A

In the shuffled dataset, the order of rows is randomized, ensuring that any model built on this data will not be biased due to the original ordering.

Curated by Brajesh Kumar