Training Day – 44

\*November 18, Monday\*

* \*Topic:\* Combining Multiple Datasets
* Consolidated datasets into a single clean dataset.
* Example: Used a combination of concat() and merge() for integration.

Combining datasets from different sources or files is a common task in data cleaning and analysis. By integrating datasets into one consolidated clean dataset, you can work with a complete set of information for further analysis or modeling. Two commonly used methods for combining datasets are concat() and merge() functions in Python, particularly with the Pandas library.

1. Concatenating Datasets with concat()

The concat() function is used to combine datasets along a particular axis (rows or columns). It’s useful when datasets have the same structure (e.g., same columns) but come from different sources or time periods.

# Example: Concatenating DataFrames by Rows

Suppose you have two DataFrames with identical columns but different rows (e.g., two sets of data collected over different months).

import pandas as pd

# Sample DataFrames df1 = pd.DataFrame({

'ID': [1, 2, 3],

'Value': [10, 20, 30]

})

df2 = pd.DataFrame({

'ID': [4, 5, 6],

'Value': [40, 50, 60]

})

# Concatenate by rows (axis=0)

df\_combined = pd.concat([df1, df2], axis=0, ignore\_index=True) print(df\_combined)

# Example: Concatenating DataFrames by Columns

If your datasets contain different features (columns), you can concatenate them side by side.

# Concatenate by columns (axis=1) df\_combined\_columns = pd.concat([df1, df2], axis=1) print(df\_combined\_columns)

Output:

Copy code

ID Value ID Value

1. 1 10 4 40
2. 2 20 5 50
3. 3 30 6 60

2. Merging Datasets with merge()The merge() function is used when datasets share common columns, and you want to combine them based on matching values. It’s similar to a SQL join (inner, outer, left, or right join).

#  Example: Merging DataFrames on Common Columns

If you have two DataFrames with a common column (e.g., "ID"), you can merge them to consolidate their information.

python Copy code df1 = pd.DataFrame({

'ID': [1, 2, 3],

'Name': ['Alice', 'Bob', 'Charlie']

})

df2 = pd.DataFrame({

'ID': [1, 2, 4],

'Value': [100, 200, 300]

})

# Merge on 'ID' df\_merged = pd.merge(df1, df2, on='ID', how='inner') print(df\_merged)

Output:

Copy code

ID Name Value

0 1 Alice 100 1 2 Bob 200