**1. Data importing and preprocessing**

* Import cleaned data
* Required libraries are imported
* Check for GPU availability to be used by the model
* Data split into features (X) and target (y)
* Data split into teain\_validation and test
* Feature scaling by StandardScaler

Requirement already satisfied: optuna in /usr/local/lib/python3.12/dist-packages (4.5.0)

Requirement already satisfied: alembic>=1.5.0 in /usr/local/lib/python3.12/dist-packages (from optuna) (1.16.5)

Requirement already satisfied: colorlog in /usr/local/lib/python3.12/dist-packages (from optuna) (6.9.0)

Requirement already satisfied: numpy in /usr/local/lib/python3.12/dist-packages (from optuna) (2.0.2)

Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.12/dist-packages (from optuna) (25.0)

Requirement already satisfied: sqlalchemy>=1.4.2 in /usr/local/lib/python3.12/dist-packages (from optuna) (2.0.43)

Requirement already satisfied: tqdm in /usr/local/lib/python3.12/dist-packages (from optuna) (4.67.1)

Requirement already satisfied: PyYAML in /usr/local/lib/python3.12/dist-packages (from optuna) (6.0.3)

Requirement already satisfied: Mako in /usr/local/lib/python3.12/dist-packages (from alembic>=1.5.0->optuna) (1.3.10)

Requirement already satisfied: typing-extensions>=4.12 in /usr/local/lib/python3.12/dist-packages (from alembic>=1.5.0->optuna) (4.15.0)

Requirement already satisfied: greenlet>=1 in /usr/local/lib/python3.12/dist-packages (from sqlalchemy>=1.4.2->optuna) (3.2.4)

Requirement already satisfied: MarkupSafe>=0.9.2 in /usr/local/lib/python3.12/dist-packages (from Mako->alembic>=1.5.0->optuna) (3.0.3)

GPU Available: False

Running on CPU

Using device: cpu

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force\_remount=True).

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

First five rows of X and y:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

First five rows of X:

Inventory Level Units Sold Units Ordered Price Discount \

0 231 127 55 33.50 20

1 204 150 66 63.01 20

2 102 65 51 27.99 10

3 469 61 164 32.72 10

4 166 14 135 73.64 0

Holiday/Promotion Competitor Pricing Store ID\_S001 Store ID\_S002 \

0 0 29.69 1.0 0.0

1 0 66.16 1.0 0.0

2 1 31.32 1.0 0.0

3 1 34.74 1.0 0.0

4 0 68.95 1.0 0.0

Store ID\_S003 ... Weather Condition\_Cloudy Weather Condition\_Rainy \

0 0.0 ... 0.0 1.0

1 0.0 ... 0.0 0.0

2 0.0 ... 0.0 0.0

3 0.0 ... 1.0 0.0

4 0.0 ... 0.0 0.0

Weather Condition\_Snowy Weather Condition\_Sunny Seasonality\_Autumn \

0 0.0 0.0 1.0

1 0.0 1.0 1.0

2 0.0 1.0 0.0

3 0.0 0.0 1.0

4 0.0 1.0 0.0

Seasonality\_Spring Seasonality\_Summer Seasonality\_Winter dayofweek\_sin \

0 0.0 0.0 0.0 -0.974928

1 0.0 0.0 0.0 -0.974928

2 0.0 1.0 0.0 -0.974928

3 0.0 0.0 0.0 -0.974928

4 0.0 1.0 0.0 -0.974928

dayofweek\_cos

0 -0.222521

1 -0.222521

2 -0.222521

3 -0.222521

4 -0.222521

[5 rows x 51 columns]

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

First five rows of y:

0 135.47

1 144.04

2 74.02

3 62.18

4 9.26

Name: Demand Forecast, dtype: float64

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Results of above cell show that the data imported properly**

# 2. Model building - PyTorch

[I 2025-10-05 22:23:57,433] A new study created in memory with name: no-name-fea3e7fc-8c9e-46e9-9f73-c98269fc9de0

[I 2025-10-05 22:25:28,446] Trial 0 finished with value: 167.54812622070312 and parameters: {'num\_layers': 1, 'hidden\_units': 96, 'activation': 'tanh', 'learning\_rate': 0.0018439709265418408, 'batch\_size': 64}. Best is trial 0 with value: 167.54812622070312.

[I 2025-10-05 22:30:33,204] Trial 1 finished with value: 144.08355712890625 and parameters: {'num\_layers': 2, 'hidden\_units': 112, 'activation': 'tanh', 'learning\_rate': 0.0010141772723690407, 'batch\_size': 16}. Best is trial 1 with value: 144.08355712890625.

[I 2025-10-05 22:34:09,047] Trial 2 finished with value: 76.54293823242188 and parameters: {'num\_layers': 1, 'hidden\_units': 80, 'activation': 'relu', 'learning\_rate': 0.00012793678980646986, 'batch\_size': 16}. Best is trial 2 with value: 76.54293823242188.

[I 2025-10-05 22:37:14,095] Trial 3 finished with value: 92.15948486328125 and parameters: {'num\_layers': 3, 'hidden\_units': 112, 'activation': 'tanh', 'learning\_rate': 0.00020668607895091818, 'batch\_size': 32}. Best is trial 2 with value: 76.54293823242188.

[I 2025-10-05 22:39:18,250] Trial 4 finished with value: 107.26720428466797 and parameters: {'num\_layers': 1, 'hidden\_units': 112, 'activation': 'tanh', 'learning\_rate': 0.00012517748390491454, 'batch\_size': 32}. Best is trial 2 with value: 76.54293823242188.

[I 2025-10-05 22:41:44,843] Trial 5 finished with value: 76.27877044677734 and parameters: {'num\_layers': 2, 'hidden\_units': 48, 'activation': 'relu', 'learning\_rate': 0.00046990542249259354, 'batch\_size': 32}. Best is trial 5 with value: 76.27877044677734.

[I 2025-10-05 22:45:16,673] Trial 6 finished with value: 75.43843078613281 and parameters: {'num\_layers': 1, 'hidden\_units': 32, 'activation': 'relu', 'learning\_rate': 0.000341234845822625, 'batch\_size': 16}. Best is trial 6 with value: 75.43843078613281.

[I 2025-10-05 22:48:49,323] Trial 7 finished with value: 617.2195434570312 and parameters: {'num\_layers': 1, 'hidden\_units': 16, 'activation': 'tanh', 'learning\_rate': 0.001188877776261688, 'batch\_size': 16}. Best is trial 6 with value: 75.43843078613281.

[I 2025-10-05 22:51:20,361] Trial 8 finished with value: 131.13430786132812 and parameters: {'num\_layers': 2, 'hidden\_units': 48, 'activation': 'tanh', 'learning\_rate': 0.0025996296943778152, 'batch\_size': 32}. Best is trial 6 with value: 75.43843078613281.

[I 2025-10-05 22:56:05,485] Trial 9 finished with value: 79.87484741210938 and parameters: {'num\_layers': 2, 'hidden\_units': 48, 'activation': 'relu', 'learning\_rate': 0.009570356814070901, 'batch\_size': 16}. Best is trial 6 with value: 75.43843078613281.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Best hyperparameters:

num\_layers: 1

hidden\_units: 32

activation: relu

learning\_rate: 0.000341234845822625

batch\_size: 16

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Final Evaluation:

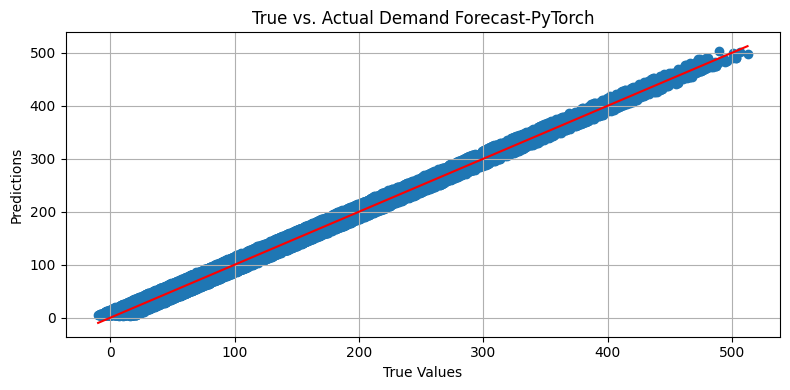
Mean Squared Error (MSE): 75.71

R-squared (R²): 0.99

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# 3. Performance analysis of the model by below plots

* Actual vs predicted
* Residual error

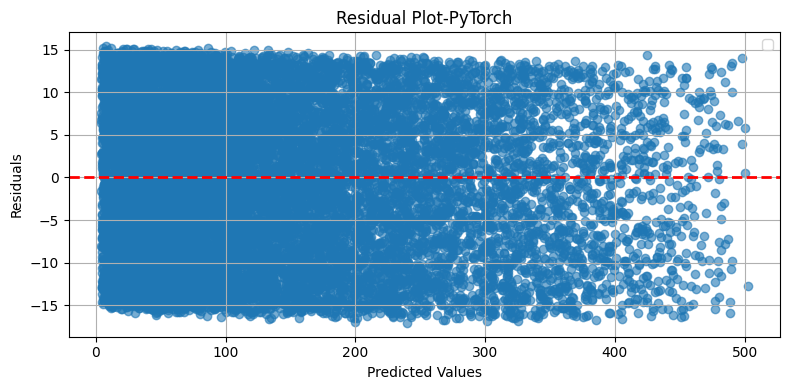


/tmp/ipython-input-2046409224.py:15: DeprecationWarning: \_\_array\_wrap\_\_ must accept context and return\_scalar arguments (positionally) in the future. (Deprecated NumPy 2.0)

residuals = y\_test - y\_pred

/tmp/ipython-input-2046409224.py:22: UserWarning: No artists with labels found to put in legend. Note that artists whose label start with an underscore are ignored when legend() is called with no argument.

plt.legend()



**According to plots and results of the model, it has been trained well.**

# 6. Save the best mode

Model saved successfully!