Course: Data Science Tools and Techniques

Data Preprocessing

Dr. Safdar Ali

Explore and discuss the process of data cleaning, with understanding of its importance, common challenges, and effective techniques along with data transformation.

Automating Data Preprocessing Pipelines

Automating Data Preprocessing Pipelines?

- Automating repetitive preprocessing steps to improve efficiency.
- Ensured consistency in data preparation
- Reduced manual intervention and human errors
- Faster model development
- Easily adaptable for new data sources
- Scalable for large datasets and machine learning workflows

Main Stages in Data Preprocessing Pipelines

- Handling missing values, etc.
- Encoding categorical data
- Feature scaling (normalization, standardization)
- Feature engineering
- Splitting data into training/testing sets
- Automating above stages using Python libraries.

Python Code for Automating Preprocessing with Pipelines

```
from sklearn.pipeline import Pipeline


from sklearn.preprocessing import
StandardScaler, OneHotEncoder


from sklearn.impute import SimpleImputer


from sklearn.compose import
ColumnTransformer


import pandas as pd
```

```
# Sample Data
df = pd.DataFrame({'Age': [25, np.nan,
30, 35], 'City': ['NY', 'LA', 'SF', 'NY']})
# Define Transformers
num_pipeline = Pipeline([('imputer',
SimpleImputer(strategy='mean')),
('scaler', StandardScaler())
])
cat_pipeline = Pipeline([('imputer',
SimpleImputer(strategy='most_frequent'))
, ('encoder', OneHotEncoder())
])
# Combine Pipelines
preprocessor = ColumnTransformer([
    ('num', num_pipeline, ['Age']),
    ('cat', cat_pipeline, ['City'])
])
# Apply Transformation
df_transformed =
preprocessor.fit_transform(df)
print(df_transformed)
```

Tracking Pipelines with MLflow

- Track preprocessing steps & experiments using MLflow.
- Allows reproducibility in machine learning workflows.
- Manage versions of preprocessing pipelines.

```
import mlflow
import mlflow.sklearn

# Start an MLflow experiment
mlflow.start_run()

# Log preprocessing pipeline
mlflow.sklearn.log_model(preprocessor, 'preprocessing_pipeline')
mlflow.end_run()
```