MINI PROJECT-1

DATA PREPEOCESSING AND

FEATURE ENGINEERING

NAME: K. VARSHITHA

EMAIL ID: [koramutlavarshitha90@gmail.com](mailto:koramutlavarshitha90@gmail.com)

BATCH: ARTIFICIAL INTELLIGENCE & MACHINE LEARNING(Feb-24-2024)

PROJECT NAME: DATA PREPROCESSING AND FEATURE ENGINEERING (Techniques applying on real world dataset for machine learning modeling).

### What is AI (Artificial Intelligence)?

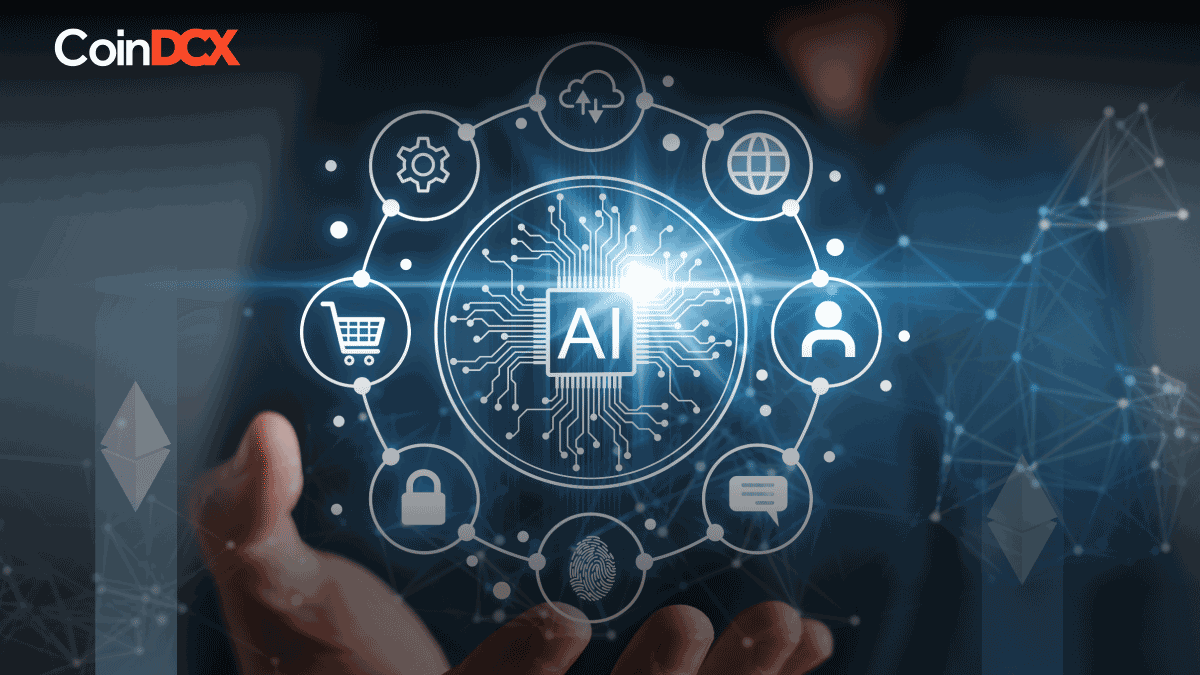


Image-1

***Artificial Intelligence (AI)*** *refers to the simulation of human intelligence in machines that are programmed to think and act like humans. AI can be categorized into several types:*

1. ***Narrow AI****: Also known as Weak AI, it is designed to perform a narrow task (e.g., facial recognition, internet searches, or self-driving cars). Narrow AI systems can outperform humans at specific tasks they are designed for, but they operate under a limited set of constraints and conditions.*
2. ***General AI****: Also known as Strong AI or AGI (Artificial General Intelligence), it refers to a type of AI that has the ability to understand, learn, and apply knowledge across a wide range of tasks at a level equal to humans. This type of AI does not yet exist.*
3. ***Superintelligent AI****: A hypothetical AI that surpasses human intelligence and capability in every field. It would be able to outperform the best human minds in every field, including scientific creativity, general wisdom, and social skills.*

**Applications of AI:**

* ***Healthcare****: Diagnosis of diseases, personalized treatment plans, drug discovery.*
* ***Finance****: Fraud detection, algorithmic trading, customer service.*
* ***Transportation****: Autonomous vehicles, traffic management, logistics optimization.*
* ***Entertainment****: Recommendation systems, content creation, interactive gaming.*
* ***Customer Service****: Chatbots, virtual assistants, sentiment analysis*.

### What is ML (Machine Learning)?

***Machine Learning (ML)*** *is a subset of AI that involves the use of statistical techniques to enable machines to improve at tasks with experience. It focuses on the development of algorithms that can analyse and learn from data to make predictions or decisions.*



Image.2

Types of Machine Learning:

1. ***Supervised Learning****: The algorithm is trained on a labelled dataset, which means that each training example is paired with an output label. Common algorithms include linear regression, logistic regression, support vector machines, and neural networks.*
   * ***Examples****: Spam detection, image classification, house price prediction.*
2. ***Unsupervised Learning****: The algorithm is used to find patterns and relationships in data that is not labelled. Common algorithms include clustering techniques like K-means and hierarchical clustering, as well as association algorithms.*
   * ***Examples****: Market basket analysis, customer segmentation, anomaly detection.*
3. ***Semi-Supervised Learning****: This approach uses a small amount of labelled data and a large amount of unlabelled data to improve learning accuracy. It is often used when acquiring a fully labelled dataset is expensive or time-consuming.*
   * ***Examples****: Web content classification, speech recognition.*
4. ***Reinforcement Learning****: The algorithm learns by interacting with its environment and receiving rewards or penalties based on its actions. It is used in areas where the sequence of actions is critical.*
   * ***Examples****: Game playing (e.g., AlphaGo), robotic control, autonomous driving.*

***Key Concepts in Machine Learning:***

* ***Feature Engineering****: The process of selecting, modifying, or creating new features to improve the performance of a machine learning model.*
* ***Model Training****: The process of fitting a machine learning model to a training dataset.*
* ***Evaluation Metrics****: Metrics like accuracy, precision, recall, F1-score, and ROC-AUC used to evaluate the performance of a model.*
* ***Overfitting and Underfitting****: Overfitting occurs when a model learns the training data too well, including noise and outliers, resulting in poor generalization to new data. Underfitting occurs when a model is too simple to capture the underlying patterns in the data.*

Applications of Machine Learning**:**

* ***Natural Language Processing (NLP)****: Language translation, sentiment analysis, chatbots.*
* ***Computer Vision****: Object detection, facial recognition, medical image analysis.*
* ***Recommendation Systems****: Personalized recommendations in e-commerce and streaming services.*
* ***Predictive Analytics****: Predicting customer churn, maintenance schedules, financial forecasts.*

*In summary, AI is a broad field that encompasses any technique enabling computers to mimic human intelligence, while ML is a subset of AI focused on algorithms that learn from and make predictions on data. Both fields are rapidly evolving and have numerous applications across various industries.*

* OBJECTIVE:

*The objective of this project is to apply Data preprocessing and Feature engineering techniques on a Real-world Dataset to prepare it for Machine learning modeling.*

# Data collection:

*Data collection is the process of gathering and measuring information on variables of interest, in a systematic manner, for the purpose of research, analysis, or decision-making. It involves several steps and methods to ensure that the data collected is accurate, reliable, and relevant to the objectives of the study or project.*

* Data Collection Methods: *Choosing the methods and tools for collecting data, which can vary widely depending on the nature of the data and the research objectives. Common methods include surveys, interviews, observations, experiments, and sensor data collection.*
* Data Quality Assurance: *Implementing measures to ensure the quality and validity of the data collected, such as using standardized protocols, training data collectors, conducting pilot studies, and using validation checks.*
* Ethical Considerations: *Ensuring that data collection is conducted ethically and respects the rights and privacy of participants. This may involve obtaining informed consent, protecting confidentiality, and adhering to relevant regulations and guidelines.*
* Data Recording and Storage: *Properly recording data in a structured format and storing it securely to prevent loss or unauthorized access. This may involve using data management tools and ensuring compliance with data protection regulations.*
* Data Cleaning and Preprocessing*: After collection, data often* *requires cleaning and preprocessing to correct errors, handle missing values, and transform it into a format suitable for analysis.*
* WHAT IS DATASET?

*A dataset is a collection of data, typically organized into a structured format, that is used for various purposes such as analysis, research, machine learning, or decision-making. Datasets can come from various sources and can be of different types.*

# Types of datasets

* *\*\*Structured Datasets\*\*: These contain organized data with a clear format, such as databases, spreadsheets, or CSV files.*
* *\*\*Unstructured Datasets\*\*: These contain data that lacks a predefined structure, such as text documents, images, videos, or audio recordings.*

*Datasets are fundamental in many fields including statistics, machine learning, data science, and research. They provide the raw material for analysis, training models, testing hypotheses, and deriving insights. Examples of datasets include collections of customer information for market analysis, image datasets for training computer vision models, and genomic datasets for bioinformatics* *research.*

WHAT IS DATA PREPROCESSING?

* *Data preprocessing is a critical step in the data analysis and machine learning pipeline. It involves transforming raw data into a clean, consistent, and usable format to improve the quality and performance of subsequent analyses and models.*

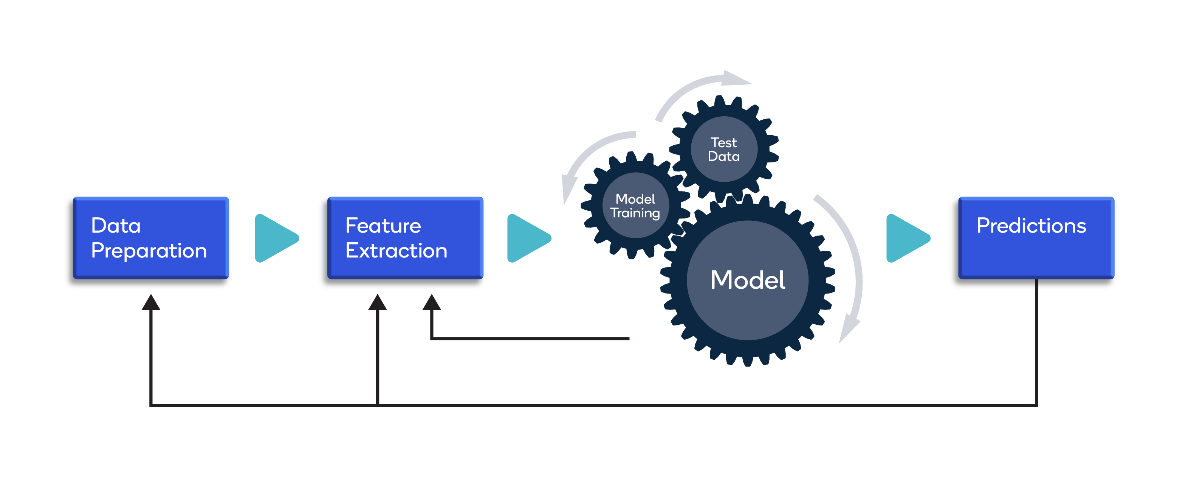


Image .3

# KEY STEPS IN DATA PREPROCESSING:

1. *Data Cleaning:*
   * ***Handling Missing Values:*** *Identifying and filling or removing missing data points.*
   * ***Removing Duplicates:*** *Identifying and removing duplicate entries.*
   * ***Outlier Detection and Treatment:*** *Identifying and handling outliers to prevent skewing results.*
2. *Data Transformation:*
   * ***Normalization and Standardization:*** *Adjusting the scale of data features to bring them within a similar range.*
   * ***Encoding Categorical Variables:*** *Converting categorical data into numerical format using techniques like One-Hot Encoding or Label Encoding.*
3. *Feature Engineering:*
   * ***Creating New Features:*** *Deriving new features from existing data to improve model performance.*
   * ***Feature Selection:*** *Identifying and selecting the most relevant features to use in analysis or modelling.*
4. *Data Integration:*
   * ***Combining Data from Multiple Sources:*** *Merging datasets from various sources into a single, coherent dataset.*
5. *Data Reduction:*
   * ***Dimensionality Reduction:*** *Reducing the number of features while preserving the most important information.*

# Importance of Data Preprocessing:

# *Improves Model Performance****:*** *Ensures that data is in the best possible shape, leading to more accurate and reliable models.*

* *Reduces Noise****:*** *Eliminates irrelevant or redundant information, making patterns in the data more apparent.*
* *Enhances Data Quality****:*** *Ensures that data is clean and consistent, reducing the likelihood of errors during analysis.*
* *Enables Better Insights****:*** *Allows for more meaningful and actionable insights to be derived from the data*.

# WHAT IS FEATURE ENGINEERING?

*Feature engineering is the process of using domain knowledge to create new features (attributes or variables) from raw data that help machine learning algorithms make better predictions. It involves transforming raw data into informative, useful, and relevant inputs for modelling.*

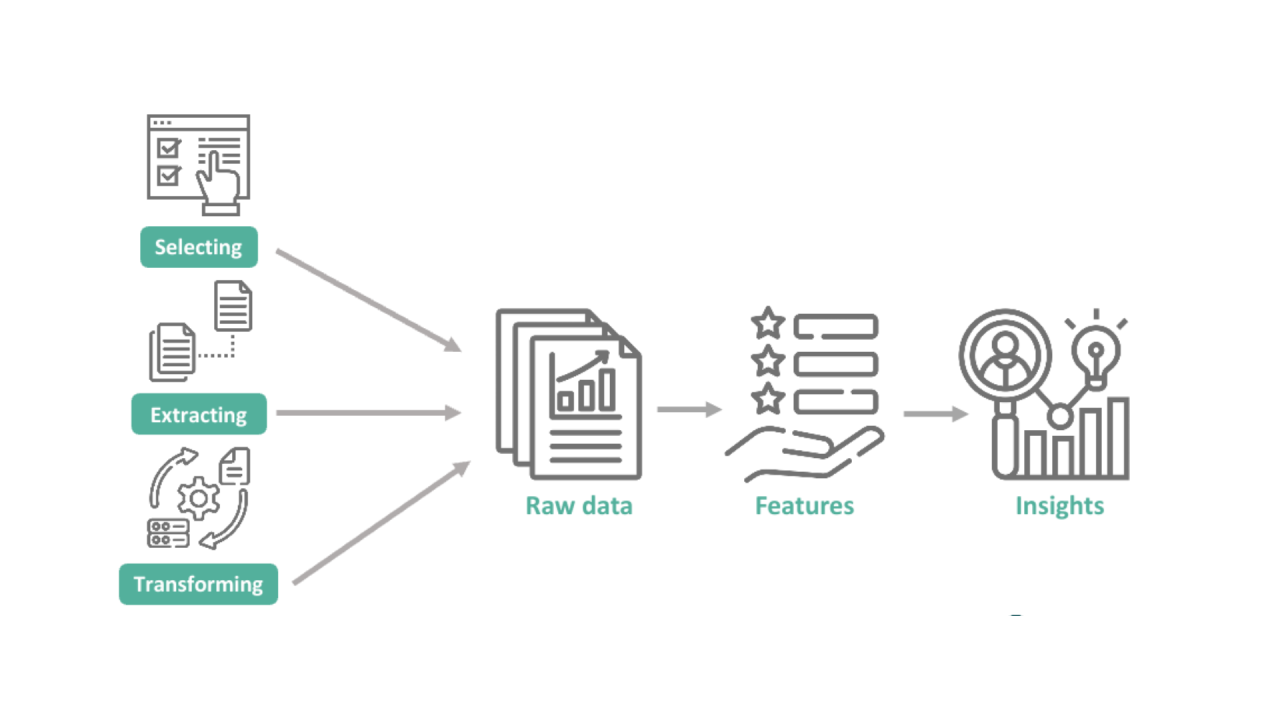


Image .4

Key Steps in Feature Engineering

1. *Creating New Features:*

* ***Date and Time Features:*** *Extracting features like day, month, year, day of the week, or time of day from datetime columns.*
* ***Aggregating Features:*** *Creating new features by aggregating existing ones, such as calculating the mean, sum, or count of*

*values.*

1. *Transforming Existing Features****:***

* ***Scaling and Normalization:*** *Standardizing numerical features to bring them into a common scale.*
* ***Encoding Categorical Variables:*** *Converting categorical data into numerical format using techniques like One-Hot Encoding or Label Encoding.*

1. *Combining Features:*

* ***Polynomial Features:*** *Creating new features by combining existing ones in polynomial forms.*

1. *Decomposing features:*
   * ***Text Features:*** *Extracting useful information from text data, such as word counts or sentiment analysis.*
2. *Handling Time Series Data:*
   * ***Lag Features:*** *Creating features that represent the value of a time series at previous time steps.*
3. *Binning:*
   * ***Discretizing Continuous Variables:*** *Converting continuous variables into categorical bins.*

* Importance of Feature Engineering:
* *Improves Model Performance: Well-engineered features provide better signal to machine learning algorithms, leading to more accurate and reliable predictions.*
* *Reduces Overfitting: Helps in creating more generalizable models by focusing on relevant features.*
* *Simplifies Models: Simplifies complex relationships into more interpretable features, making models easier to understand and explain.*
* *Enhances Insight: Provides deeper insights into the data by highlighting important patterns and relationships.*

# Data INSPECTION:

*Provide a brief overview of the chosen dataset, including the number of samples, features and the target variable.*

* Table of real-world dataset:

*Creating a sample dataset for an "Employee Table" can help illustrate its structure and typical fields. Below is a simplified example of what an Employee Table might look like:*

*And applying data preprocessing and feature engineering techniques on the given employee table*

EMPLOYEE TABLE:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| EMPLOYEE ID | NAME | AGE | DEPARTEMENT | SALARY |
| 101 | VARSHI | 28 | ENGINEER | 60000 |
| 102 | BARATH | 29 | IT | 65000 |
| 103 | NIRSHU | None | SALES | 55000 |
| 104 | BOBBY | 29 | IT | NONE |
| 105 | NIKKI | 27 | HR | 60000 |
| 106 | DEEPU | 28 | SALES | 55000 |

TABLE 1: T

*NOW APPLYING THE DATA PREPROCESSING AND FEATURE ENGINEERING ON THE ABOVE GIVEN DATASET:*

# COMPLETE PYTHON CODE:

import pandas as pd

from sklearn. preprocessing import StandardScaler, LabelEncoder

# Sample Employee Data

data = {

'EmployeeID': [101,102,103,104,105,106],

'Name': ['varshi', 'Barath', 'nirshu', 'bobby', 'Nikki’,’ Deepu'],

'Age': [28, 29, None, 29, 27,28], # Nan represents missing value

'Department': ['engineer', 'IT', 'Sales', 'IT', 'HR','sales'],

'Salary': [60000, 65000, 55000, None, 60000,55000] # NaN represents missing value

}

# Creating DataFrame

df = dataFrame(data)

# Step 1: Handling Missing Values

# Identify missing values

missing values = df. isnull(). sum ()

print ("Missing values:\n", missing\_values)

# Fill missing values in Age with mean

mean\_age = df['Age']. mean()

df['Age']. fillna(mean\_age, inplace=True)

# Fill missing values in Salary with median

median\_salary = df['Salary'].median()

df['Salary'].fillna(median\_salary, inplace=True)

# Step 2: Feature engineering

# Define numerical columns for scaling (Age and Salary)

numerical\_columns = ['Age', 'Salary']

# Standardization

scaler = StandardScaler ()

df[numerical\_columns] = scaler.fit\_transform(df[numerical\_columns])

# Step 3: Handling Categorical Data

# Categorical columns for encoding (Gender and Department)

categorical\_columns = [ 'Department']

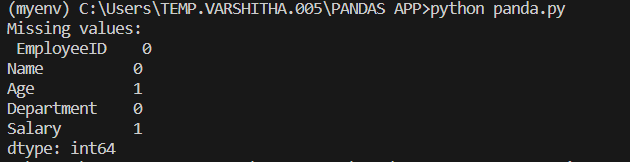
# One-Hot Encoding

df = pd.get\_dummies (df, columns=categorical\_columns)

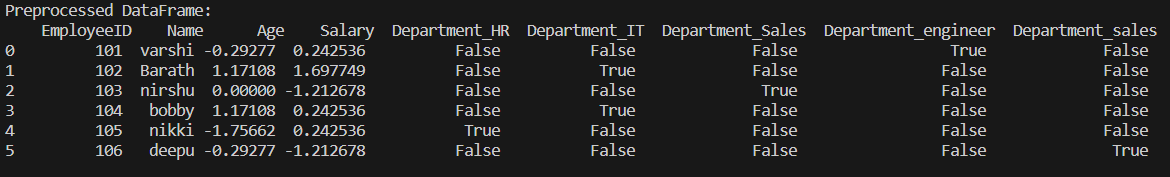
# Display the preprocessed DataFrame



print ("\nPreprocessed DataFrame:\n", df)

OUTPUT:





# EXPLANATION OF CODE:

* *DATA PREPROCESSING AND FEATURE ENGINEERING TECHQUIES HANDLED IN THE CODE ARE:*

1. ***Handling Missing Values****:*
   * *Identify and count the missing values in each column.*
   * *Fill missing values in the ‘****Age’*** *column with the mean age.*
   * *Fill missing values in the ‘****Salary’*** *column with the median salary.*
2. ***Feature Scaling:***

* *Define numerical columns* ***(‘Age’ and ‘Salary’)*** *for scaling.*
* *Standardize the numerical columns using ‘****StandardScaler’*** *to ensure they have a mean of 0 and a standard deviation of 1.*

1. ***Handling Categorical Data****:*

* *Apply One-Hot Encoding to the ‘****Department****’ column to convert categorical data into a numerical format suitable for machine learning models.*
* *Use* ***pd.get\_dummies*** *to perform One-Hot Encoding on the ‘****Department’*** *column, which converts categorical values into binary indicator variables.*
* *The Department column was one-hot encoded, resulting in the creation of several binary columns (Department\_HR, Department\_IT, Department\_Sales, Department\_engineer, Department\_sales), each indicating the presence of a specific department for an employee.*

# ANALYSIS:

*IMPACT OF DATA PREPROCESSING AND FEATURE ENGINEERING ON THE DATASET****:***

*STEP BY STEP PROCESS:*

1. ***Load the dataset****.*
2. ***Perform initial exploratory data analysis (EDA)****:*

* *Summary statistics.*
* *Visualizations (e.g., histograms, scatter plots).*

1. ***Apply data preprocessing techniques****:*

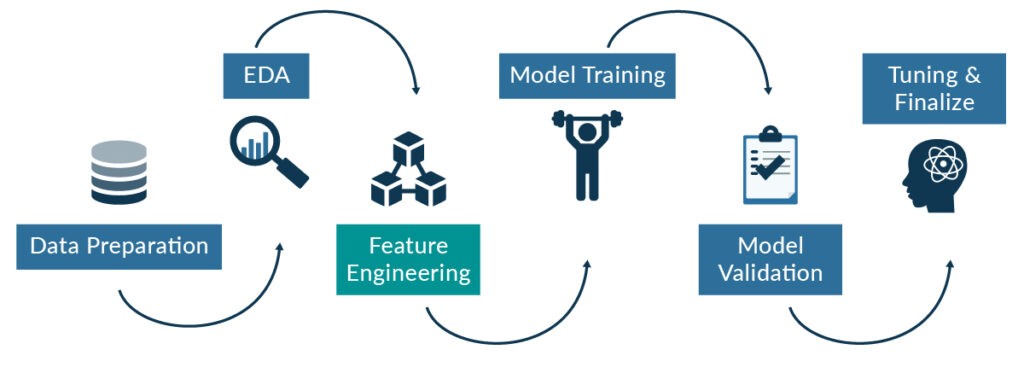
* *Handling missing values.*
* *Scaling/normalizing features.*
* *Encoding categorical variables.*

1. ***Perform feature engineering****:*

* *Creating new features.*
* *Transforming existing features.*

1. ***Perform EDA on the processed dataset****:*

* *Summary statistics.*
* *Visualizations.*

 Image.5

# COMPLETE PYTHON CODE:

*Python code of impact of data preprocessing and feature engineering:*

import pandas as pd

import numpy as np

# Creating a hypothetical dataset

np.random.seed(42)

data = pd.DataFrame({

    'feature1': np.random.normal(100, 20, 1000),

    'feature2': np.random.exponential(1, 1000),

    'category': np.random.choice(['A', 'B', 'C'], 1000),

    'target': np.random.choice([0, 1], 1000)

})

# Introduce some missing values

data.loc[data.sample(frac=0.1).index, 'feature1'] = np.nan

data.loc[data.sample(frac=0.1).index, 'feature2'] = np.nan

data.head()

# Initial summary statistics

initial\_summary = data.describe(include='all')

initial\_summary

import matplotlib.pyplot as plt

import seaborn as sns

# Histograms for numerical features

data.hist(bins=30, figsize=(15, 10))

plt.suptitle('Initial Feature Distributions')

plt.show()

# Pairplot to observe relationships

sns.pairplot(data, hue='target')

plt.suptitle('Initial Feature Relationships')

plt.show()

# Handling missing values by filling with the mean for numerical features

data['feature1'].fillna(data['feature1'].mean(), inplace=True)

data['feature2'].fillna(data['feature2'].mean(), inplace=True)

from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()

numerical\_features = ['feature1', 'feature2']

data[numerical\_features] = scaler.fit\_transform(data[numerical\_features])

data = pd.get\_dummies(data, columns=['category'])

# Creating a new feature as an example

data['feature1\_feature2\_product'] = data['feature1'] \* data['feature2']

# Applying a logarithmic transformation to a skewed feature

data['log\_feature2'] = np.log1p(data['feature2'])

# Summary statistics after preprocessing and feature engineering

processed\_summary = data.describe(include='all')

processed\_summary

#visualizations

# Histograms for numerical features after preprocessing and feature engineering

data.hist(bins=30, figsize=(15, 10))

plt.suptitle('Processed Feature Distributions')

plt.show()

# Pairplot to observe relationships after preprocessing and feature engineering

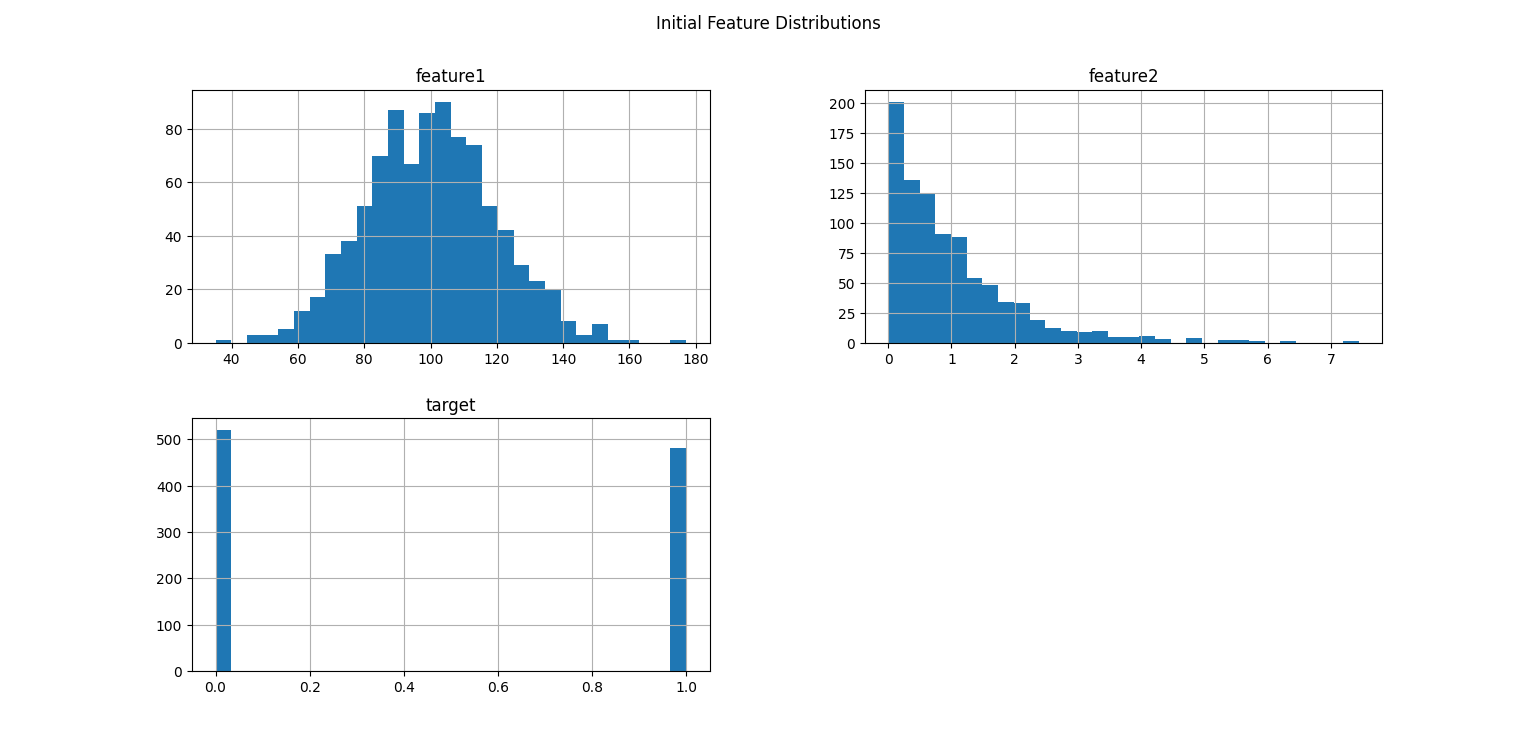
sns.pairplot(data, hue='target')

plt.suptitle('Processed Feature Relationships')

plt.show()

initial\_summary, processed\_summary

# OUTPUT:



# IMPROVEMENTS:

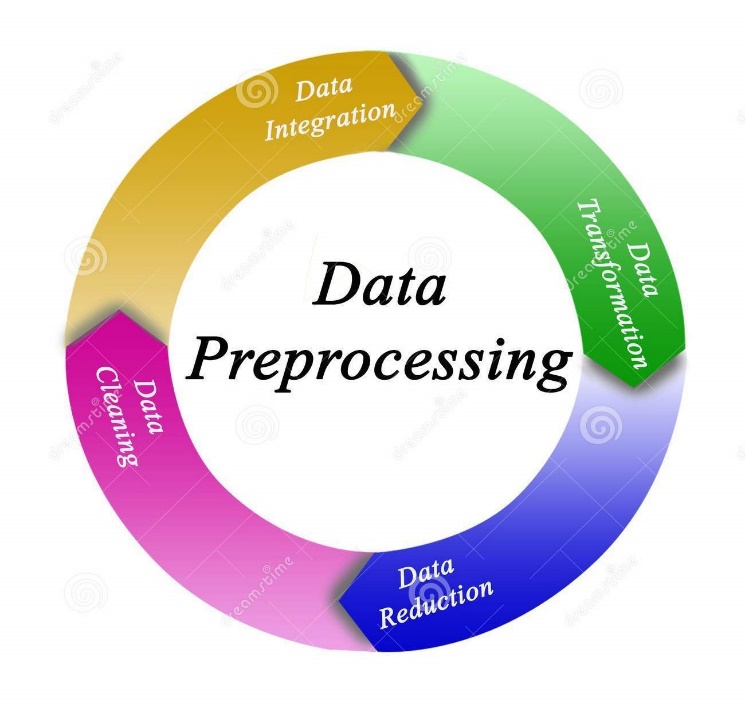
* *Data preprocessing and feature engineering have significantly improved the dataset's suitability for machine learning tasks.*

### Summary of Improvements:

#### Before Preprocessing and Feature Engineering

* ***Data Quality Issues****: Missing values, varying scales, unencoded categorical variables.*
* ***Model Performance****: Potentially lower due to noise, bias, and suboptimal feature representation.*
* ***Feature Relationships****: Not fully captured, leading to missed insights and poor predictive power.*

#### After Preprocessing and Feature Engineering



* ***Enhanced Data Quality****: Complete, scaled, and properly encoded dataset.*
* ***Improved Model Performance****: Better training and generalization due to cleaner, more informative data.*
* ***Richer Feature Set****: New and transformed features that provide deeper insights and improve predictive accuracy.*

### Overall Impact on Machine Learning:

1. ***Data Integrity****: Ensures the dataset is free from errors and inconsistencies, providing a solid foundation for model building.*
2. ***Model Efficiency****: Streamlines the training process, allowing algorithms to converge faster and more effectively.*
3. ***Predictive Accuracy****: Enhances the model’s ability to generalize from training data to unseen data, leading to more reliable predictions.*
4. ***Insight Discovery****: Reveals new patterns and relationships within the data that can inform business decisions and strategies.*

*In summary, data preprocessing and feature engineering are critical to transforming raw data into a form that is suitable for machine learning. They improve data quality, enhance feature representation, and ultimately lead to more accurate and reliable models. Investing time and effort in these steps is crucial for the success of any machine learning project.*

# CONCLUSION:

### *Key Takeaways from Data Preprocessing and Feature Engineering:*

#### **1. Handling Missing Values**

* ***Importance****: Missing values can lead to errors during model training and can bias the model if not handled properly.*
* ***Techniques****: Filling missing values with mean for numerical features and mode for categorical features ensures the dataset is complete and ready for model ingestion.*
* ***Impact****: Improves the robustness and reliability of the model by preventing errors due to null values.*

#### 2. **Scaling/Normalizing Features**

* ***Importance****: Features with different scales can disproportionately influence the model, leading to biased results and slower convergence.*
* ***Techniques****: Standardizing features to have zero mean and unit variance ensures all features contribute equally to the model.*
* ***Impact****: Enhances the performance of gradient-based algorithms and ensures the model treats all features uniformly.*

#### 3. **Encoding Categorical Variables**

* ***Importance****: Machine learning models require numerical input, so categorical variables must be converted to numerical form.*
* ***Techniques****: One-hot encoding converts categorical variables into binary vectors, allowing models to process and interpret them.*
* ***Impact****: Enables the inclusion of categorical data in the model, which can improve predictive accuracy by leveraging more information.*

#### 4. **Creating New Features**

* ***Importance****: New features can capture additional information and relationships within the data that the original features may not fully represent.*
* ***Techniques****: Generating features like the product of two existing features can provide new insights for the model.*
* ***Impact****: Potentially enhances the model's predictive power by introducing new, informative variables.*

#### 5. **Transforming Existing Features**

* ***Importance****: Some features may have skewed distributions that can negatively affect the model’s performance.*
* ***Techniques****: Applying transformations like logarithmic scaling to skewed features reduces skewness and makes patterns more apparent.*
* ***Impact****: Improves model accuracy by normalizing skewed data, making it easier for the model to learn from the data.*

### *6. Summary Statistics Comparison*

* ***Initial Summary Statistics****: Provides an overview of the raw data, showing issues like missing values and varying feature scales.*
* ***Processed Summary Statistics****: Displays the dataset after preprocessing, highlighting improvements such as filled missing values, standardized scales, and encoded categorical variables.*

### *7.Visualizations Comparison*

* ***Initial Visualizations****: Histograms and pairplots of raw data show distributions and relationships before preprocessing, often revealing skewness, missing values, and scale differences.*
* ***Processed Visualizations****: Post-preprocessing histograms and pairplots show more uniform distributions, standardized scales, and additional relationships introduced by new features and transformations…*

***\*\* THE END \*\*\****