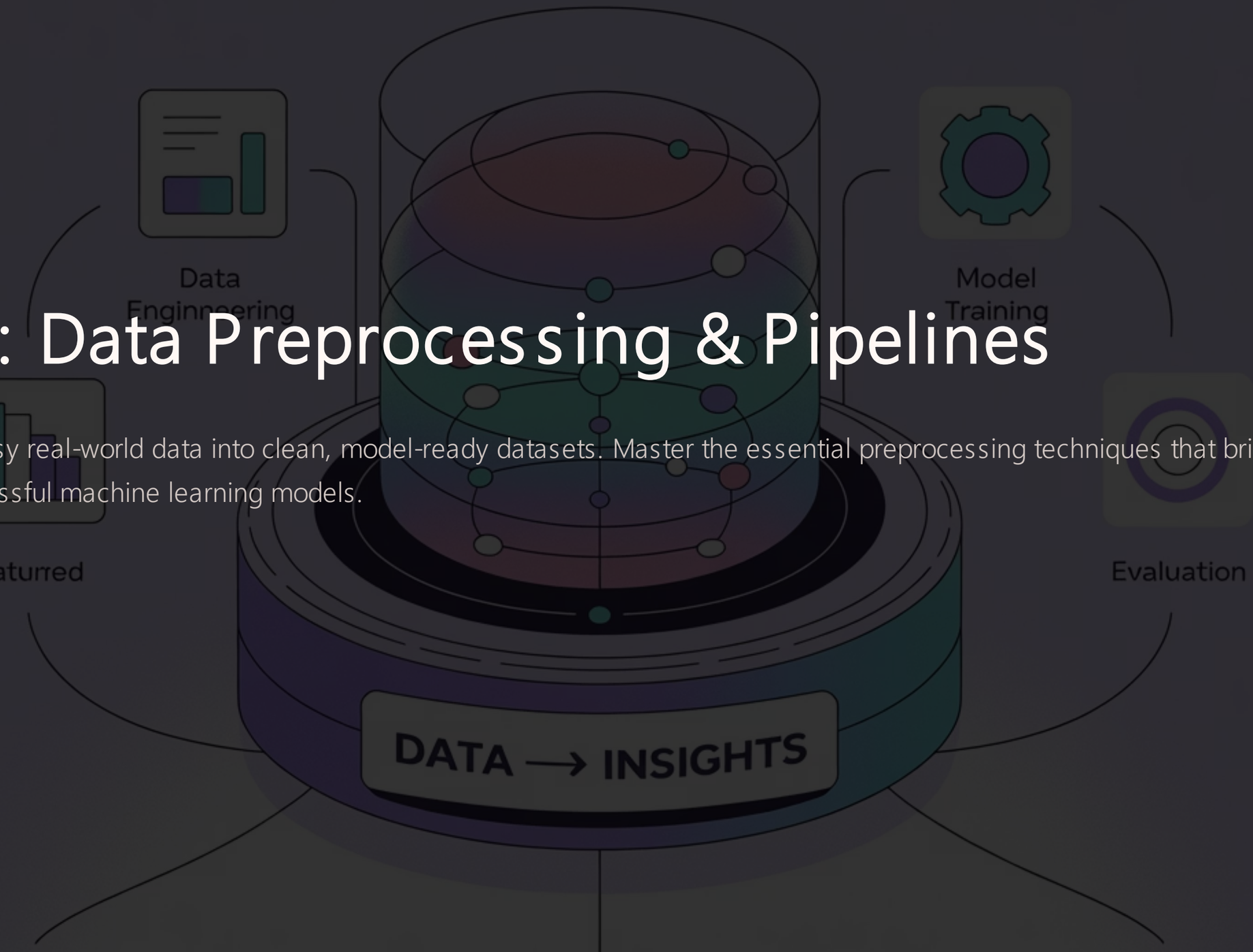
Day 5: Data Preprocessing & Pipelines

Transform messy real-world data into clean, model-ready datasets. Master the essential preprocessing techniques that bridge raw data and successful machine learning models.



Skill Morph
Ignite Your Mind with Intelligent Learning

Why Data Preprocessing Matters

Raw Data Reality

Real-world data is often messy with missing values, noise, inconsistent formats, and outliers that confuse algorithms.

Quality Boost

Preprocessing dramatically improves data quality, making machine learning models more accurate and reliable in predictions.

Study Analogy

Think of preprocessing as cleaning and organizing your study notes before an important exam - preparation leads to better results.

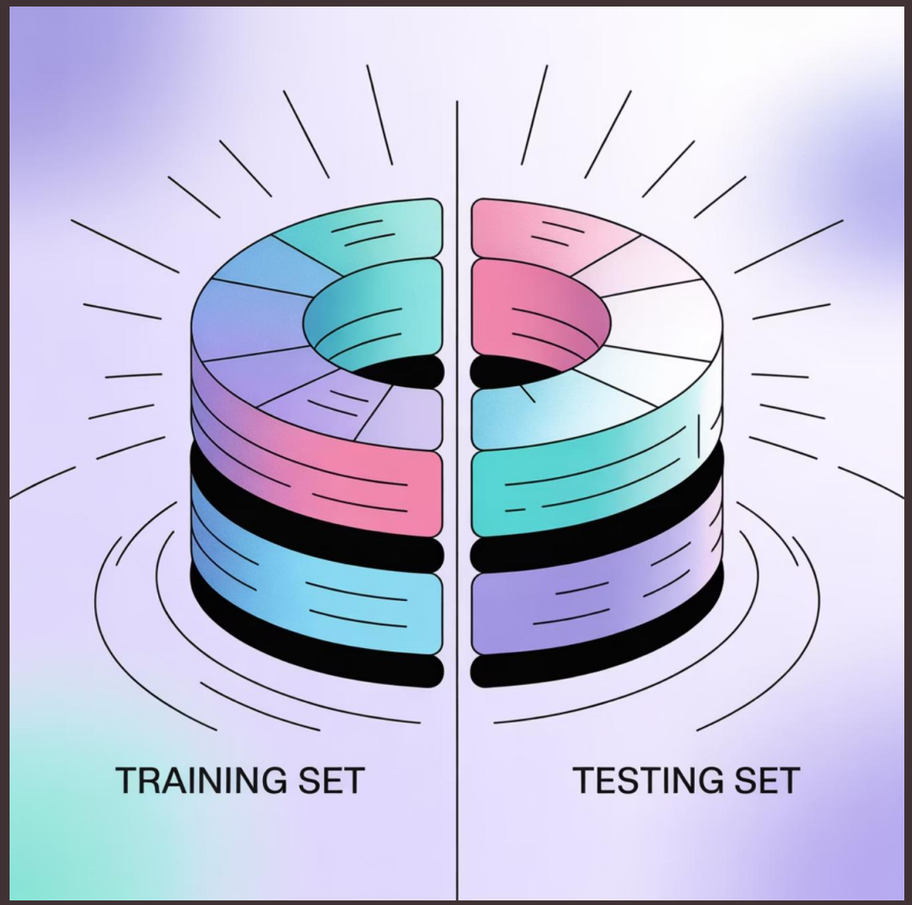
Train-Test Split: Preparing Data for Learning

The Fundamental Split

Split your dataset into two parts: training data (to learn patterns) and testing data (to evaluate performance objectively.

Common split ratio is 80% training and 20% testing. This prevents "cheating" by testing on data the model has already seen during training.

```
train_test_split(X, y, test_size=0.2, random_state=42)
```



Training Set (80%)

Used to teach the model patterns and relationships in the data

Testing Set (20%)

Used to evaluate how well the model performs on unseen data

Cross-Validation: Reliable Model Evaluation

01

Divide into Folds

Split data into multiple folds (typically 5 or 10 equal parts) instead of just one train-test split.

02

Rotate Testing

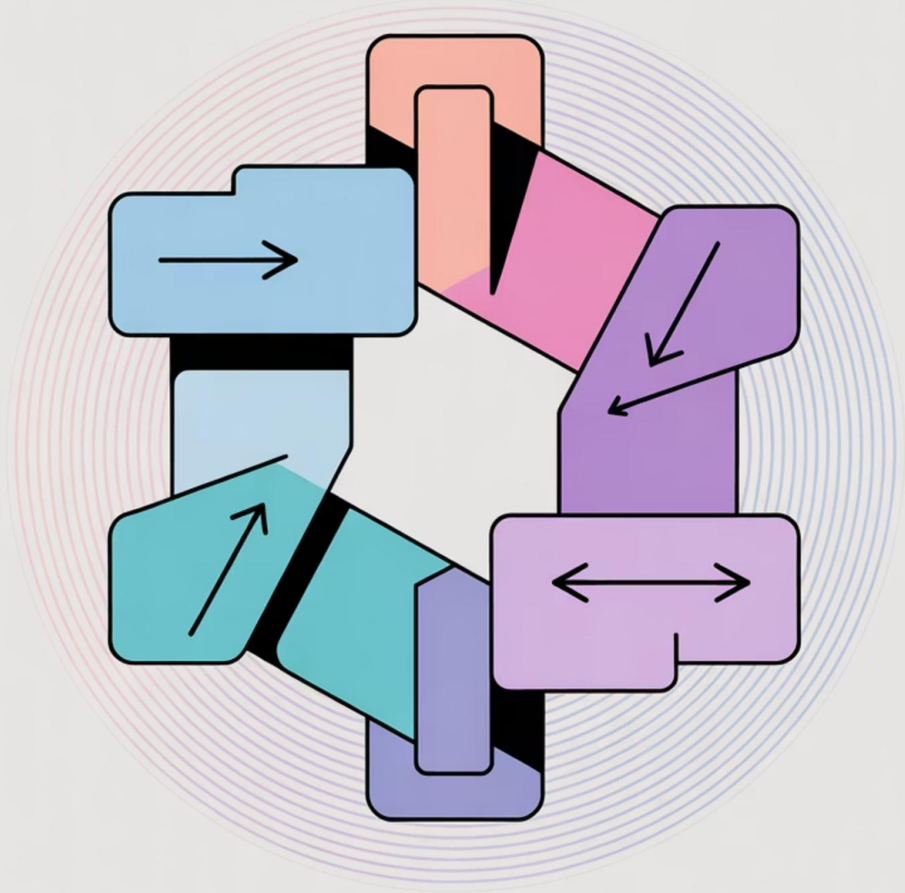
Train on some folds, test on remaining fold. Repeat process for all fold combinations systematically.

03

Average Results

Provides better, more reliable estimate of model performance on truly unseen data across multiple iterations.

Cross-validation reduces the risk of getting lucky or unlucky with a single train-test split, giving you confidence in your model's true performance.



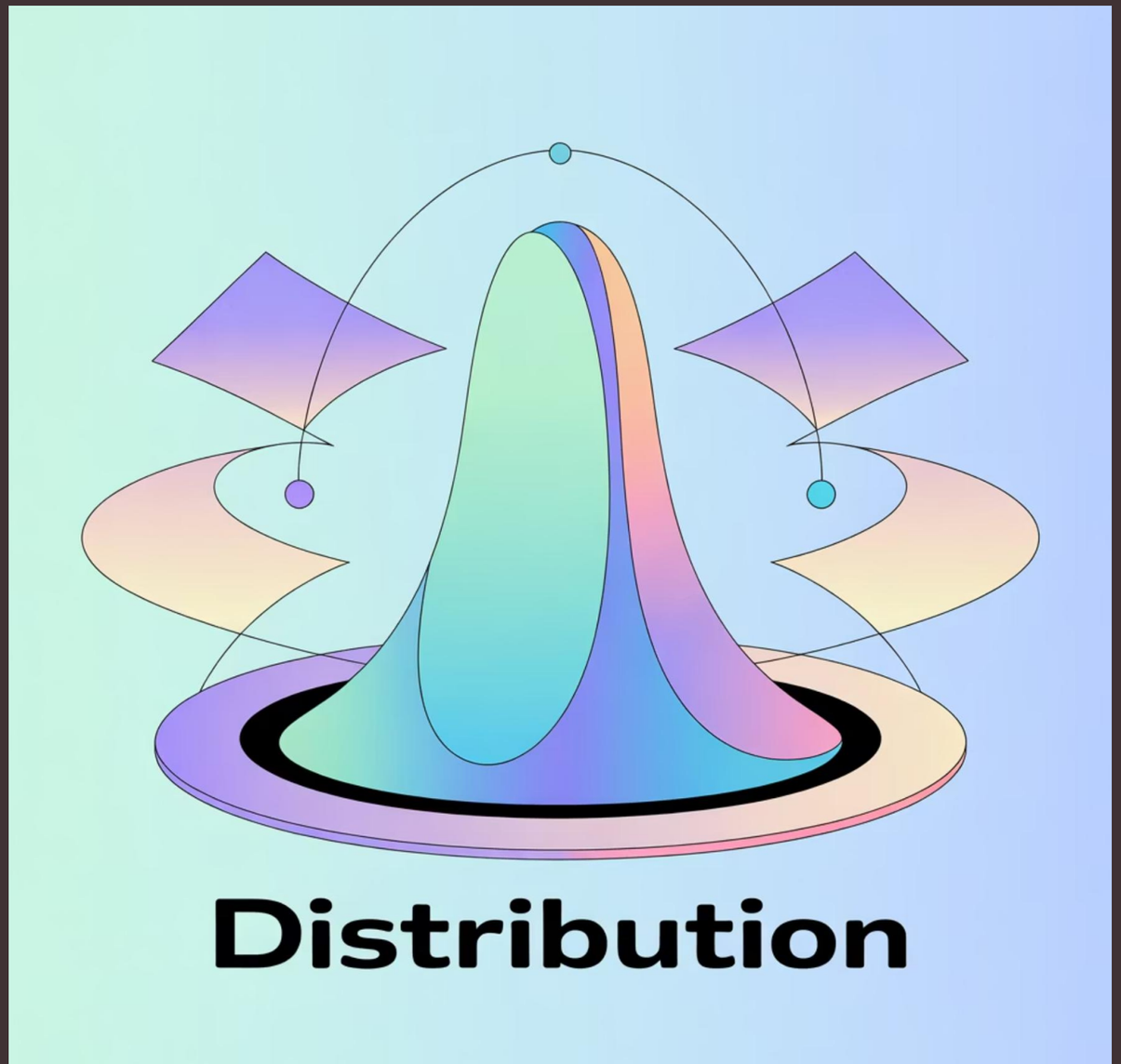
K-Fold Cross-validation

Skill Morph
Ignite Your Mind with Intelligent Learning

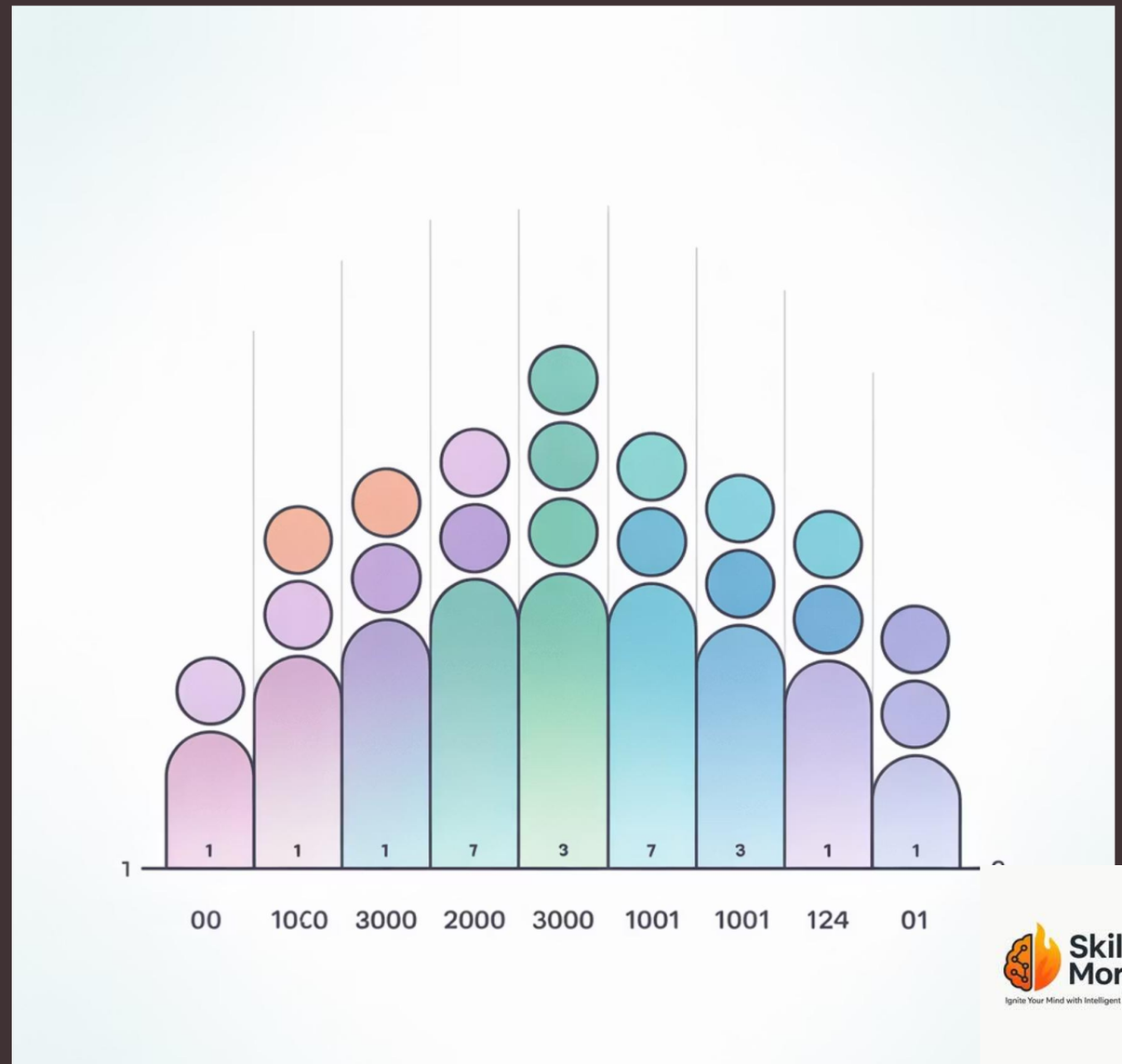
Scaling Data: Making Features Comparable

Features often have different units and ranges (age vs. income). Scaling transforms features to a common scale without distorting the underlying relationships between data points.

StandardScaler



MinMaxScaler



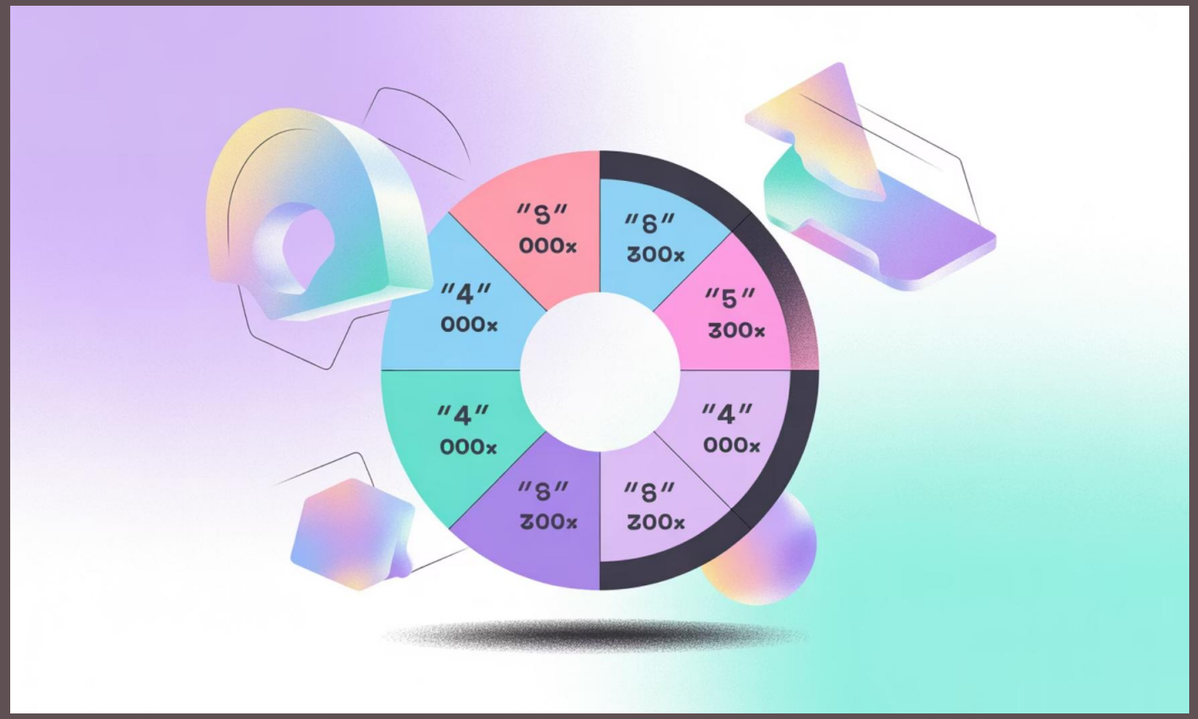
Skill Morph
Ignite Your Mind with Intelligent Learning

Encoding Categorical Data: Numbers from Words

Machine learning models need numerical input, but real data often contains categories like colors, countries, or product types. Encoding converts text categories into numbers.

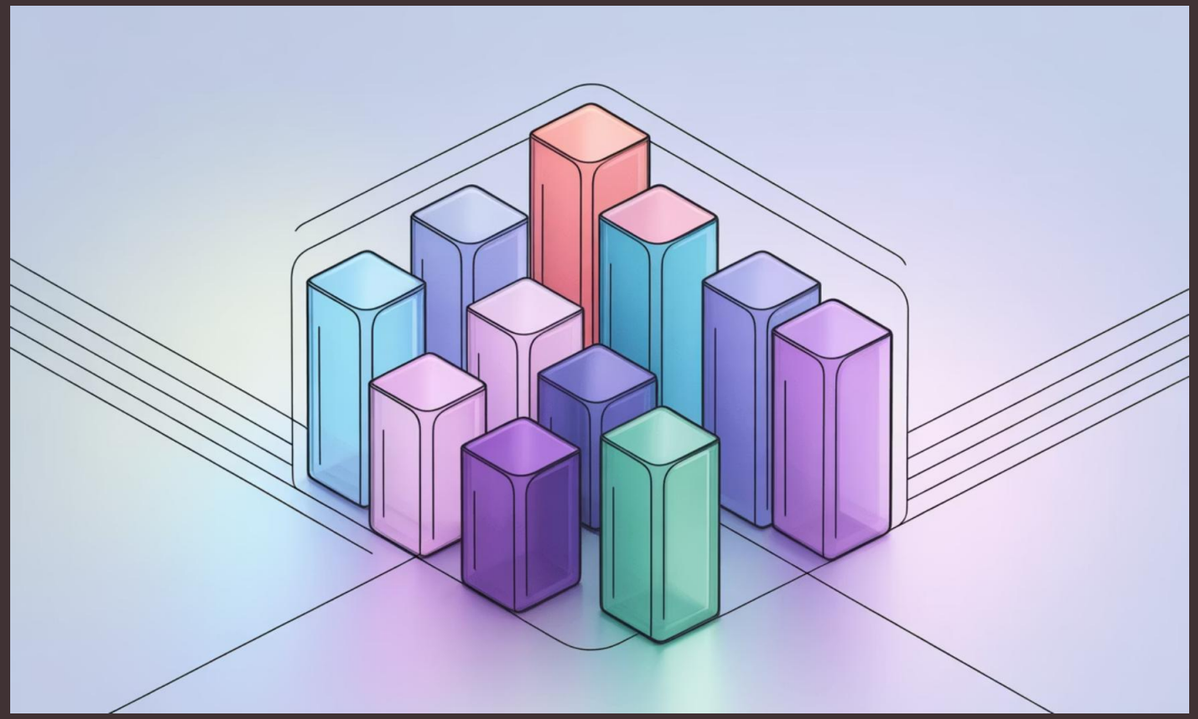
Label Encoding

Assigns each unique category a number (red=0, blue=1, green=2). Simple but implies order that may not exist.



One-Hot Encoding

Creates binary columns for each category (red: 1/0, blue: 1/0). Avoids implying false order between categories.



Skill Morph
Ignite Your Mind with Intelligent Learning

Building Pipelines: Automating Preprocessing

1

Chain Steps

Pipelines connect multiple preprocessing operations into one automated workflow sequence.

2

Ensure Consistency

Guarantees same preprocessing steps applied to training, validation, and new data consistently.

3

Reduce Errors

Simplifies code, minimizes human mistakes, and makes experiments easily reproducible across projects.

Pipeline Example: Raw Data → Missing Value Handling → Scaling → Encoding → Model Training

Pipelines are your best friend for maintaining clean, professional, and error-free machine learning workflows that you can trust and reuse.