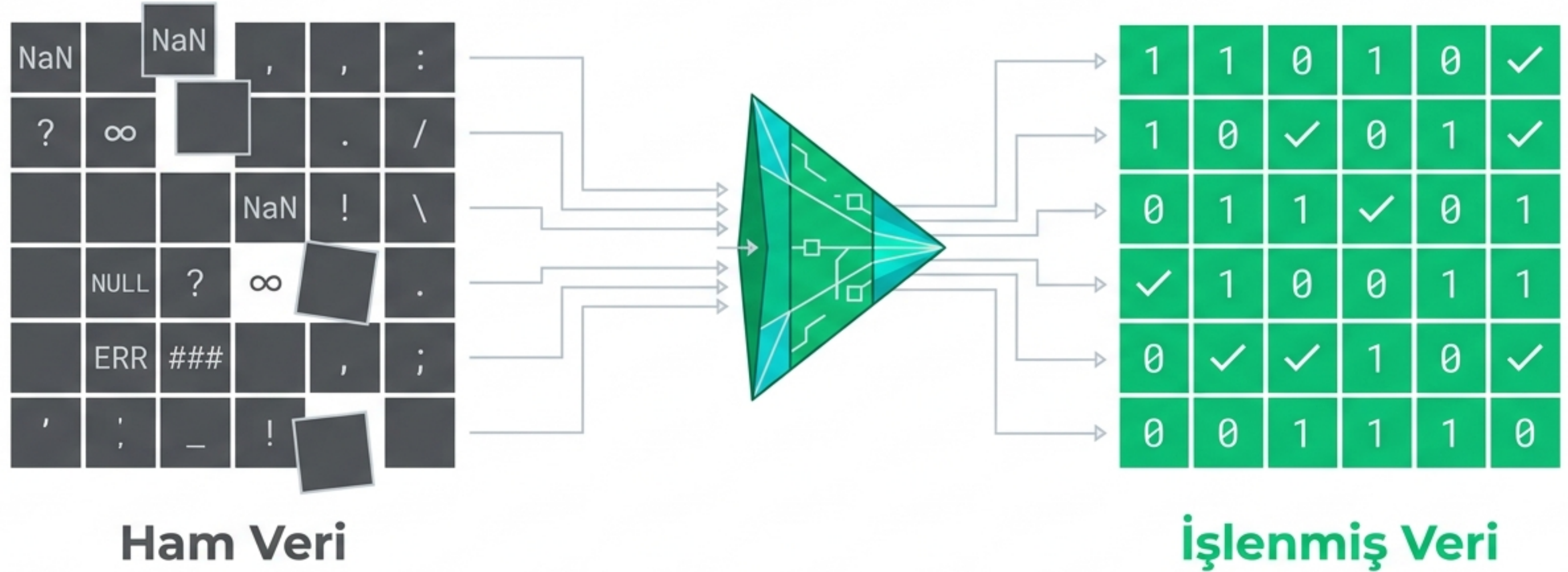


Büyük Veri Ön İşleme (Data Preprocessing)

Makine Öğrenmesinin Görünmez Kahramanı



Garbage In, Garbage Out (GIGO)

Zamanın **%80'i:**
Veri Temizleme ve
Ön İşleme

Zamanın **%20'si:**
Model Kurma ve
Optimizasyon



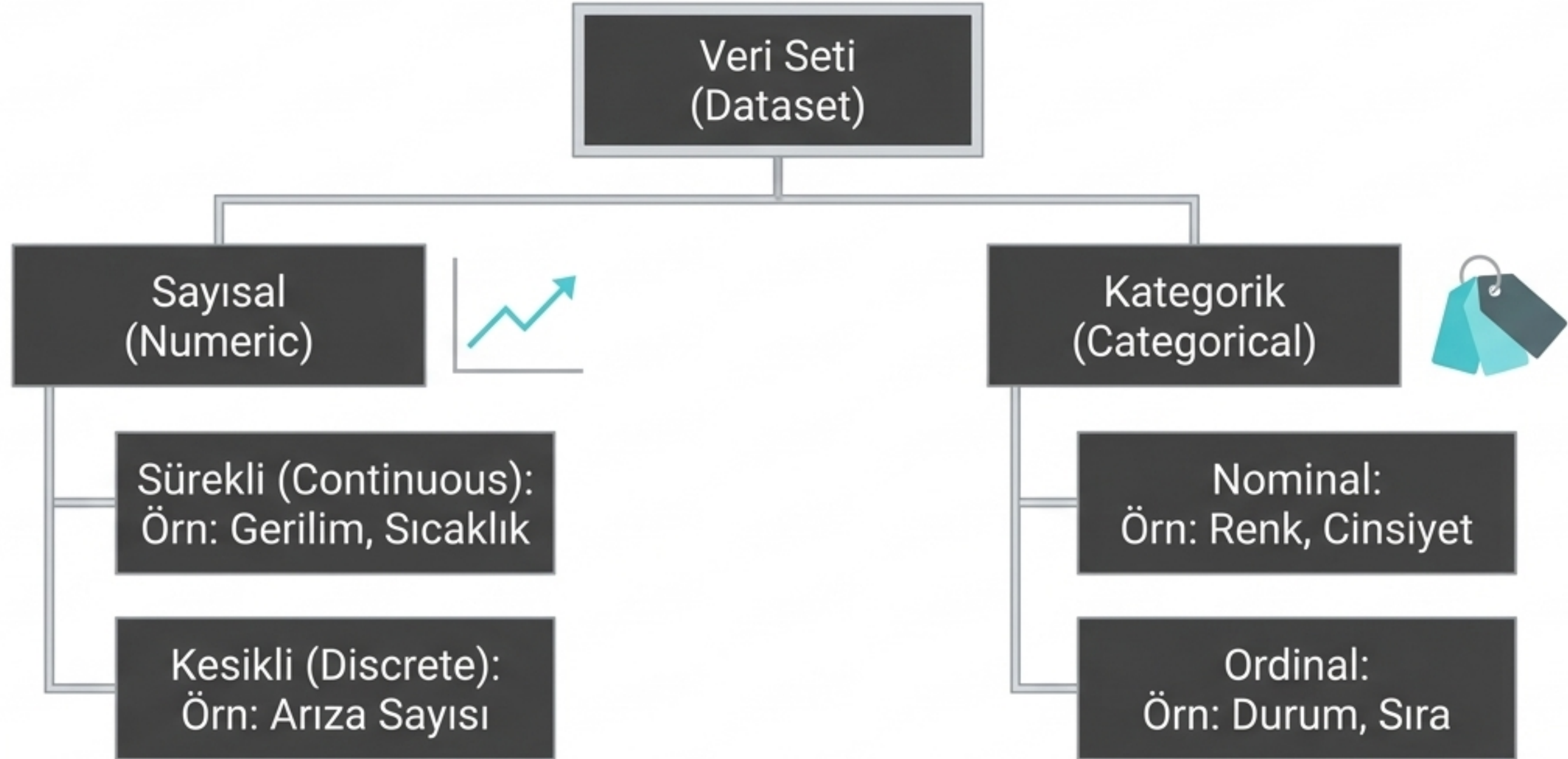
Bozuk veya kirli veriyle kurulan hiçbir karmaşık algoritma, doğru sonuçlar üretemez. Yüksek kaliteli bilgi, yüksek kaliteli veriden doğar.

Veri Bilimi Yaşam Döngüsü



Makine öğrenmesi algoritmalarının veriyle buluştuğu köprü. Ön işleme aşaması, verinin "ham" halinden "model için hazır" haline geçişidir.

Veri Anatomisi: Değişkenleri Tanımak



Algoritmalar yalnızca matematiksel dilden anlar. Nihai hedefimiz, veri setindeki her şeyi mantıksal bir sayısal formata dönüştürmektir.

Veri Rafinerisinin 4 Temel Adımı

1. Temizleme (Cleaning)



Eksik veriler ve aykırı değerlerle başa çıkma stratejileri.

2. Dönüştürme (Transformation)



Veri ölçeklendirme ve kategorik verileri sayısallaştırma (Encoding).

3. İndirgeme (Reduction)



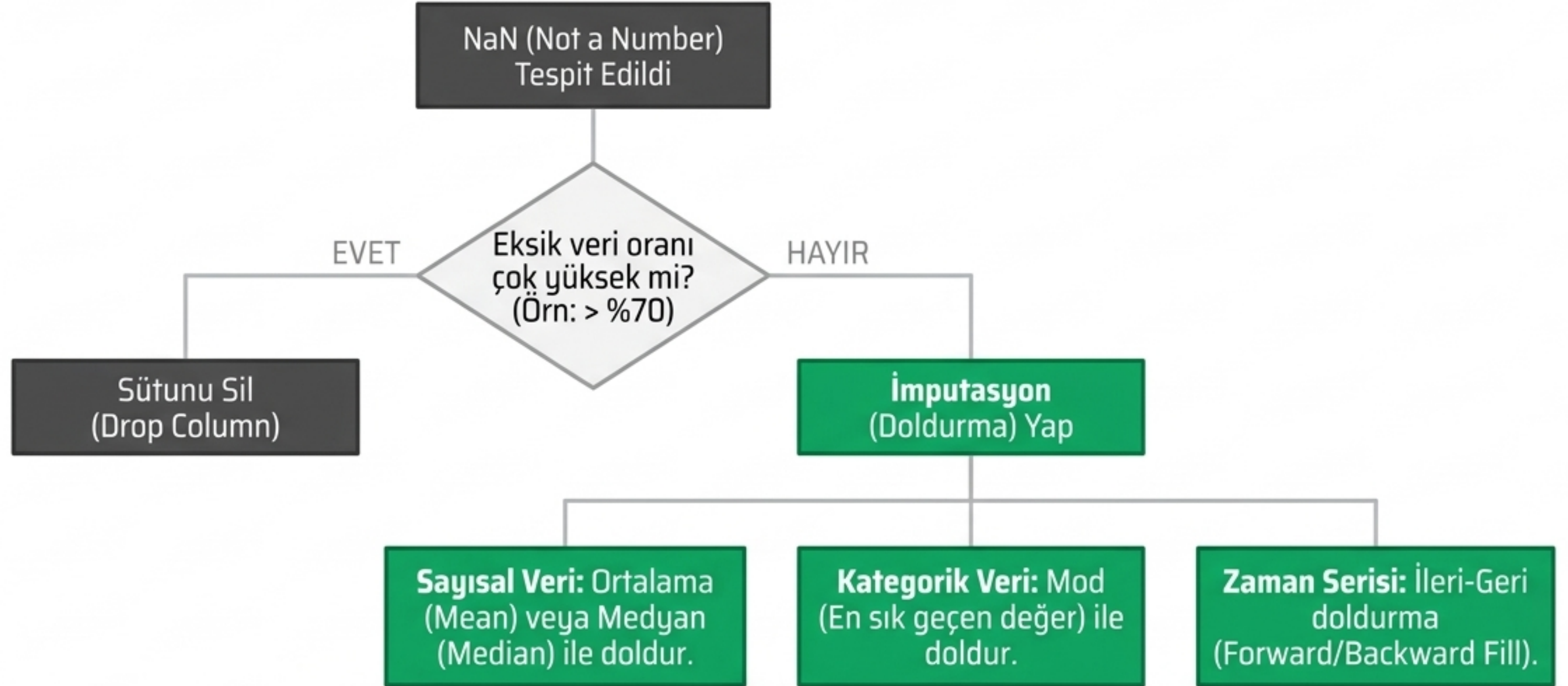
Özellik seçimi ve veri boyutunu (sütun sayısını) azaltma.

4. Bölme (Splitting)



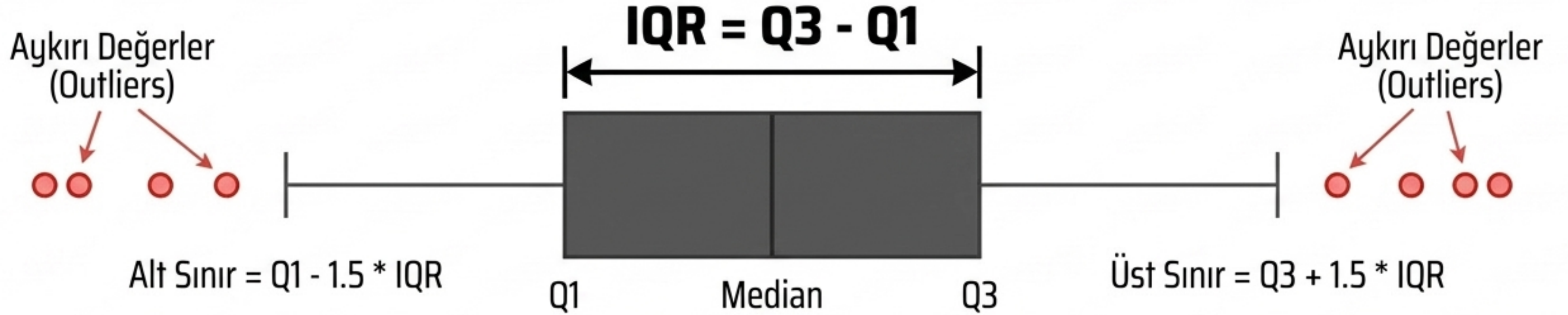
Veriyi Eğitim (Train), Doğrulama (Validation) ve Test setlerine ayırma.

Adım 1: Eksik Verilerle Başa Çıkmak (Missing Data)



Uyarı: Silme işlemi değerli bilgilerin kaybolmasına yol açabilir. Çoğu durumda istatistiksel doldurma (Imputation) tercih edilmelidir.

Adım 2: Aykırı Değerler (Outlier Detection - IQR)



Strategic Action Panel: Aykırı Değer Bulununca Ne Yapılır?

1. Silme (Drop):

Eğer değer açıkça bir ölçüm hatasıysa.

2. Baskılama (Clipping/Capping):

Değerleri alt ve üst sınırlara eşitleme.

3. Koruma:

Eğer gerçek ve önemli bir anomaliyse (örn: kredi kartı dolandırıcılığı).

Adım 3: Kategorik Verileri Kodlama (Encoding)

Label Encoding

Kırmızı ->	0
Mavi ->	1
Yeşil ->	2



Tehlike: Algoritma $2 > 1 > 0$ şeklinde yapay bir büyüklük hiyerarşisi algılayabilir. Nominal veriler (sırasız) için risklidir.

One-Hot Encoding (Dummy Variables)

Kırmızı ->	[1, 0, 0]
Mavi ->	[0, 1, 0]
Yeşil ->	[0, 0, 1]



Avantaj: Herhangi bir hiyerarşi yaratmaz.
Dezavantaj: Veri setinin boyutunu (sütun sayısını) genişletir.

Adım 4: Veri Ölçeklendirme (Feature Scaling)

Min-Max Normalizasyon

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

- Veriyi kesin olarak 0 ile 1 aralığına sıkıştırır.
- Aykırı değerlere (outliers) karşı son derece hassastır.

En uygun kullanım: Sinir Ağları (Neural Networks), Görüntü İşleme.

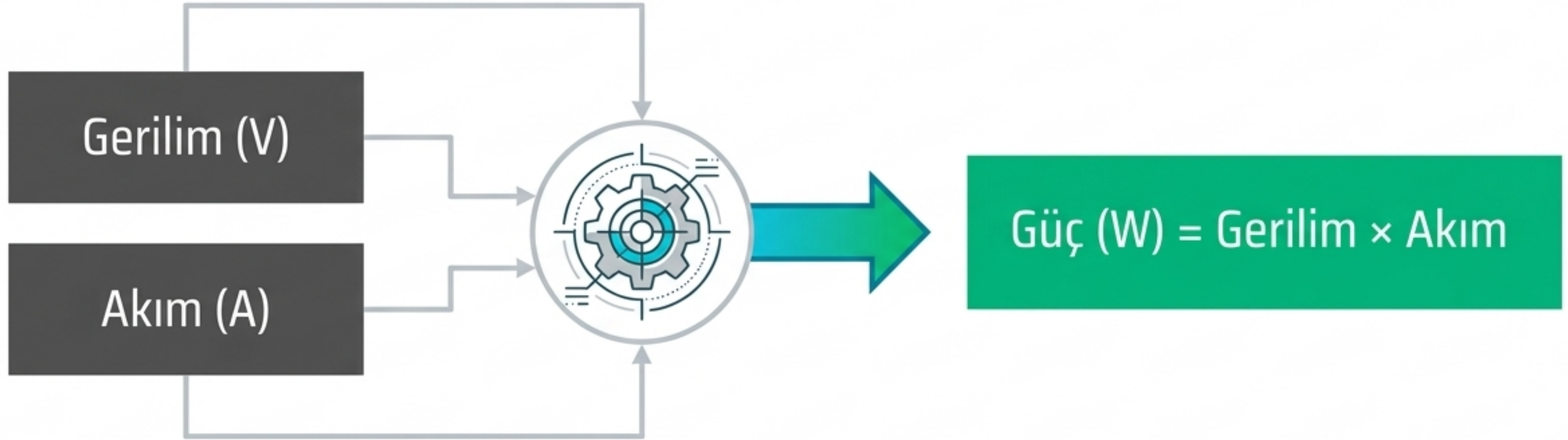
Z-Score Standardizasyon

$$z = \frac{x - \mu}{\sigma}$$

- Ortalama (Mean) = 0, Standart Sapma = 1 olacak şekilde dağılımı ayarlar.
- Belirli bir minimum-maksimum aralığı yoktur; aykırı değerlere karşı daha dirençlidir.

En uygun kullanım: PCA, Lojistik Regresyon, Doğrusal Modeller.

Adım 5: Özellik Mühendisliği (Feature Engineering)



Modele veriden daha fazla anlam çıkarttırma sanatı. Alan bilgisi (Domain Knowledge) kullanılarak ham özelliklerin matematiksel kombinasyonlarından, model için çok daha kritik olan yeni sentez özellikler (Features) üretilir.

Kritik Hata: Veri Sızıntısı (Data Leakage)

YANLIŞ YAKLAŞIM



Test verisinin istatistiksel bilgisi eğitim sürecine sızdı! Modelin başarısı sahte (over-optimistic) olur.

DOĞRU YAKLAŞIM



Ölçeklendirme parametreleri (Ortalama, Min-Max) **YALNIZCA** Eğitim verisinden öğrenilmelidir.

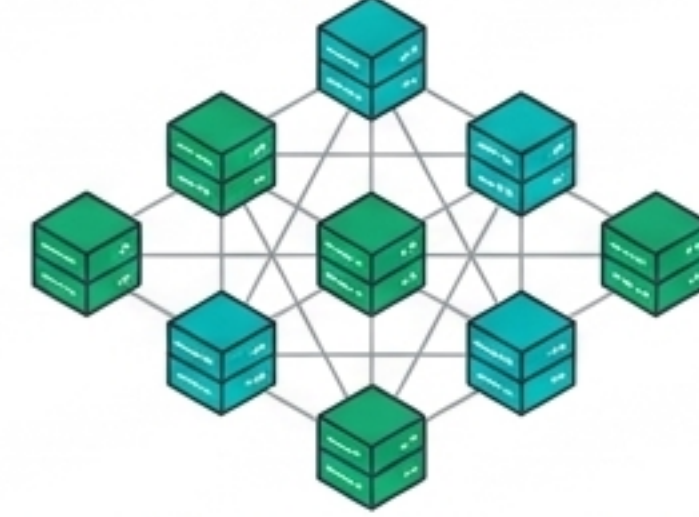
Asla tüm veri seti üzerinde işlem yapmayın. Önce bölün, sonra dönüştürün.

Büyük Veri (Big Data) Bağlamında Ön İşleme



Klasik Veri (Single Node)

- **Hafıza:** RAM kapasitesi ile sınırlı.
- **Yürütme:** Tek bir makine üzerinde (Pandas, Scikit-Learn).
- **Kapsam:** Küçük ve orta ölçekli veri setleri.



Büyük Veri (Distributed Systems)

- **Hafıza:** Dağıtık (Distributed) ve yatayda ölçeklenebilir.
- **Yürütme:** MapReduce mantığı ve RDD/DataFrame kullanımı.
- **Araçlar:** Apache Spark (MLlib), Hadoop.

Kavramlar ve matematik (Temizleme, Normalizasyon, Encoding) aynı kalır; ancak kullanılan mimari yatayda ölçeklenebilir (Scale-out) sistemlere dönüşür.

Modern Araç Seti: Python Ekosistemi

Scikit-Learn:

Dönüştürme, kodlama (OneHotEncoder) ve makine öğrenmesi boru hatları (Pipelines).

```
scaler = MinMaxScaler()  
X_train_scaled = scaler.fit_transform(X_train)  
X_test_scaled = scaler.transform(X_test)
```

Pandas: Veri manipülasyonu, eksik değer kontrolü ve EDA işlemleri.

Sentez: Uçtan Uca Boru Hattı (The Master Pipeline)



Scikit-Learn 'Pipeline' Mimarisi: Tüm bu adımlar, otomasyon, tekrarlanabilirlik ve veri ell sızıntısını önlemek için tek bini. Tüm bu adımlar, otomasyon, tekrarlanabilirlik ve veri sızıntısını önlemek için tek bir kod bloğu (ColumnTransformer / Pipeline) halinde zincirlenir.

Özet ve Temel Çıkarımlar



Kaliteli Veri > Karmaşık Model: En ileri düzey yapay zeka algoritması bile, iyi temizlenmemiş ve rafine edilmemiş bir veri setiyle başarısız olmaya mahkumdur.



Sızıntıya İzin Vermeyin: Veri ön işleme zincirinde 'fit' (öğrenme) işlemini her zaman ve SADECE Eğitim (Train) verisi üzerinde yapın.



Alan Bilgisi (Domain Knowledge) Şarttır: Aykırı değerleri silmek veya yeni özellikler üretmek (Feature Engineering) sadece istatistiksel bir işlem değil, verinin geldiği sektörü anlama sanatıdır.