Standardizace a normalizace dat

Feature Scaling (úprava měřítka) je proces sjednocení rozsahu hodnot všech vstupních proměnných. Většina algoritmů strojového učení (zejména ty využívající metriky vzdálenosti jako k-NN, SVM nebo PCA) vyžaduje, aby data byla ve stejném měřítku, jinak budou proměnné s většími čísly neoprávněně dominovat modelu.

1. Normalizace (Min-Max Scaling)

Normalizace mění měřítko dat tak, aby všechny hodnoty ležely v pevném intervalu, obvykle **<0, 1>** (nebo ←1, 1>).

Vzorec: $$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

- **Kdy použít:** Pokud víte, že data mají jasně definované hranice a neobsahují extrémní odlehlé hodnoty (outliers). Často se používá u neuronových sítí a při zpracování obrazu (pixely 0–255 se mění na 0–1).
- **Nevýhoda:** Pokud se v datech objeví jeden extrémně vysoký bod, zbytek dat se „smrskne“ do velmi malého prostoru blízko nuly.

2. Standardizace (Z-score Normalization)

Standardizace transformuje data tak, aby měla **průměr 0** a **směrodatnou odchylku 1**. Výsledné hodnoty nejsou omezeny konkrétním rozsahem.

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

Kde:

- **μ** je průměr (mean).
- **σ** je směrodatná odchylka (standard deviation).

- **Kdy použít:** Je robustnější vůči odlehlým hodnotám a je nezbytná pro algoritmy jako PCA, lineární regrese nebo logistická regrese, které předpokládají, že data jsou soustředěna kolem nuly.
- **Vlastnost:** Většina hodnot (cca 99 %) se po standardizaci bude nacházet v intervalu ←3, 3>.

3. Srovnání: Kterou metodu zvolit?

Metoda	Rozsah výstupu	Citlivost na outliers	Typické využití
Normalizace	Fixní (0 až 1)	Velmi vysoká	Image processing, NN
Standardizace	Neomezený	Nižší	PCA, Regrese, SVM

4. Praktické tipy

- **Vždy po rozdělení dat:** Scaling provádějte až **po** rozdělení dat na trénovací a testovací sadu. Parametry (min, max, průměr) vypočítejte na trénovacích datech a ty pak aplikujte na testovací data. Zabráníte tím „úniku informací“ (Data Leakage).
- **Algoritmy, které scaling nepotřebují:** Rozhodovací stromy a jejich varianty (**Random Forest**, **XGBoost**) jsou k měřítku dat imunní, protože pracují s prahovými hodnotami (vytvářejí uzly typu „je hodnota > 50?“).

Ukázka v Pythonu (Scikit-Learn)

```
from sklearn.preprocessing import StandardScaler, MinMaxScaler

# Inicializace
scaler_std = StandardScaler()
scaler_norm = MinMaxScaler()

# Aplikace standardizace
X_train_std = scaler_std.fit_transform(X_train)
X_test_std = scaler_std.transform(X_test)

# Aplikace normalizace
X_train_norm = scaler_norm.fit_transform(X_train)
```

Související články:

- Metriky vzdálenosti
- Redukce dimenzionality (PCA)
- Jak pracovat s odlehlými hodnotami

Tagy: ml preprocessing statistika scaling normalization