📆 DAY 18 — Regularization (Ridge, Lasso, ElasticNet)

Goal : Prevent overfitting and improve generalization

1️⃣ Why Regularization Exists

The Problem: Overfitting

A model can:

- Fit training data extremely well
- Learn noise instead of signal
- Fail on unseen data

This happens when:

- Too many features
- Multicollinearity
- Small datasets

2️⃣ What is Regularization? (CORE IDEA)

Regularization = Penalizing model complexity

Instead of minimizing just error:

$$Loss = Error + Penalty$$

This penalty:

- Shrinks coefficients
- Prevents extreme values
- Forces simplicity

> Simple models generalize better.

3️⃣ L2 Regularization — Ridge Regression

Formula (Intuition)

$$\text{Loss} = \text{MSE} + \lambda \sum w^2$$

What it does:

- Penalizes **Large weights**
- Shrinks them toward zero
- Keeps all features

When to use:

- Many correlated features
- You want stability

4️⃣ L1 Regularization — Lasso Regression

Formula (Intution)

$$\text{Loss} = \text{MSE} + \lambda + \sum |w|$$

What it does:

- Pushes some weights to **exactly zero**
- Performs **feature selection**

When to use:

- Many features
- You want simpler models

5️⃣ ElasticNet (Best of Both Worlds)

Combines:

- L1 (feature selection)
- L2 (stability)

When to use:

- Many correlated features
- You want robustness + sparsity

Default choice in real-world ML.

6️⃣ The Regularization Strength (λ / alpha)

Key parameter: `alpha`

Value	Effect
Small alpha	Behaves like normal regression
Large alpha	Very simple model

⚠️ Too large → underfitting

⚠️ Too small → overfitting

7️⃣ Why Scaling is REQUIRED

Regularization depends on **weight magnitude**.

❌ Unscaled features → unfair penalty

✅ Always scale features before regularization

8️⃣ Visual Intuition

Method	Effect
Linear	Big weights
Ridge	Small weights
Lasso	Zero weights

9️⃣ Implement Regularized Models (sklearn)

```
from sklearn.liner_model import Ridge, Lasso, ElasticNet
```
