LASSO Regularization and Cross-Validation

Group 1

Setup

The LASSO estimator solves

$$\hat{\boldsymbol{\beta}}_\lambda \;=\; \arg\min_{\boldsymbol{\beta}\in\mathbb{R}^p} \left\{ \frac{1}{2n} \, \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 \;+\; \lambda \, \|\boldsymbol{\beta}\|_1 \right\}, \qquad \lambda \geq 0.$$

The tuning parameter λ controls the trade-off between fidelity to the training data (first term) and sparsity/shrinkage (second term).

Comments (3 points)

(1 point) Effect of very large λ vs. very small λ

Coefficients.

- **Very large λ:** The ℓ_1 penalty dominates, producing *exact zeros* for many coordinates (strong sparsity). Surviving coefficients are heavily shrunk in magnitude. In the limit $\lambda \to \infty$, $\hat{\boldsymbol{\beta}}_\lambda \to \mathbf{0}$.
- **Very small λ:** The penalty is negligible; shrinkage vanishes and few (if any) coefficients are zero. As $\lambda \downarrow 0$, $\hat{\boldsymbol{\beta}}_\lambda \to \hat{\boldsymbol{\beta}}_{\text{OLS}}$ whenever OLS is defined (and to a minimum-norm least-squares solution in underdetermined cases).

Training vs. test error.

- **Training error:** *Increases* monotonically as λ grows (the model becomes simpler and underfits); *decreases* as λ shrinks (the model fits the training data more closely).
- **Test error:** Typically exhibits a *U-shaped* curve as a function of λ (often plotted against $\log \lambda$): starting from $\lambda \approx 0$, a small increase in λ reduces variance and *lowers* test error (mitigating overfitting); beyond an optimal $\lambda^\star$ the model underfits and test error *rises*.

(2 points) What cross-validation is, why it is useful, and a sketch

Definition. k-fold cross-validation (CV) estimates out-of-sample performance by partitioning the dataset into k disjoint folds $\mathcal{D}_1, \ldots, \mathcal{D}_k$. For $i = 1, \ldots, k$, train the model on $\mathcal{D} \setminus \mathcal{D}_i$ and evaluate its loss on the held-out fold $\mathcal{D}_i$ to obtain L_i. The estimated predictive risk is

$$\text{CV}_k \;=\; \frac{1}{k} \sum_{i=1}^{k} L_i.$$

For LASSO, select the regularization via

$$\lambda^\star \;=\; \arg\min_{\lambda\in\Lambda} \; \text{CV}_k(\lambda),$$

optionally applying the "one-standard-error" rule to prefer a sparser model whose CV error is within one standard error of the minimum.

Why it is useful (machine learning perspective).

1. **Hyperparameter/model selection.** Provides an unbiased, data-driven criterion for choosing λ and comparing models.

2. **Controls overfitting.** Rewards models that generalize to unseen data; penalizes those that only memorize the training set.
3. **Stability and efficiency.** Averaging across folds reduces variance relative to a single split and ensures all observations serve as both training and validation across the k iterations.

Cross-Validation and λ Selection

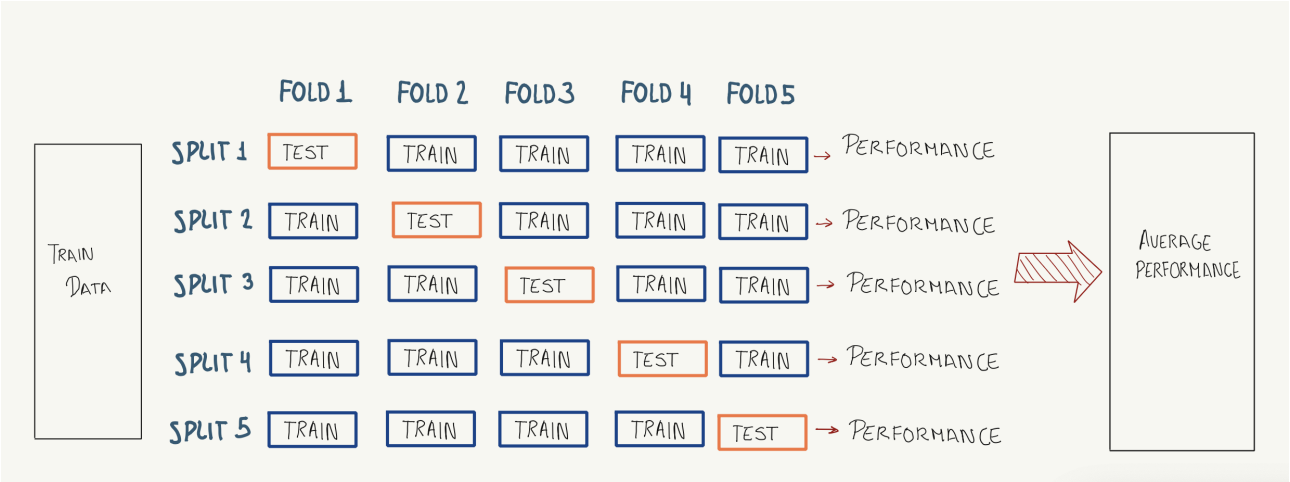

Figure 1: Own elaboration (Group 1)