Mathematical techniques in data science

Lecture 29: Cross-validation

Reminders

- Homework 4 on the course's webpage
- Check in with groups about projects this week
- I'm giving a talk at the Math Department's colloquium this Friday (3:30pm, 104 Gore Hall).
  Topic: Feature selection for non-linear models: (phylogenetic) trees and (deep neural) networks

Linear model: settings

- Linear model

$$Y = \beta^{(0)} + \beta^{(1)} X^{(1)} + \beta^{(2)} X^{(2)} + \dots \beta^{(p)} X^{(p)} + \epsilon$$

- Equivalent to

$$\mathbf{Y} = \mathbf{X}\beta, \qquad \beta = \begin{bmatrix} \beta^{(0)} \\ \beta^{(1)} \\ \dots \\ \beta^{(p)} \end{bmatrix}$$

Trade-off: complexity vs. interpretability

Linear model

$$Y = \beta^{(1)} X^{(1)} + \beta^{(2)} X^{(2)} + \ldots \beta^{(p)} X^{(p)} + \epsilon$$

- Higher values of p lead to more complex model $\rightarrow$ increases prediction power/accuracy
- Higher values of p make it more difficult to interpret the model

Regularization

- ℓ_0 regularization

$$\hat{\beta}^0 = \min_{\beta} \|\mathbf{Y} - \mathbf{X}\beta\|_2^2 + \lambda \sum_{i=1}^{p} \mathbf{1}_{\beta^{(i)} \neq 0}$$

- Ridge regression/Tikhonov regularization

$$\hat{\beta}^{RIDGE} = \min_{\beta} \|\mathbf{Y} - \mathbf{X}\beta\|_2^2 + \lambda \sum_{j=1}^{p} [\beta^{(j)}]^2$$

- Lasso

$$\hat{\beta}^{lasso} = \min_{\beta} \|\mathbf{Y} - \mathbf{X}\beta\|_2^2 + \lambda \sum_{j=1}^{p} |\beta^{(j)}|$$

Choosing parameters: cross-validation

- ℓ_0, ridge, lasso have regularization parameters λ
- We obtain a family of estimators as we vary the parameter(s)
- optimal parameters needs to be chosen in a principled way
- cross-validation is a popular approach for rigorously choosing parameters.

K-fold cross-validation

K-fold cross-validation:

Split data into K equal (or almost equal) parts/folds at random.
for each parameter λ_i **do**
  for $j = 1, \ldots, K$ **do**
    Fit model on data with fold j removed.
    Test model on remaining fold $\to$ j-th test error.
  end for
  Compute average test errors for parameter λ_i.
end for
Pick parameter with smallest average error.

K-fold cross-validation

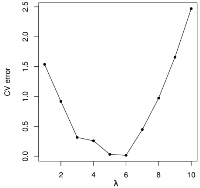
More precisely,

- Split data into K folds $F_1, \ldots, F_K$.

1	2	3	4	5
Train	Train	Validation	Train	Train

- Let $L(y, \hat{y})$ be a *loss function*. For example, $L(y, \hat{y}) = \|y - \hat{y}\|_2^2 = \sum_{i=1}^n (y_i - \hat{y}_i)^2$.
- Let $f_\lambda^{-k}(\mathbf{x})$ be the model fitted on all, but the k-th fold.
- Let

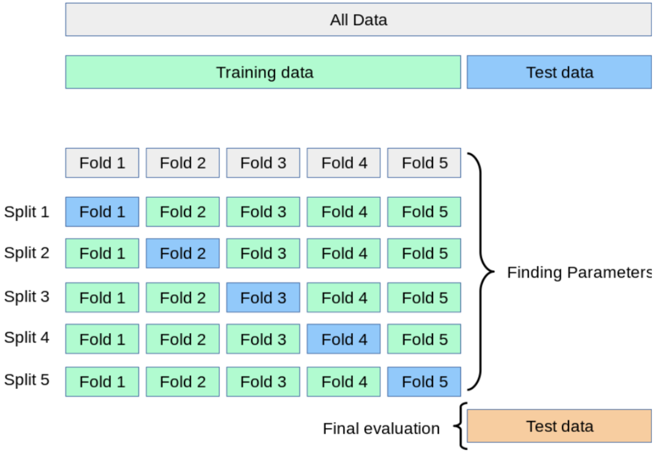
$$CV(\lambda) := \frac{1}{n} \sum_{k=1}^{n} \sum_{i \in F_k} L(y_i, f_\lambda^{-i}(\mathbf{x}_i))$$



- Pick λ among a *relevant* set of parameters

$$\hat{\lambda} = \underset{\lambda \in \{\lambda_1, \ldots, \lambda_m\}}{\operatorname{argmin}} CV(\lambda)$$

K-fold cross-validation



Demo: Cross-validation with Lasso