Lecture 4: Cross-Validation

MAST30034 Applied Data Science

Dr. Karim Seghouane
School of Mathematics & Statistics
The University of Melbourne

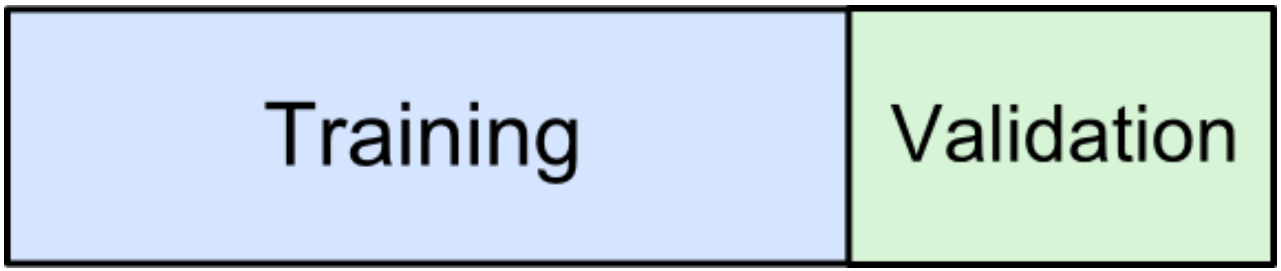
August 8, 2023

Outline

- Finding optimal tuning parameter settings
- Cross validation
- Measures of accuracy
- Measures of clustering quality

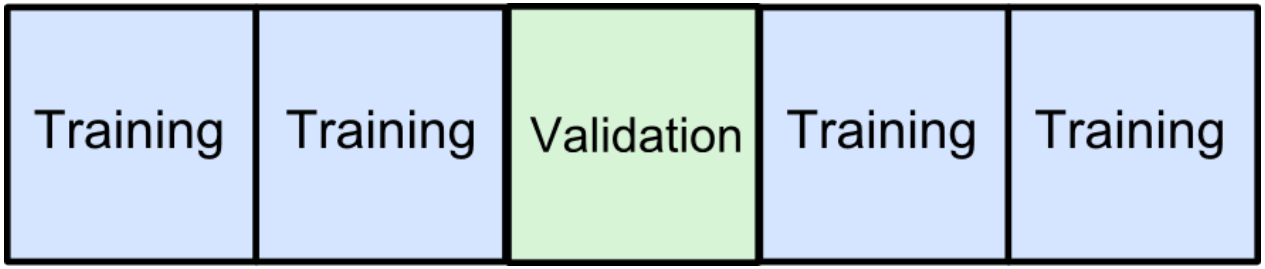
Strategies for designating training and testing data

- A large designated test dataset that has similar characteristics to training dataset
- Naive approach: randomly divide available dataset into: training set and validation set
- Validation estimates of test error based on naive approach can be highly variable, depending on how lucky you are.

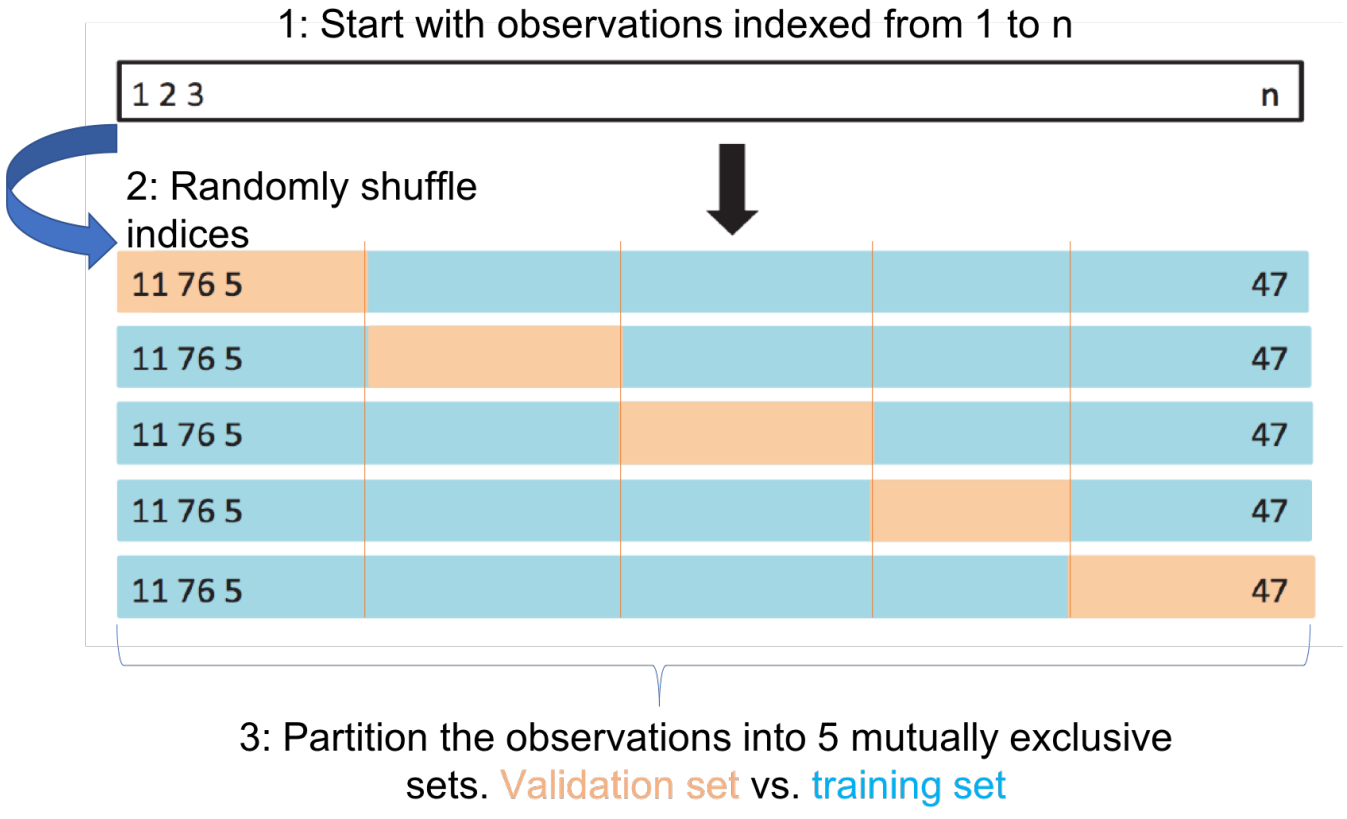


Strategies for designating training and testing data

- Reliable approach: K−fold cross validation (CV)
- Rigorous approach: Repeated K−fold cross validation (CV)



K−fold Cross Validation



Is there a particular type of data that is unsuitable for cross-validation?

Finding the optimal tuning parameter settings

Lasso Regression

Training data $\{\mathbf{x}_i, y_i\}_{i=1}^{n_{train}}$ and $\boldsymbol{\beta} \in \mathbb{R}^p$. The lasso regression coefficients is the minimiser

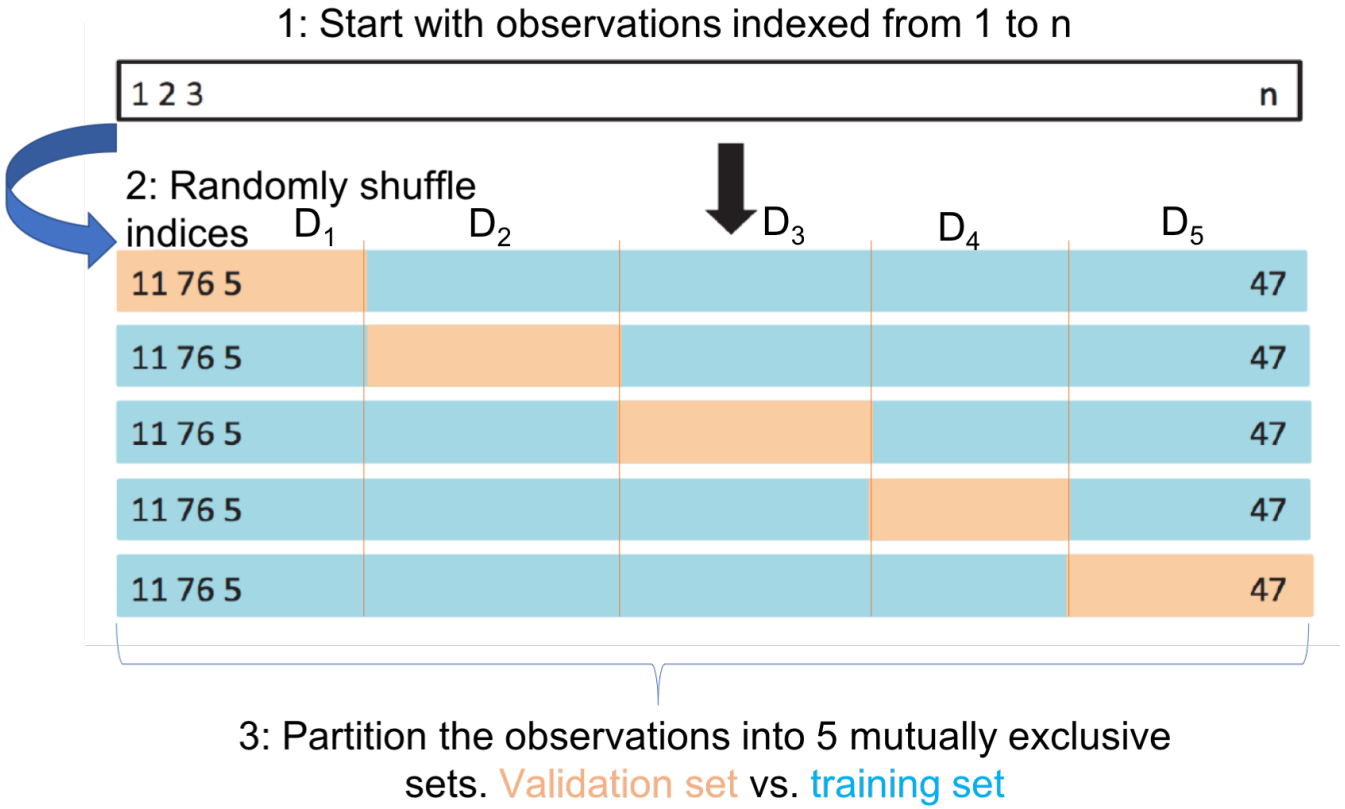
$$\widehat{\boldsymbol{\beta}}_{lasso} = \arg\min_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{n_{train}} (y_i - \mathbf{x}_i^\top \boldsymbol{\beta})^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}.$$

Prediction equation:

$$\widehat{y}_i = \mathbf{x}_i^\top \widehat{\boldsymbol{\beta}}_{lasso}$$

How do we choose λ?

Step 1: Use this to partition training data into 5 sets



5−fold cross validation for choosing λ

1. Partition TRAINING data into 5 sets
$\mathcal{D} = \mathcal{D}_1 \cup \mathcal{D}_2 \cup \mathcal{D}_3 \cup \mathcal{D}_4 \cup \mathcal{D}_5$.

2. For $s = 1, \ldots, 5$, use $\mathcal{D}_s$ as validation set and $\mathcal{D}_{-s} = \mathcal{D} \backslash \mathcal{D}_s$ as training set. Compute sum of squared squared error of each partition

$$E_s(\lambda) = \sum_{i \in \mathcal{D}_s} (y_i - \widehat{y}_i)^2.$$

3. Compute mean squared error

$$\text{MSE}(\lambda) = \frac{1}{n_{train}} \sum_{s=1}^{5} E_s(\lambda),$$

where $n_{train} = |\mathcal{D}|$.

4. Repeat steps 2 to 4 for various candidate values $\lambda_1, \ldots, \lambda_M$.

The optimal value of λ is

$$\lambda_{opt} = \underset{l=1\ldots,M}{\text{argmin}}\, \text{MSE}(\lambda_l)$$

Repeated 5-fold cross validation for choosing λ

λ_{opt} depends on our choice of $\mathcal{D}_1, \ldots, \mathcal{D}_5$. For a dataset of size $5n$, the number of ways to partition data is $\frac{(5n)!}{n!^5}$. Too many possibilities to consider!

Repeated 5-fold CV:
For $t = 1, \ldots, B$,

1. Draw a random partition $\mathcal{D}^{(t)} = \{\mathcal{D}_1^{(t)}, \ldots, \mathcal{D}_5^{(t)}\}$ of the training data

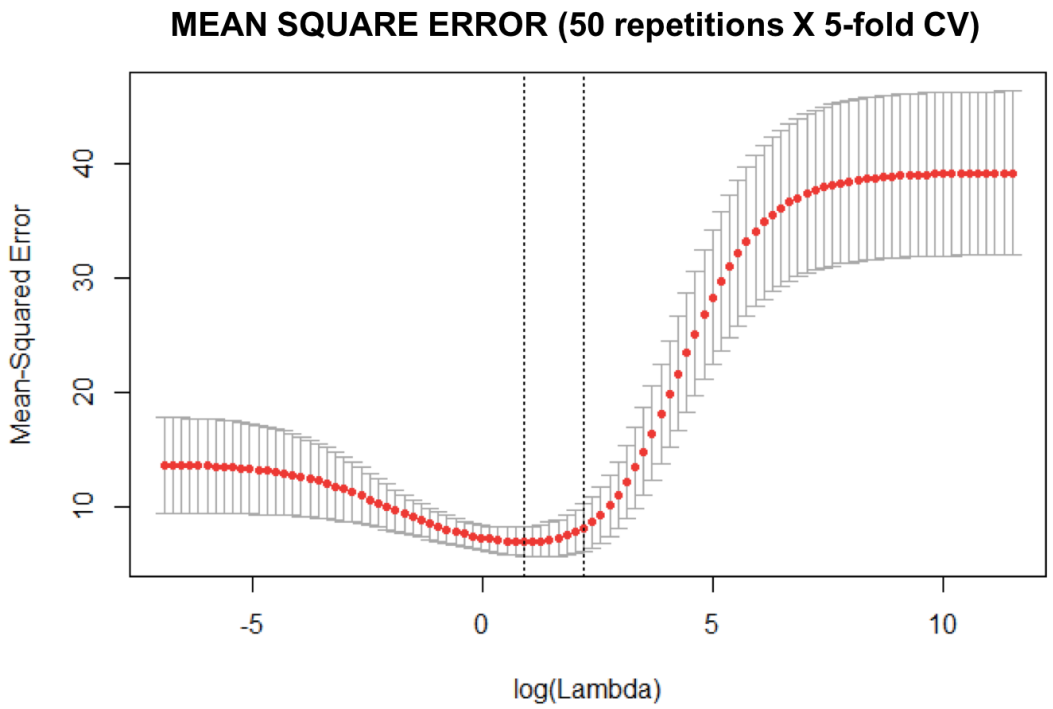
2. For each candidate value λ_l, run 5-fold CV and compute

$$\text{MSE}^{(t)}(\lambda_l) = \frac{1}{n_{train}} \sum_{s=1}^{5} \sum_{i \in \mathcal{D}_s^{(t)}} (y_i - \widehat{y}_i)^2.$$

3. The optimal λ is

$$\lambda_{opt} = \underset{l=1\ldots,M}{\text{argmin}} \sum_{t=1}^{B} \text{MSE}^{(t)}(\lambda_l).$$

MSE (with error bars) over various candidate values of λ



Assessing performance of final model

Back to Lasso Regression Example

Testing dataset: y_i is response, $\mathbf{x}_i$ are features.

The predicted response of the i-th test data point is:

$$\widehat{y}_i = \mathbf{x}_i^\top \widehat{\boldsymbol{\beta}}_{opt}$$

where $\widehat{\boldsymbol{\beta}}_{opt}$ minimises

$$\sum_{i=1}^{n_{train}} (y_i - \mathbf{x}_i^\top \boldsymbol{\beta})^2 + \lambda_{opt} \sum_{j=1}^{p} |\beta_j|.$$

Selected features = set of features with non-zero coefficients,

$$= \{j : |\widehat{\beta}_{opt,j}| \geq 0\}.$$

Supervised learning: measures of accuracy

For continuous response, the **mean squared prediction error** is

$$\text{MSPE} = \frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} (y_i - \widehat{y}_i)^2.$$

For categorical response, we calculate the balanced classification error rate

$$\text{Balanced error} = \frac{1}{C} \sum_{r=1}^{C} \frac{1}{n_r} \sum_{i:y_i=C_r} \mathbb{I}\{y_i \neq \widehat{y}_i\},$$

where C is the total number of categories and C_r is the number of observations in testing data that belong to category r.

Unsupervised learning

Example: K-means clustering

Recall that in k-means clustering we have $\mathbf{x}_i$ as the input. We assume that the are K underlying groups. The algorithm is:

1. Intialise the centroids $\boldsymbol{\mu}_1^{(0)}, \ldots, \boldsymbol{\mu}_K^{(0)}$.

At iteration t,

2. We assign $\mathbf{x}_i$ to their respective group

$$g_i^{(t)} = \underset{k=1,\ldots,K}{\text{argmin}} \, \text{Dist}(\mathbf{x}_i, \boldsymbol{\mu}_k^{(t)})$$

.

3. Re-evaluate centroids

$$\boldsymbol{\mu}_k = \frac{1}{|C_k|} \sum_{i \in C_k} \mathbf{x}_i,$$

where C_k is the set of indices of the k-th group.

Next iteration $t \leftarrow t + 1$

4. Repeat steps 2 to 3 until SSE is sufficiently small, where

$$\text{SSE} = \sum_{k=1}^{K} \sum_{i \in C_k} (\mathbf{x}_i - \boldsymbol{\mu}_k^{(t)})^2$$

Optimal K: Elbow method