M.L * Lec - VI

Date :
Page :

* Cross-validation : Training set

This is the portion of dataset used to train the ML model. The model learns patterns and relation from the data in training set.

validation-set : After, training the model on training set, it's important to evaluate it's performance on data it hasn't seen before.

tuning hyperparameter → ie, λ in ridge regression
learning rate in gradient descent
nos of hidden layer in neural network

Accessing generalization ability - Generalization refer to how well the trained model performed well on unseen data. By evaluating model's perfor. on validation set we can estimate how well it will generalize to new unseen data

** If model perform well on validation-set it indicate good-generalization ability. However, if perform poorly it may indicate overfitting (memorizing training data without capturing underlying patterns) or Underfitting (fail to capture underlying pattern in data).

Test set : reserved for final evaluation of the model's performance.

Types

* k-fold cross validation
* Leave-one-out cross validation
* Hold-out validation
* Stratified cross validation

Cross-validation is a technique used in ML to evaluate the performance of a model on unseen data

It involve dividing the available data into multiple folds or subsets using, one of these folds as validation set & training remaining folds.

Leave-one-out cross validation (LOO CV)

Leave-p-out, dataset n=100 Let p=5

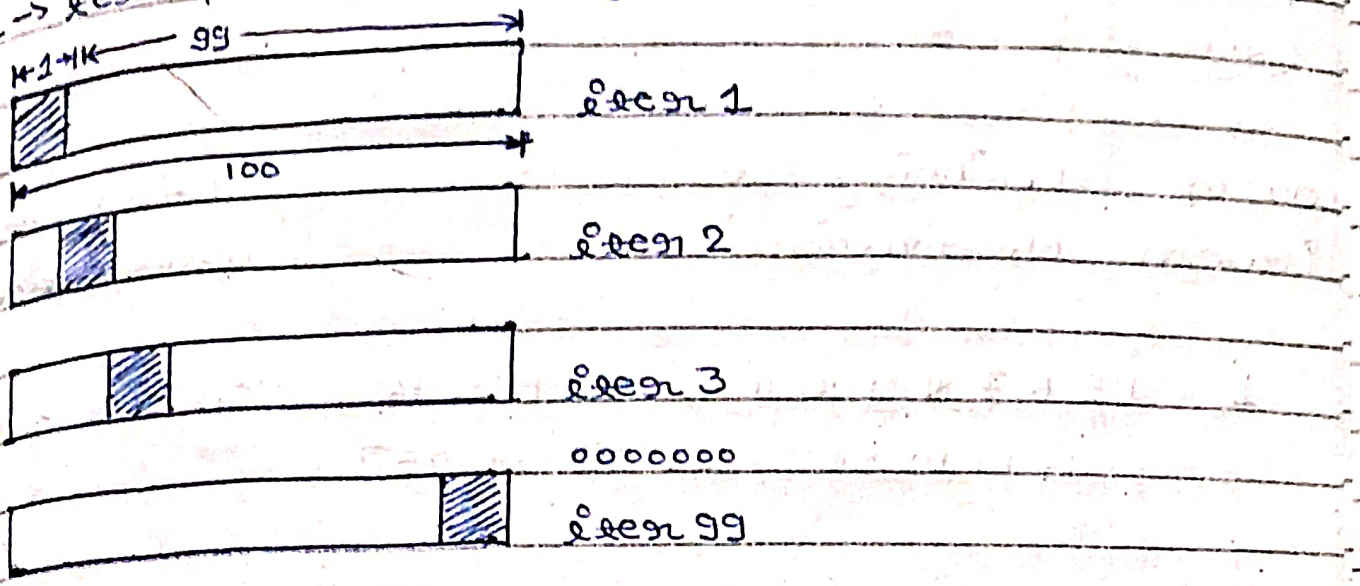
when p = 1 → leave-one-out

iterative check
train : n-p
test : p.
(validate-set)

LOO CV is a special case, where nos of folds k equals to the nos of datapoints in dataset.

The model is trained on n-1 samples, and tested on the one omitted sample, repeating the process for each datapoint in the dataset.

1 -> test , 99 -> training



- we can make use of all data-points hence it's low bias

- leads to higher variation in the testing model as we are testing against one data-point (outlier - lead to variation)
- lot of execution time.

K-fold cross validation

We split the dataset into k-numbers of subsets (known as folds) then we perform training on the all the subsets but leave one (k-1) subset for the evaluation of the trained model.

In this method, we iterate k times with a different subset reserved for testing purpose each time

Total instance : 25
value of k : 5

nos of iteration	Training set observation	Testing set observation
1	[5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25]	[0 1 2 3 4]
2	[0 1 2 3 4 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25]	[5 6 7 8 9]
3	[0 1 2 3 4 5 6 7 8 9 15 16 17 18 19 20 21 22 23 24 25]	[10 11 12 13 14]
4	[0 1 2 3 4 5 6 7 8 9 20 21 22 23 24 25]	[15 16 17 18 19]
5	[0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19]	[20 21 22 23 24]

Note: It's always suggested that value of k should be 10 as the lower value of k is takes towards validation and higher value of k leads to LOO CV method.

Cross-validation

- overcoming overfitting
- model selection
- hyperparameter tuning
- data-efficient

- computationally expensive
- time-consuming
- bias-variance tradeoff : The choice of nos of folds in cross-validation can impact b-v tradeoff. ie if too few folds result in high variance while too many folds lead in high bias-

Cross-validation Exercise 9

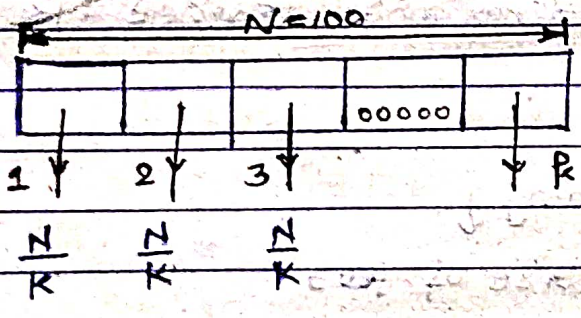
Q. Suppose we want to compute 10-Fold Cross-validation error on 100 training examples. We need to compute error N1 times, and the cross validation error is the average of the errors. To compute each error, we need to build a model with data of size N2, and test the model on data of size N3. What are the appropriate numbers for N1, N2, N3?

- ✓ N1 = 10, N2 = 90, N3 = 10
- ✗ N1 = 1, N2 = 90, N3 = 10
- ✗ N1 = 10, N2 = 100, N3 = 10
- ✗ N1 = 10, N2 = 100, N3 = 100.

k = 10

N = Samples = 100

k-cross validation,
N = samples in training



* generalization error of your model
* hyperparameter tuning.

* k models are trained and evaluated.

model 1 : 1, 2, 3, ..., k-1, k

k-1 fold for training

1 fold for testing.

model 2 : 1, 2, 3, ..., k-1, k

for testing

for training

model k th : 1, 2, 3, ..., (k-1), k → testing.

all the folds come for testing at least once.

f_k models, f_k folds

Nos of samples for training

$\Rightarrow (f_k - 1)\left(\frac{N}{f_k}\right)$

Nos of samples for testing

$\Rightarrow (1)\left(\frac{N}{f_k}\right)$

$N_1 = ?$

N_2 = training datasize

N_3 = testing data size

$N_1 \Rightarrow$ Nos of times error is computed = nos of times model trained = f_k

$(f_k - 1)$ folds ↴

total samples in training ↴

$\therefore (f_k - 1) \times$ size of fold

$\Rightarrow (f_k - 1) \times \frac{N}{f_k} \Rightarrow \frac{(10-1)(100)}{10}$

$\Rightarrow \frac{9 \times 100}{10} \Rightarrow 90$ → N_2

for testing

(1) fold $\left(\frac{100}{10}\right) \Rightarrow 10$ samples → N_3

12) Suppose we are performing leave-one-out (LOO) validation and 10-fold cross validation on a dataset of size 100,000 to pick between 4 different values of a single hyperparameter. How many times greater is the number of models that need to be trained for LOO validation versus 10-fold cross-validation?

Hyperparameter → 4 values.

$x_1, x_2, x_3, \ldots, x_n$

for every value → train models / CV

compute errors

value with least cross-validation error is chosen

$N = 100,000$, $x = 4$ times

How many times → models trained for LOO CV → model trained for 10-fold CV

(M1) 10-fold CV

$N = 100,000$

10 fold = K

model will be trained $K = 10$ times

for 4 parameter ⇒ 4 × 10

⇒ 40 times

(M2) LOO-CV

N samples ⇒ (1), 2, 3, ..., N-1, N ∴ model 1 (1: test, 2 to N: train)

1, (2), 3, 4, ..., N-1, N ∴ model 2 (2: test, 1 and 3 to N: train)

1, 2, 3, ..., N-1, (N) ∴ model k (N: test, 1 to N-1: train)

testing ⇒ $\frac{N}{K}$ samples here $\frac{N}{K} = 1$ (LOO CV)

$K = N$

Nos of models for LOO-CV ⇒ N

for 4 parameter

⟶ 4 × 100,000 ⇒ 400,000

$$\frac{\text{LOO CV}}{\text{K-fold}} \rightarrow \frac{400,000}{40}$$

⇒ 10,000 times greater

(3) cross-valid used for

↓

(i) evaluate performance of ML model on unseen data ✓

↓

(2) select hyper parameters ✓

↓

(3) to determine the generalization of ML model ✓

↓

on both test / train performance

↓

(4) To train multiple ML model on diff. datasets ✗

∈ all the model that we train in CV are same models