



CS 329P : Practical Machine Learning (2021 Fall)

4.3 Model Validation

Qingqing Huang, Mu Li, Alex Smola

<https://c.d2l.ai/stanford-cs329p>

Generalization Error

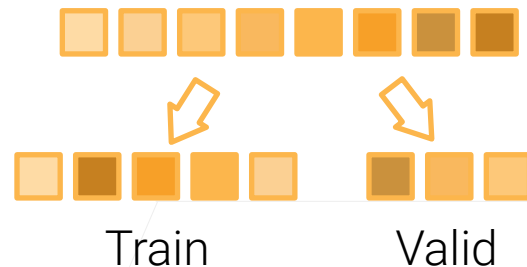


- Approximated by the error on a holdout **test dataset**, which has never been seen by the model and can be only used **once**
 - Your midterm exam score
 - The final price of a pending house sale
 - Dataset used in private leaderboard in Kaggle
- **Validation dataset:**
 - Often a subset of the dataset, not used for model training
 - Can be used multiple times for hyper param tuning
 - “test error” usually refers to error on “validation” dataset

Hold Out Validation



- Split your data into “train” and “valid” sets (often calls “test”)
 - Train your model on the train set, use the error on the valid set to approximate the generalization error
- Often randomly select $n\%$ examples as the valid set
 - Typical choices $n = 50, 40, 30, 20, 10$



Split non I.I.D. data



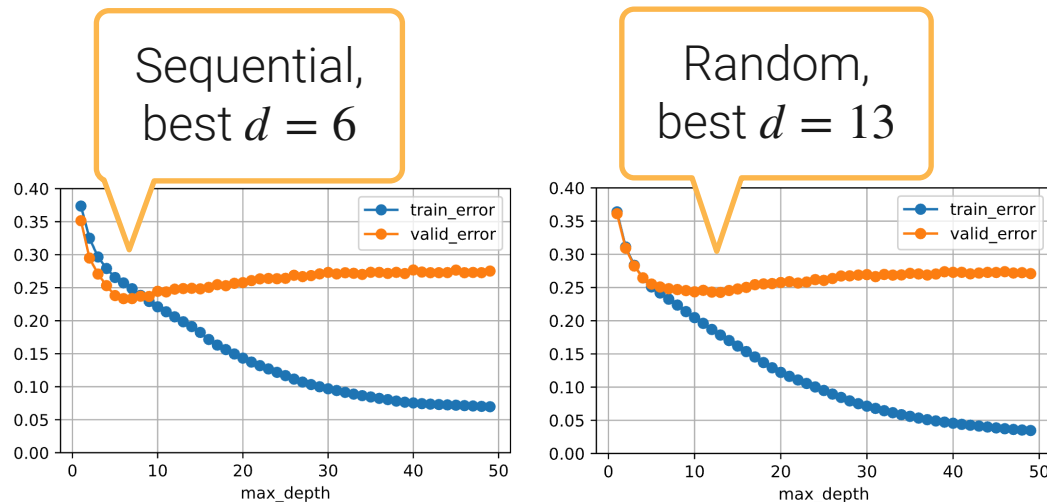
- Random data splitting may lead to underestimate of generalization error
- Sequential data
 - e.g. house sales, stock prices
 - Valid set should not overlap with train set in time
- Examples are highly clustered
 - e.g. photos of the same person, clips of the same video
 - Split clusters instead of examples
- Highly imbalanced label classes
 - Sample more from minor classes

Case Study on House Sales Data



- Split by 50%, test both random and sequential splittings

Decision tree



Linear regression

Split	Train	Valid
Random	0.126	0.136
Sequential	0.109	995901.7

K-fold Cross Validation



- Useful when not sufficient data
- Algorithm:
 - Partition the training data into K parts
 - For $i = 1, \dots, K$
 - Use the i -th part as the validation set, the rest for training
 - Report the validation error averaged over K rounds
- Popular choices: $K = 5$ or 10

Data

--	--	--

Fold 1:

Valid	Train	Train
-------	-------	-------

Fold 2:

Train	Valid	Train
-------	-------	-------

Fold 3:

Train	Train	Valid
-------	-------	-------

Common Mistakes



- If your ML model performance is too good to be true, very likely there is a bug, and contaminated valid set is the #1 reason.
- Valid set has duplicated examples from train set
 - Often happens when integrating multiple datasets
 - Scrape images from search engine to evaluate models trained on ImageNet
- Information leaking from train set to valid set
 - Often happens for non I.I.D data
 - use future to predict past, see a person's face before
- Excessive use of valid set for hyper param tuning is cheating

Summary



- The test data is used once to evaluate your model
- One can hold out a validation set from the training data to estimate the test data
 - You can use valid set multiple times for model selections and hyper param tuning
 - Validation data should be close to the test data
 - Improper valid set is a common mistake that lead to over estimate of the model performance