6. Cross-Validation

Cross Validation

Exhaustive Cross-Validation

- Exhaustive cross-validation methods learn and test on all possible ways of dividing the original dataset into a training and a validation set.
- Exhaustive cross-validation methods are very computationally intensive.
- Examples include (a) leave-p-out cross-validation (b) leave-one-out cross-validation.

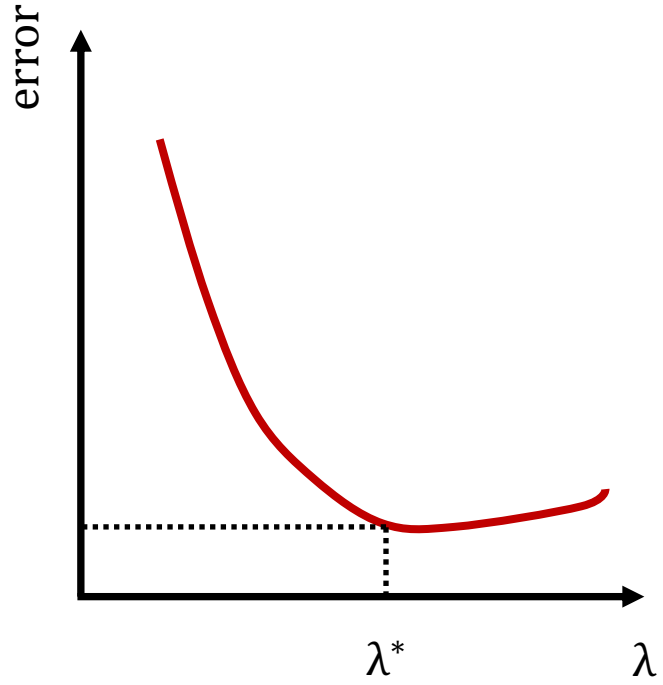
Non-Exhaustive Cross-Validation

- Non-exhaustive cross-validation methods do not learn and test on all possible ways of dividing the original dataset into a training and a validation set.
- Non-exhaustive cross-validation methods are less computationally intensive.
- Examples include k-fold cross-validation.

Cross Validation: K-fold CV

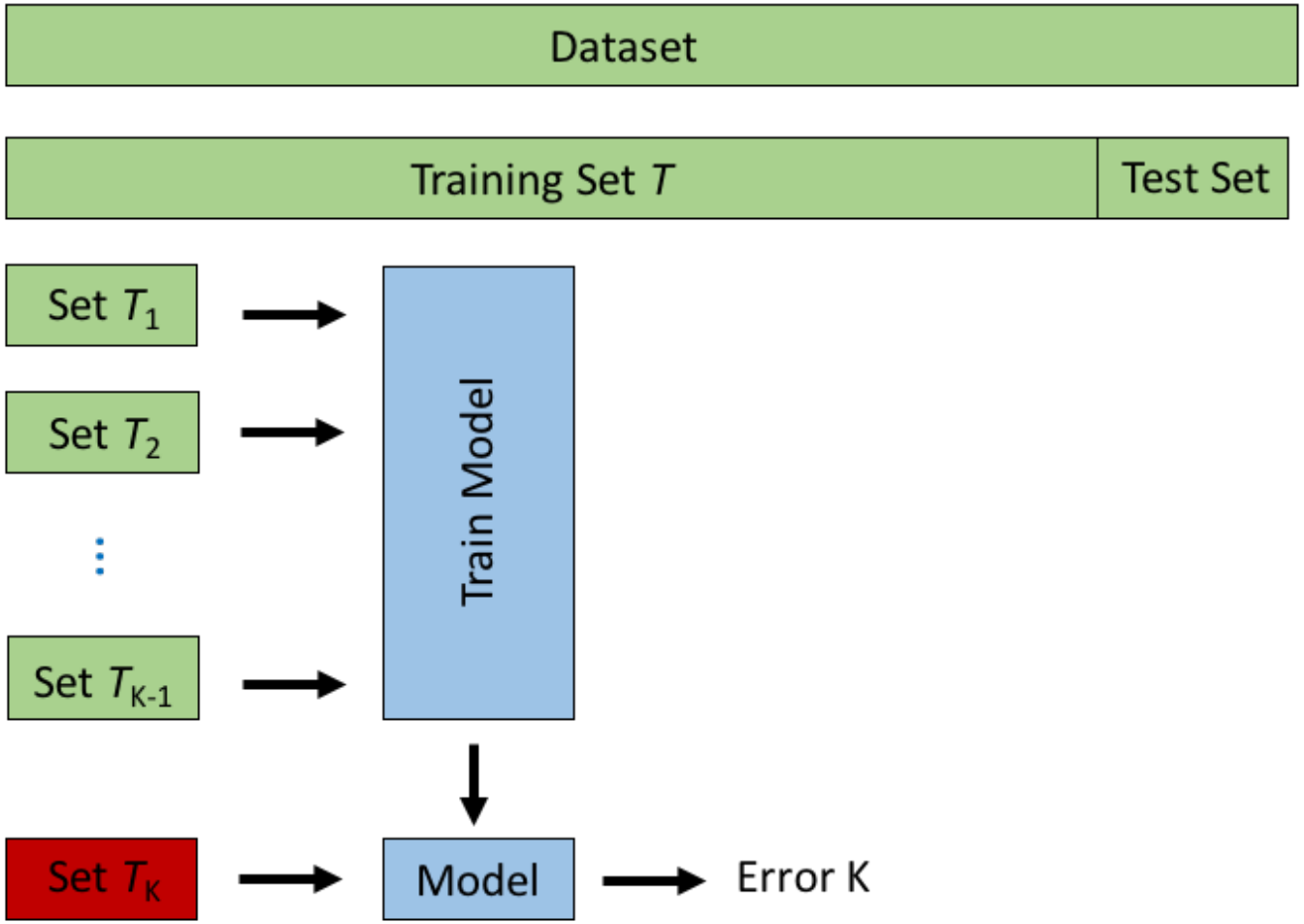
Procedure

- Select a grid of potential values for the hyper-parameter value $\lambda \in \{\lambda_1, \lambda_2, \ldots, \lambda_t\}$
- Partition the entire dataset onto a training and testing set
- Partition the entire training set *T* onto *K* subsets T_1, T_2, …, T_K
- Calculate the error $error_k$ on subset T_k based on a model learnt on the training set *T* except subset T_k
- Calculate the overall error corresponding to the average of the individual errors errork (k=1,…,K)
- Choose the hyper-parameter value leading to the smallest overall error
- The model associated with such optimal hyper-parameter is then trained on the entire training set and it is tested on the test set



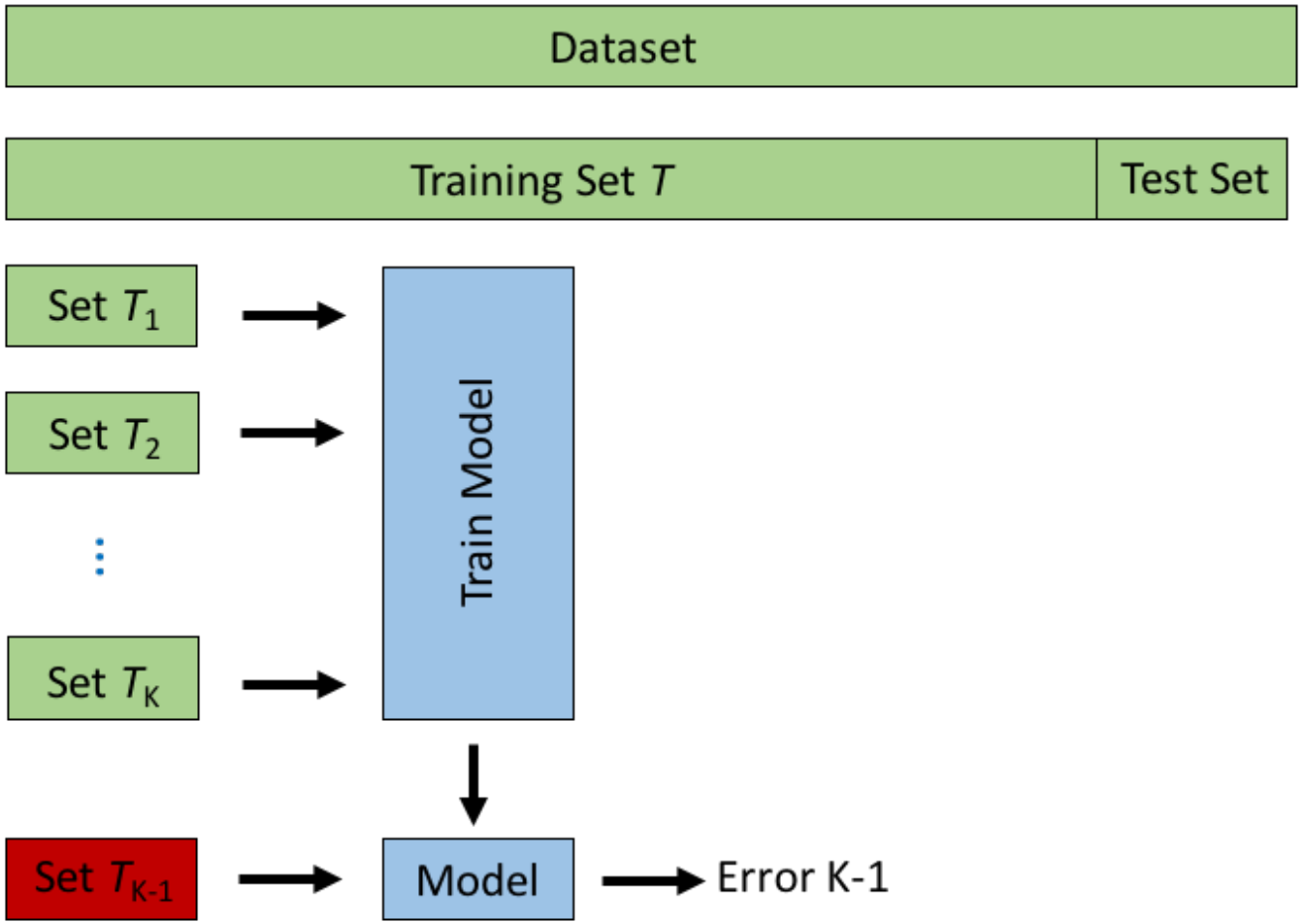
Cross Validation: K-fold CV

Procedure



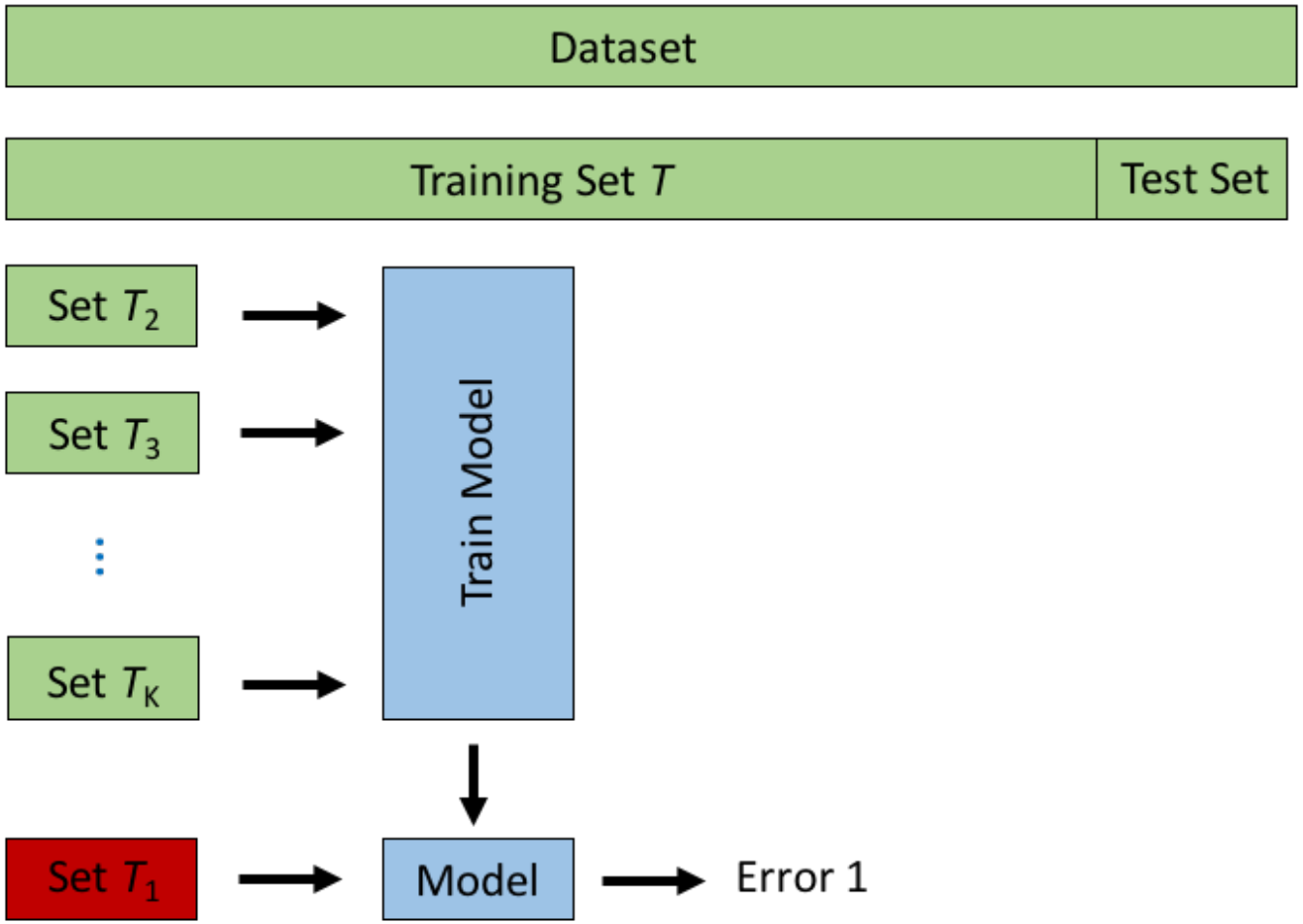
Cross Validation: K-fold CV

Procedure



Cross Validation: K-fold CV

Procedure



Cross Validation: K-fold CV

Procedure

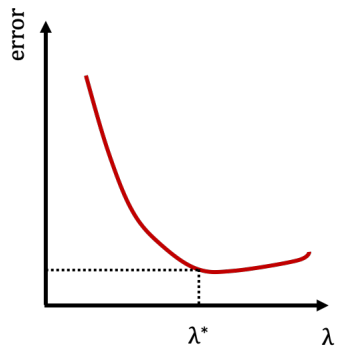
$\text{Error}_1(\lambda)$

$\text{Error}_2(\lambda)$

$\vdots$

$\text{Error}_K(\lambda)$

→ Avg. Error(λ) → λ^* → Model



→ Model is trained on the entire training set

→ Trained model is tested on the testing set