CSC 461: Machine Learning

Fall 2024

Model Selection

Prof. Marco Alvarez, Computer Science
University of Rhode Island

Model selection

- Model selection
  - **<u>crucial step</u>** when applying machine learning
  - involves choosing the best model from a set of candidate models
  - process involves evaluating and comparing different models based on their performance (use evaluation metrics)
- Goals of selecting the best model
  - enhancing generalization
  - preventing overfitting/underfitting
- Hyperparameter tuning
  - optimizing model-specific parameters
  - techniques: grid search, random search

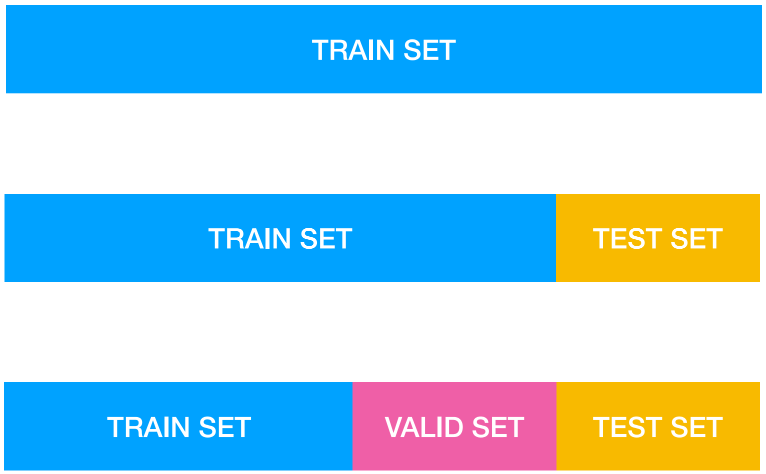
Overfitting and underfitting

- Overfitting
  - a model learns the training data too well, leading to poor generalization performance on unseen data
- Underfitting
  - a model is too simple to capture the underlying patterns in the data

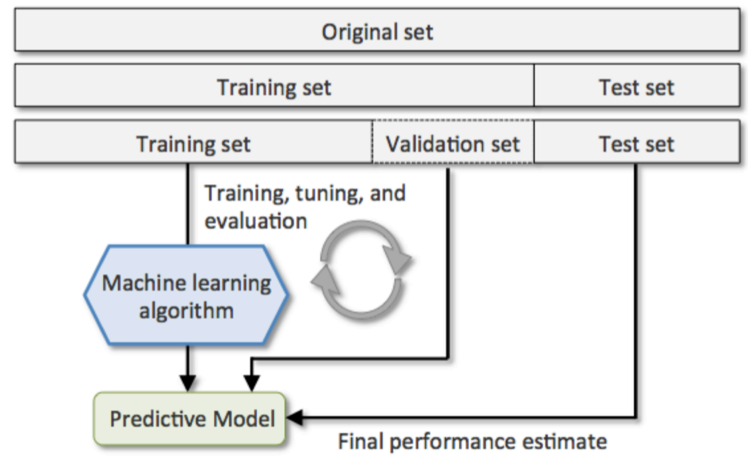
Train, validation, and test

- Divide the available data into three subsets:
  - <u>training set</u>: used to train the model
  - <u>validation set</u>: used to tune hyperparameters and select the best model
  - <u>test set</u>: used to evaluate the final model's performance
- Rationale
  - aims to simulate the model's performance on unseen data
  - by keeping a portion of the data (test set) completely separate from the training and model selection, we can get an unbiased estimate of the model's performance in real-world scenarios

Train, validation, and test



Train, validation, and test



from INFO-4604: Applied Machine Learning, Fall 2017, Michael Paul, Univ. of Colorado

k-fold cross-validation

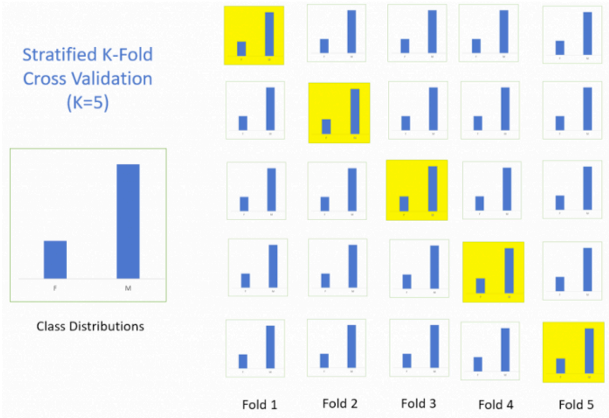
- Resampling procedure used to evaluate machine learning models
  - generally applied with small datasets
- Process
  - split the dataset into k subsets (folds)
  - train a model on k-1 folds and evaluate performance on the remaining fold (validation set)
  - repeat the last step k times, with each fold serving as the validation set once
  - calculate the average performance across all k folds
- Stratified cross-validation
  - variation of k-fold cross-validation that ensures that the proportion of samples for each class is roughly the same in each fold

k-fold cross-validation

k = 5							
	fold 1	valid	train	train	train	train	performance
	fold 1	train	valid	train	train	train	performance
	fold 1	train	train	valid	train	train	performance
	fold 1	train	train	train	valid	train	performance
	fold 1	train	train	train	train	valid	performance

The model's performance is averaged across all k iterations

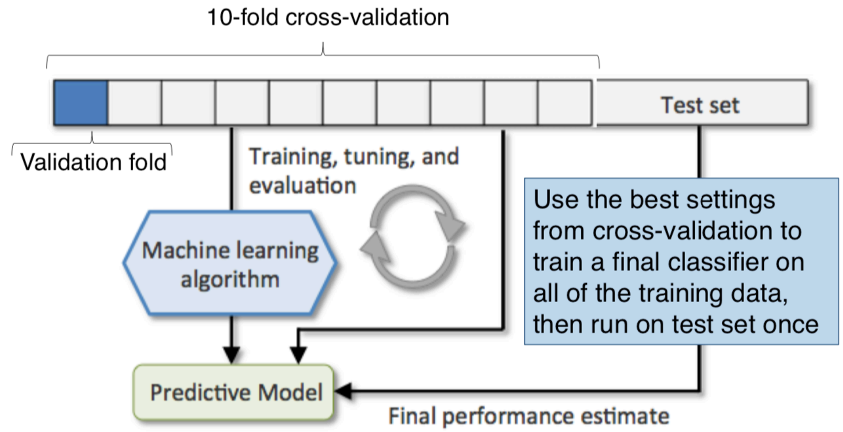
Stratified k-fold cross-validation



Stratified CV aims at keeping the same class distribution within each fold

https://towardsdatascience.com/cross-validation-explained-evaluating-estimator-performance-e51e5430ff85

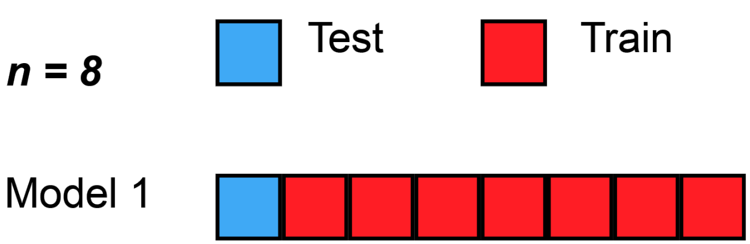
CV and a separate holdout test set



from INFO-4604: Applied Machine Learning, Fall 2017, Michael Paul, Univ. of Colorado

Leave-one-out cross-validation

- Special case of CV when $k = n$
- Can be expensive for large n



https://en.wikipedia.org/wiki/Cross-validation_(statistics)

Data normalization

- Goal
  - scaling features to a common range
- Why?
  - helps preventing overfitting by ensuring that all features contribute equally to the model
  - many ML algorithms converge faster when data is normalized
- Common techniques
  - standardization, min-max scaling, robust scaling

The normalization parameters (e.g., mean and standard deviation) should be computed only on the training set and then applied to the validation and test sets.

This principle applies to train/validation/test splits and cross-validation approaches.

Strategy	Pros	Cons	Best Used When
train/validation/test split	– simple to implement – fast	– less efficient use of data – results can vary depending on the specific split	– large datasets – quick initial model evaluation
k-fold cross-validation	– more robust estimates of model performance – makes better use of available data	– computationally expensive – can be slow for large datasets	– smaller to medium-sized datasets – when robust performance estimates are crucial
stratified k-fold cross-validation	– maintains class proportions in each fold	– slightly more complex to implement – can be slow for large datasets	– imbalanced datasets – when maintaining class proportions is important