Chapter Five

Model Evaluation

Prepared by: Tsehay A. (B.Sc., and M.Sc., in Computer Science)

Department of Computer Science

2015 E.C

Contents

- Data processing
- Model selection and tuning
- Methods of dimensionality reduction
- Performance evaluation methods
- Optimize the performance of the model
- Control model complexity
- Over-fitting and Under-fitting
- Cross-Validation and Re-sampling methods
- Bootstrapping
- Bias and variance

Data processing

- Real-world data is incomplete, noisy, and contains irrelevant and redundant information or errors.
- Data processing transforms the raw data into an understandable format.
- Data processing involves
  - ✓ Data cleaning
  - ✓ Feature scaling
  - ✓ Outlier removal
  - ✓ Data reduction
  - ✓ Resampling

Cont.

- Data cleaning removes noise and corrects inconsistencies in the data.
- Data integration merges data from multiple sources into a coherent data store.
- Data reduction can reduce the data size by aggregating, eliminating redundant features, or clustering.
- Data transformations, such as normalization, may be applied, where data are scaled to fall within a smaller range such as 0.0 to 1.0.

Model selection and tuning

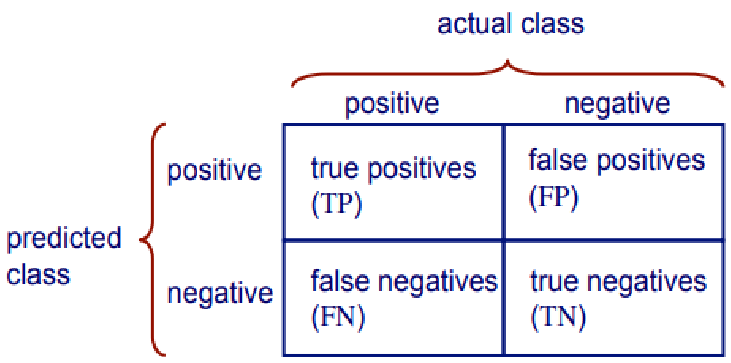
- The performance of a model can be improved with parameter tuning.
- Model selection is the process of choosing one among candidate models for a predictive modeling problem.
- There may be many competing concerns when performing model selection such as performance, complexity, maintainability, and available resources.
- The two main classes of model selection techniques are probabilistic measures and resampling methods.

Cont.

- Model selection can be applied to different types of models (e.g. logistic regression, SVM, KNN, etc.) and models of the same type configured with different model hyperparameters (e.g. different kernels in an SVM).
- The best approach to model selection requires "*sufficient*" data, which is nearly infinite depending on the complexity of the problem.
- In the ideal situation, we would split the data into training, validation, and test sets, then fit candidate models on the training set, evaluate and select them on the validation set, and report the performance of the final model on the test set.

Model performance evaluation methods

- How can we understand what types of mistakes a learned model makes?
- Confutation matrix-. A confusion matrix is a technique for summarizing the performance of a classification algorithm.
- Accuracy-the fraction of the total samples that were correctly classified by the classifier. To calculate accuracy, use the following formula: A=(TP+TN)/(TP+TN+FP+FN).
- Is accuracy an adequate measure of predictive performance?

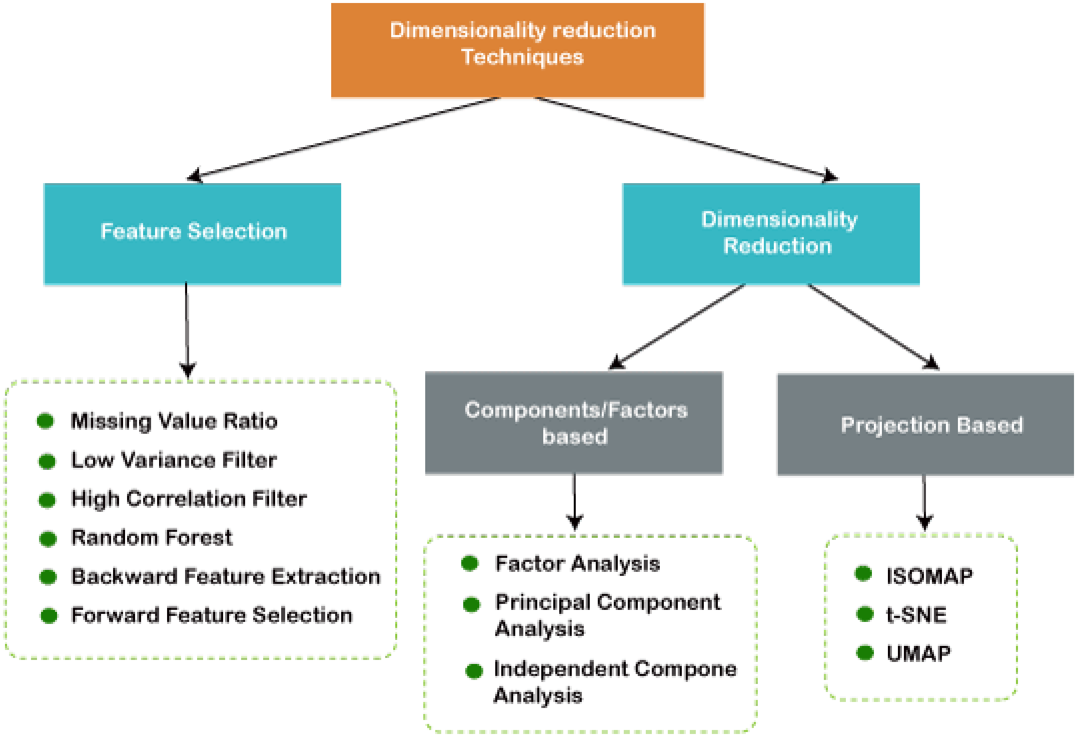


		actual class	
		positive	negative
predicted class	positive	true positives (TP)	false positives (FP)
	negative	false negatives (FN)	true negatives (TN)

$$\text{accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

Method of dimensionality reduction

- Reducing variables in a training dataset used to develop machine learning models.
- Dimensionality reduction projects high dimensional data to a lower dimensional space that encapsulates the essence of the data.



Optimize the performance of the model

- Some of the common methods for optimization of model performance include:
  - ✓ Parameter tuning
  - ✓ Feature selection
  - ✓ Feature scaling
  - ✓ Data resampling

Control model complexity

- Model complexity often refers to the number of features or terms included in a given predictive model, as well as whether the chosen model is linear, or nonlinear.
- Choosing the value to use for the tuning parameter is critical and can be done using a technique such as cross-validation.
- The number of predictor or independent variables or features that a model needs to take into account in order to make accurate predictions.

Methods of controlling model complexity

- There are a few ways to prevent these problems.

    ✓Use simpler models.

    ✓Split the data into a training set and a test set.

    ✓Use early stopping

    ✓Use cross-validation

    ✓Monitor the performance of the model

Over-fitting and under-fitting

- Overfitting is a common pitfall in machine learning modelling, in which a model tries to fit the training data entirely and ends up “memorizing” the data patterns and the noise/random fluctuations.
- These models fail to generalize and perform well in the case of unseen data scenarios, defeating the model’s purpose.
- Overfit model results in poor test accuracy.
- Example of overfitting situation in classification and regression: If a model has 99% accuracy on the training set but only 55% accuracy on the test set.

Cont.

- Underfitting: occurs when the model cannot create a mapping between the input and the target variable that reflects the underlying system, for example due to under-observing the features.
- Underfitting often leads to a higher error in the training and unseen data samples.
- A model which does not capture the underlying relationship in the dataset on which it's trained is called overfit model.
- Example of underfiting situation in classification and regression: the model performs 90 % on the training but 80% on testing set.

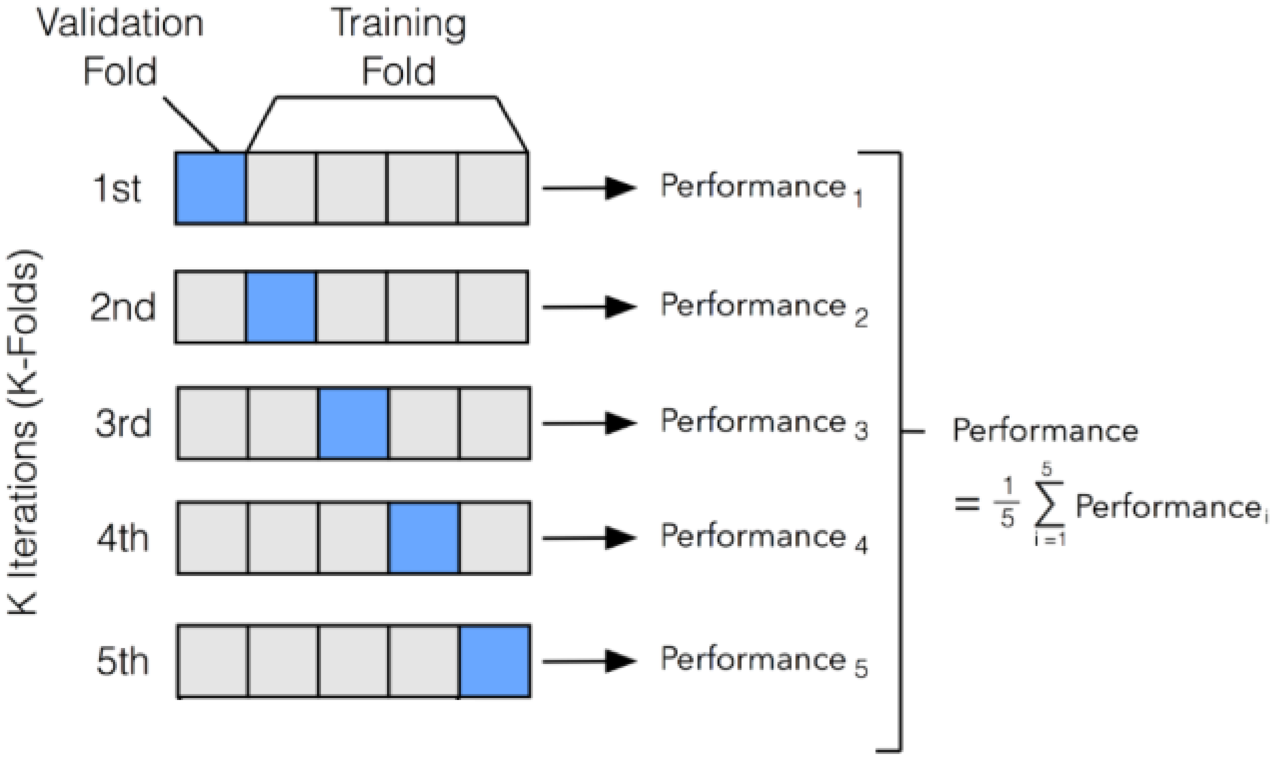
Cross-validation and resampling methods

- The k-fold Cross-validation (CV) can be conducted on a single training set for both training and validation.
- The CV randomly divides the dataset of n observations into k groups (or folds) of approximately equal size.
- Then, for each group i=1,...,k fold i is treated as a validation set, and the model is fit on the remaining k-1 folds.
- The performance metric, Performance, (e.g. accuracy), is then computed based on observations of the held out fold i.

Cont.

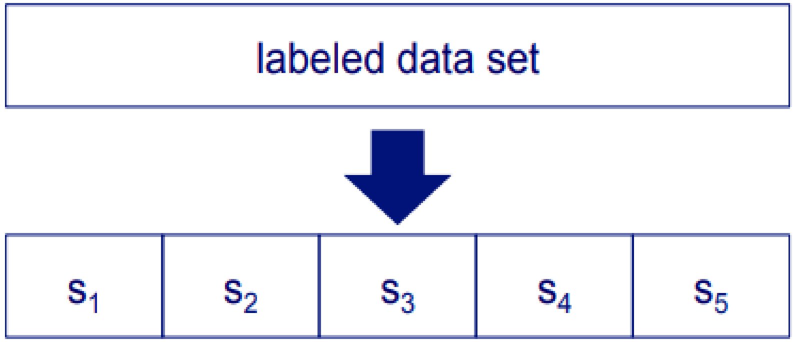
- This process results in k estimates of the test performance, $Perfromance_1$,...., $Perfromance_{\mathrm{k}}$. The k-fold CV estimate is computed by averaging these values.

$$CV_{(k)} = \frac{1}{k}\sum_{i=1}^{k} Performance_i$$



Cross-Validation Continued’.

partition data
into *n* subsamples



iteratively leave one subsample out for the test set, train on the rest

iteration	train on	test on
1	s_2 s_3 s_4 s_5	s_1
2	s_1 s_3 s_4 s_5	s_2
3	s_1 s_2 s_4 s_5	s_3
4	s_1 s_2 s_3 s_5	s_4
5	s_1 s_2 s_3 s_4	s_5

Suppose we have 100 instances, and we want to estimate accuracy with cross validation

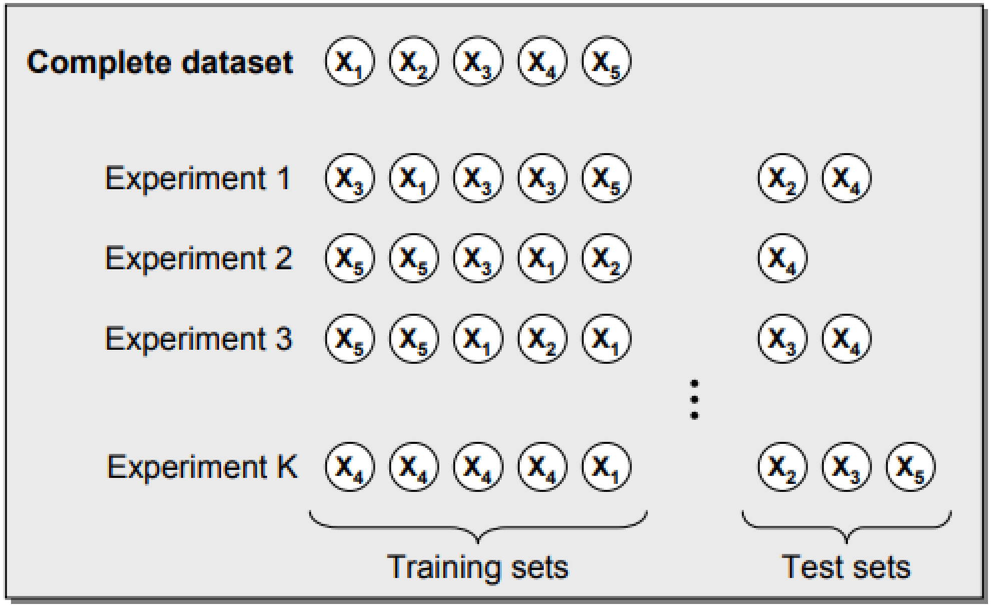
iteration	train on	test on	correct
1	s_2 s_3 s_4 s_5	s_1	11 / 20
2	s_1 s_3 s_4 s_5	s_2	17 / 20
3	s_1 s_2 s_4 s_5	s_3	16 / 20
4	s_1 s_2 s_3 s_5	s_4	13 / 20
5	s_1 s_2 s_3 s_4	s_5	16 / 20

accuracy = 73/100 = 73%

Bootstrapping

- The bootstrap is a resampling technique with replacement n from a dataset with N examples.
- Randomly select (with replacement) N examples and use this set for training.
- The remaining examples that were not selected for training are used for testing.
- This value is likely to change from fold to fold.
- Repeat this process for a specified number of folds (K).
- As before, the true error is estimated as the average error rate on test examples

Cont.



	Training sets	Test sets
Complete dataset	x_1 x_2 x_3 x_4 x_5	
Experiment 1	x_3 x_1 x_3 x_3 x_5	x_2 x_4
Experiment 2	x_5 x_5 x_3 x_1 x_2	x_4
Experiment 3	x_5 x_5 x_1 x_2 x_1	x_3 x_4
⋮	⋮	⋮
Experiment K	x_4 x_4 x_4 x_4 x_1	x_2 x_3 x_5

- Compared to basic cross-validation, the bootstrap increases the variance that can occur in each fold.

Bias and variance

- A supervised learning model performance can help us to identify or even quantify overfitting or underfitting.
- The difference between the actual values and predicted values to evaluate the model, such prediction error can in fact be decomposed into three parts:
- Bias: The difference between the average prediction of our model and the correct value which we are trying to predict.

Cont.

- Model with high bias pays very little attention to the training data and oversimplifies the model. It always leads to high error on training and test data.
- Variance: Variance is the variability of model prediction for a given data point or a value which tells us how uncertainty our model is.
- Model with high variance pays a lot of attention to training data and does not generalize on the data which it hasn’t seen before.
- As a result, such models perform very well on training data but has high error rates on test data.
- Noise: Irreducible error that we cannot eliminate.

Review questions

1) What is model complexity? Explain the methods for reducing model complexity.

2) What is the difference between overfitting and underfitting models?

3) Discuss the concept of model tuning.

4) What is bootstrapping?

5) Describe the different methods of controlling model complexity.

6) How do you identify overfitting, and underfitting in a classification problem?

7) What is the cause of overfitting.

8) Why is overfitting called high variance?

9) How can we reduce the bias and variance of a classification model?

10) Differentiate between accuracy and confusion matrix.