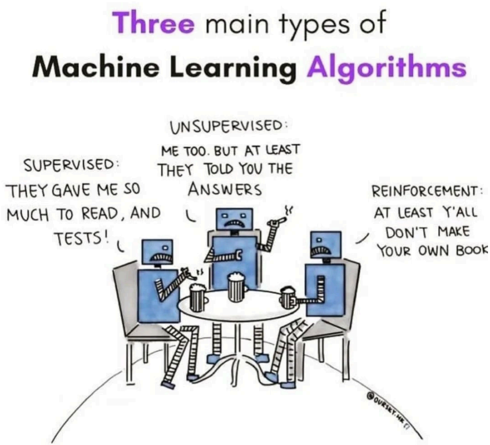
Introduction to Machine Learning

曹迥仪

清华大学计算社科平台

1. Supervised Learning
2. Coding with Python


Three main types of
Machine Learning Algorithms
SUPERVISED: THEY GAVE ME SO MUCH TO READ, AND TESTS!
UNSUPERVISED: ME TOO. BUT AT LEAST THEY TOLD YOU THE ANSWERS
REINFORCEMENT: AT LEAST Y'ALL DON'T MAKE YOUR OWN BOOK
@OURSKY.HK

1. Supervised Learning
2. Coding with Python

Supervised Learning

Training set: a set of labeled examples (samples, observations) of the form $(x_1, y_1), (x_2, y_2), ..., (x_i, y_i), ..., (x_n, y_n)$ where $x_i = (x_{i1}, ..., x_{ip})$are vectors of input variables (covariates,predictors,features) and y is the output.

What to learn: A function f $:\mathcal{X}_1 \times \mathcal{X}_2 \times \cdot\ \cdot\ \cdot \times \mathcal{X}_p \longrightarrow \mathcal{Y}$ which maps the input variables into the output domain.

$$y_i = f(x_i) + \epsilon_i$$

The goal is to learn f from n examples

- f(x): the part of Y to learn form X, the signal.
- ϵ: the part of Y you don't want to learn from X, the noise.

How do we estimate $f()$?

First we identify the problem

Two types of problems:

1. Regression: Y is (continuous) numerical, (eg. height,price etc.)
2. Classification:Y is (discrete) categorical, (eg. text classification,spam)

Then choose models.

Two general methodological approaches:

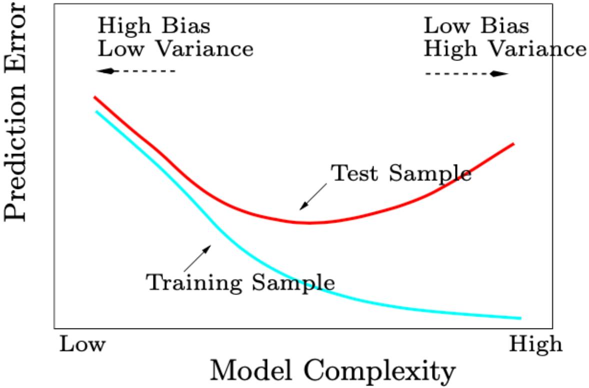
1. parametric models:
   eg. linear regression,logistic regression
2. non-parametric models：
   eg. decision tree, random forest, KNN

Remark

Some methods work for regression or classification only, some works for both. Different methods have their strengths and weaknesses (explainable vs. accuracy, works well on specific data)

$f(\cdot)$	Regression	Classification
Linear Regression	✓	
Logistic Regression		✓
K-nearest neighbor	✓	✓
Naive Bayes		✓
Decision Tree	✓	✓
Random Forest	✓	✓
SVM		✓

Bias Variance Trade-off



Training Error vs. Test Error

The training error can be easily calculated by applying the statistical learning method to the observations used in its training.

The test error is the average error that results from using a machine learning model to predict the response on a new observation, one that was not used in training the model.

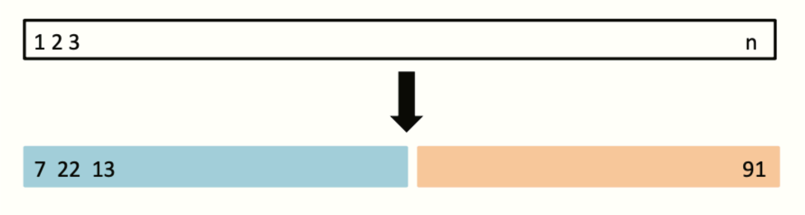
The training error rate often is quite different from the test error rate, and in particular the former can dramatically underestimate the latter.

$$y_i = f(x_i) + \epsilon_i$$

The Goal: We want $\hat{f}$ Complex enough to find the signal, but not so complex that chase the noise in the training data. Only signal will help you predict new Y given new X.

Methods to Assess Predictive Performance

Validation-set Approach

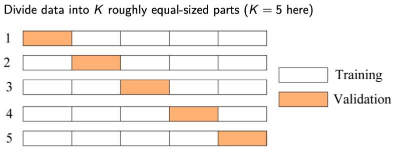


A random splitting into two parts: left part is training set, right part is validation set

The model is fitted on the training set, and the fitted model is used to predict the responses for the observations in the validation set.

Methods to Assess Predictive Performance

K-Fold Cross-validation


Divide data into K roughly equal-sized parts ($K = 5$ here)

Algorithm

1. Randomly divide the data into K equal-sized parts.
2. Leave out part k, fit the model to the other $K-1$parts (combined), and then obtain predictions for the left-out kth part.
3. Repeat for k = 1,2,...,K and then results are combined.

- can be used to tune parameter for a model. eg. what k to choose for knn.
- can be used to compare and select best model.

1. Supervised Learning
2. Coding with Python