The Elements of Statistical Learning

Hastie, Tibshirani, and Friedman (2009). The Elements of Statistical Learning. Second Edition. Springer.

1 Introduction

Statistical learning plays a key role in science, finance, industry, and many more areas. This book is about learning from data: supervised learning (presence of outcome variable for learning, the focus of this book) and unsupervised learning (outcome variable is unobserved).

Running examples:

- Classification of spam emails
- Explaining prostate specific antigen from clinical measurements via regression
- Classification of handwritten digits
- Clustering of DNA microarray data for cancer diagnostic

2 Overview of Supervised Learning

Two simple but powerful prediction methods are least squares linear models and k-nearest neighbors. The former makes huge assumptions about structure and yields stable but possibly inaccurate predictions (low variance, high bias), the latter makes very mild assumptions with often accurate but unstable predictions (at least if k is low, leading to low bias and high variance).

Local methods like k-nearest neighbors suffer from the curse of dimensionality: in high dimensions, samples only sparsely populate the input space and are close to an edge (extrapolation might be required). By imposing restrictions on the model class (e.g., linear models), this can be avoided. Many models have been proposed that lie in the spectrum between rigid model assumptions and flexibility, they will be presented in the book.

3 Linear Methods for Regression

Linear regression models are simple and often adequate and interpretable. The Gauss-Markov theorem states that the least squares estimates have the smallest variance among all linear unbiased estimates. However, it might be a good idea to trade a little bit bias for a large reduction in variance. This is possible with different variable subset selection and shrinkage methods:

- Best-subset selection: For each given size, find the best subset of variables that minimizes the residual sum of squares. Strategies for choosing the size will be discussed later. Not applicable for a large number of variables.
- Forward- and backward-stepwise selection: Searching through all possible subsets quickly becomes infeasible. Instead, sequentially add or remove variables.
- Forward-stagewise regression: A more constrained version of forward-stepwise, with benefits in high dimensions.

- Ridge regression: The idea is to make the selection process continuous by shrinking the coefficient values, which can further reduce variance. The penalty for coefficient sizes is quadratic (L_2). Variables with small variance are shrunk the most.
- The lasso (least absolute shrinkage and selection operator): Similar to ridge regression, but the penalty is in absolute coefficient value (L_1). This can make some coefficients to be exactly zero.
- Elastic-net penalty: A convex combination of the ridge and the lasso penalty.
- Least angle regression: Similar to forward-stepwise regression with the difference, that entering variables are not fit completely but only until it no longer has the highest correlation with the current residual. A slight modification provides an efficient way of computing the entire lasso path.
- Principal components regression: Use transformed variables with large sample variance.
- Partial least squares: Also constructs a set of linear combinations of the inputs for regression, but unlike principal components regression it uses the dependent variable for construction.

4 Linear Methods for Classification

The goal is to classify inputs into a finite number of categories. This means dividing the input space into a collection of regions. Linear here means that the regions are separated by linear boundaries.

- One option is linear regression of inputs to indicators of the response. When there is a large number of classes, classes can be masked by others. That means, that the predicted regression value never dominates. This can be avoided by adding polynomial terms to the regression equation.
- Another option is linear discriminant analysis, where each class density is modeled as a multivariate Gaussian with constant error covariance. With two classes, this is the same as linear regression of the class indicators. If the error covariance is not constant but class-dependent, this yields quadratic discriminant functions. The difference in covariances can be regularized. In the linear case, since only the relative differences to the class centroids matter, the data can be projected in a subspace of dimension at most number of classes minus 1, which can be a significant drop in dimension. This subspace can be further decomposed in term of centroid separation. By choosing an optimal subspace dimension, this projection can also be used for classification.
- Logistic regression updates the regression approach by ensuring that the dependent variables are proper probabilities. The lasso penalty can be used for variable selection.
- Separating hyperplane classifiers construct linear decision boundaries that explicitly try to separate the data into different classes as well as possible by minimizing the distance of misclassified points to the decision boundary. The optimal separating hyperplane maximizes the distance to the closest point from either class, which provides a unique solution with generalizes better.

5 Basis Expansions and Regularization

To transcend the constraints of linear models, one can enhance the input vector by applying transformations to it and then utilize linear models in the resulting expanded input space. One approach is to use piecewise polynomials, also known as splines, which involve dividing the input domain into contiguous intervals and fitting a separate polynomial in each interval. However, it is crucial to determine the appropriate polynomial order, number of knots, and their placement. The B-spline basis is a convenient way to represent them numerically. By using a maximal set of knots and employing regularization, the knot selection problem can be avoided, and complexity can be controlled. Another alternative is to use wavelets, which use a complete orthonormal basis to represent the function, but selectively shrink and choose the coefficients for a sparse representation. This technique is particularly useful for signal compression.

6 Kernel Smoothing Methods

For greater flexibility, one can fit a different, yet simple model at each query point separately by utilizing data from the closest observations. To ensure that the resulting regression function is smooth, one can apply kernels that assign weights to observations that decay smoothly with distance from the target point. These kernels are typically parameterized to dictate the width of the neighborhood; examples include the Epanechnikov kernel, tri-cube kernel, or Gaussian density (with non-compact support). To avoid bias on the boundary of the domain due to asymmetry of the kernel in that region, one can fit straight lines or higher-order polynomials instead of constants. However, there is a trade-off between reducing bias and increasing variance. The kernel smoothing technique generalizes naturally to multiple dimensions, although boundary effects become a more significant issue. Additionally, kernel density estimation is an unsupervised learning procedure that can be utilized for classification.

7 Model Assessment and Selection

Models generalize well if they have a good prediction capability on independent test data. To decide between competing models, a reasonable criterium is their generalization.

The prediction error of a model has three sources: error in the data generating process which we cannot avoid, squared bias which is the amount by which the average of our estimate differs from the true parameter, and the variance of our estimates from the truth for different data. Typically, the more complex the model, the lower the bias, but the higher the variance.

Prediction error, however, always depends on a specific loss function and can be behave differently for, e.g., the squared-error loss versus the 0-1 loss.

Prediction error on the training sample is not a good estimate of the generalization error, since it consistently decreases with model complexity. However, a model with zero training error is overfit to the training data and will typically generalize poorly.

We need another method for estimating the expected test error for a model. Typically, a model has a tuning parameter, and we seek to find the optimal parameter.

In a data-rich situation, one approach is dividing the data into training, validation, and test set (rule of thumb is 50%, 25%, 25%). But for insufficient amount of data, this does not work.

One option are information criteria. The AIC estimates the in-sample prediction error, assuming a log-likelihood loss function. One ingredient are the number of model parameters. This concept needs to be generalized to the effective number of parameters in case of regularization. Another option is the BIC, which penalizes model complexity by the factor $\log(N)$ and is motivated from a Bayesian approach of model selection, namely Bayes factors. There is no clear choice between AIC and BIC for model selection: BIC is consistent but chooses too simple models in the finite sample case.

Cross-validation

Bootstrap

8 Model Inference and Averaging

9 Additive Models, Trees, and Related Methods

10 Boosting and Additive Trees

11 Neural Networks

12 Support Vector Machines and Flexible Discriminants

13 Prototype Methods and Nearest-Neighbors

14 Unsupervised Learning

15 Random Forests

A random forest is an average of trees. The hope is to reduce variance. The bias in the average remains the same. Trees are high-variance and low-bias procedures, so this averaging of trees fitted to bootstrap-sampled versions of the training data (bagging) is beneficial. The averaging reduces more variance if the trees are uncorrelated, so the idea for random forests is also to reduce the correlation between the single trees. This is achieved by random selection of input variables in the tree-growing process. We must be careful because this procedure increases the variance again.

16 Ensemble Learning

Ensemble learning means to build a prediction model by combining the strengths of a collection of simpler models (e.g., random forest).

17 Undirected Graphical Models

18 High-Dimensional Problems