Elements of Statistical Learning

book by Hastings, Tibshirani, and Friedmann

Chapter 2

Overview of Supervised Learning

2.1 Notation

- Input variable typically denoted by X.
- if X is vector, its components accessed by subscript X_j
- upper case refers to generic/abstract variable. Observed values are written with lowercase: i.e. ith observed value is x_i (which again can be scalar or vector)
- Matrices represented by bold upper case $\mathbf{X}$

  example: a set of N input p-vectors x_i where $i = 1..N$, is represented as the $N \times p$ matrix X
- In general vectors are not bolded, except when they have N components. This distinguishes a p-vector of inputs x_i for the ith observation from the N-vector $\mathbf{x_j}$ consisting of all observations of variable X_j.
- All vectors are assumed to be column vectors. Hence the ith row of $\mathbf{X}$ is x_i^T.

2.2 Two Simple Approaches to Prediction: Least Squares and Nearest Neighbors

2.2.1 Linear Model

- Given a vector of inputs $X^T = (X_1, \cdots, X_p)$, we predict the output Y via

$$\hat{Y} = \beta_0 + \sum_{j=1}^{p} X_j \hat{\beta}_j$$

- often convenient to include the **bias** (aka intercept) term β_0 into X by including a constant variable 1 in X. Then letting $\hat{\beta}$ be the vector of coefficients including the bias, we can write the linear model in vector form as the inner product

$$\hat{Y} = X^T \hat{\beta}$$

- in general, $\hat{Y}$ could be a k-vector (i.e. the output is not scalar valued), in which case $\hat{\beta}$ would be a $p \times K$ matrix of coefficients.
- most popular approach to fit the linear model is the method of **least squares**: Pick the coefficients $\hat{\beta}$ to minimize the residual sum of squares:

$$RSS(\beta) := \sum_{i=1}^{n} (y_i - x_i^T \beta)^2$$

  in matrix notation

$$RSS(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T (\mathbf{y} - \mathbf{X}\beta)$$

  for nonsingular $X^T X$ solution is

$$\hat{\beta} = \left(X^T X\right)^{-1} X^T Y$$

2.3 Nearest-Neighbor Methods

- The k-nearest neighbor fit for $\hat{Y}$ is defined as follows:

$$\hat{Y} = \frac{1}{k} \sum_{x_i \in N_k(x)} y_i$$

  where $N_k(x)$ is the neighborhood of x defined by the k closest points x_i in the training sample.

  requires metric to define closest - typically euclidean

  No training required! That is no parameters are fit.

- while k-NN appears to have a single parameter k, in truth it has N/k effective parameters, where N is data size.

  this is generally much larger than linear model parameters and thus requires much more data.

  heuristic: if you have nonoverlapping clusters ok k points, then you have N/k such nhoods, and thus need N/K parms (the means) to describe result/fit.

- cannot use RSS on training to pick k, because it would always pick $k = 1$, which leads to zero training error.

2.4 Statistical Decision Theory

- let $X \in \mathbb{R}^p$ be real valued random input vector, $Y \in \mathbb{R}$ be real valued randome ouput, with joint distribution $Pr(X, Y)$
- we seek function $f(X) : \mathbb{R}^p \to \mathbb{R}$ for prediting Y value
- for this we need a **loss function** $L(Y, f(X))$ for penalizing errors in prediction
- most common loss function is: **squared error loss**

$$L(Y, f(X)) = (Y - f(X))^2$$

- criterion for choosing f: minimize the expected prediction error (EPE)

$$EPE(f) = E(Y - f(X))^2 = \int [y - f(x)]^2 Pr(dx, dy)$$

- He presents some short argument that via conditioning on X and pointwise minimization, you can arrive at the solution

$$f(x) = E(Y|X = x),$$

  that is, the *conditional expecation*, aka the **regression function**.

- I am interested in how to rigorously solve this - perhaps a constrained variational calculus problem?
- KNN is an approximation to this, where instead of relying only on observations at x exactly, a neighborhood of x is use to obtain the expected value of y.
- for linear regression - we propose ansatz $f(x) \sim x^T \beta$ - so don't search over all functions

  this is a model based approach

  theoretical solutions is:

$$\beta = \left[E(XX^T)\right]^{-1} E(XY)$$

- Using L1 instead of L2 as loss function leads to $\hat{f}(x) = median(Y|X = x)$
- For categorical output, loss function is matrix $K \times K$ matrix L, where K is number of categories. Zero in the diagonals, and nonnegative elsewhere

  typical is **zero-one loss function** - where all off diagonals are 1.

- The $EPE = E[L(G, \hat{G}(x)]$ where expectation is taken over $Pr(G, x)$
- using same conditioning argumetn and pointwise minimization, and zero one loss, leads to solution known as the **Bayes classifier**

$$\hat{G} = g_k \quad \text{if} \quad Pr(g_k|X = x) = max_{g \in G} Pr(g|X = x)$$

  that is, classify as the most probable class, using discrete conditional distribution $Pr(G|X)$.

  Error rate of the bayes classifier is often called the **Bayes rate**.