**2.3 Partial dependence plots**

Harrison and Rubinfeld (1978) analyzed a data set containing suburban Boston housing data from the 1970 census that now permeates statistics literature. They sought a housing value equation using the assortment of variables provided; see Harrison and Rubinfeld (1978, Table IV) for a description of each variable. The assumptions usually assumed for any regression-based analysis, such as normality, linearity, and constant variance, were clearly violated, but through an exhausting series of transformations, significance testing, and grid searches, they were able to build a model which fit the data reasonably well (R2 = 0.81). Their prediction equation is given in Equation (1).

EQ 1

Using modern computing power, many supervised learning algorithms can fit large data sets in seconds, producing powerful, highly accurate models. The downfall of many of these machine learning algorithms, however, is decreased interpretability. Unlike prediction formulas such as Equation (1), many machine learning algorithms are left attempting to convey some relative measure of variable importance which provide insight, but cannot match the simplicity of Equation (1) and others like it.

Tree-based methods naturally assign variable importance scores, but to gain even more insight, one can construct a *partial dependence plot* (PDP); see Friedman (2001) for details. Not only do PDPs rank the variables in terms of importance, but can also visually convey the relationship between variables in your model. “Black box” models such as random forests and support vector machines methods that are particularly conducive to using PDPs, which are mathematically rooted as follows:

Let represent the predictors in a model whose prediction function is . If we partition into an interest set, , and its complement, , then the “partial dependence” of the response on is defined as

EQ 2

Where is the marginal probability density of . Equation (2) can be estimated from a set of training data by

EQ 3

Where are the values of that occur in the training sample; that is, we average out the effects of all the other predictors in the model.

Constructing a PDP (3) in practice is rather straightforward. To simplify, let be the predictor variable of interest with unique values . The partial dependence of the response can be constructed as follows:

ALGORITHM 1

Algorithm (1) can be computationally intensive since it involves *k* passes over the training records, but fortunately, is easily parallelizable. In addition, this method can be extended to larger subsets of two or more features.