A.2.1. Less is More: The Case for *Lasso* and *Ridge*

Lasso and *ridge* are both linear models that use a regularization term to achieve a balance between bias and variance. They do so by minimizing a loss function that includes in-sample fit and a penalty term that favors simple models, thereby reducing variance. Prediction accuracy is thus improved by setting some coefficients to zero and shrinking others. To achieve this goal, lasso and ridge combine the minimization of the sum of the squared errors with the norm of parameters. The lasso estimator solves the problem:

$$\min_{\beta} \sum_{j=1}^{k} (y_i - x_i\beta)^2 \; + \; \lambda \cdot \|\beta\|_1$$

where $\|\beta\|_1$ is the ℓ_1-norm (least absolute deviation). The penalty weight (λ) on the sum of the absolute values of coefficients is set using the default parameter in scikit-learn[25].

Ridge is similar to *lasso* except that the bound on the parameter estimates is the ℓ_2-norm (least squares), therefore shrinking estimates smoothly towards zero, as opposed to setting some estimates to zero as Lasso does.[26]

A.2.2. Gradient Boosting Trees

Gradient Boosting Trees are related to random forests. A decision tree is the basic building block of random forests. A decision tree defines a tree-shape flow graph to support decisions. An instance is classified by starting from the root of the tree, testing the feature specified by the node, moving down the branch corresponding to the feature value in the given instance.

A key difference between decision tree learning and Ridge and Lasso regression lies in the fact that there is no explicit objective function that a decision tree optimizes. Instead, the learning process is a greedy recursive algorithm that finds the best feature to split the current data based on a criterion. In our paper, we_use a decision tree regressor

[25] http://scikit-learn.org/stable/

[26] For a detailed discussion of sparse estimators, we refer interested readers to Hastie, Tibshirani and Wainwright (2015).