Tree-based Models for Regression

- Regression trees
- Regression forests
  - `randomForest` based on bagging
  - `gbm` based on boosting

Bagging (Bootstrap Aggregation)

- Training data: $\mathbf{Z} = \{(x_i, y_i)_{i=1}^n\}$
- Bootstrap samples[a]: $\mathbf{Z}^{*b} = \{(x_i^{*b}, y_i^{*b})_{i=1}^n\}$, where $b = 1 : B$

$$
\begin{array}{lcl}
\mathbf{Z}^{*1} & : & (x_1^{*1}, y_1^{*1}),\ (x_2^{*1}, y_2^{*1}),\quad \cdots,\quad (x_n^{*1}, y_n^{*1}) \\
\mathbf{Z}^{*2} & : & (x_1^{*2}, y_1^{*2}),\ (x_2^{*2}, y_2^{*2}),\quad \cdots,\quad (x_n^{*2}, y_n^{*2}) \\
\vdots & & \\
\mathbf{Z}^{*B} & : & (x_1^{*B}, y_1^{*B}),\ (x_2^{*B}, y_2^{*B}),\quad \cdots,\quad (x_n^{*B}, y_n^{*B})
\end{array}
$$

[a] sample with replacement from $\mathbf{Z}$.

Bagging (Bootstrap Aggregation)

- Training data: $\mathbf{Z} = \{(x_i, y_i)_{i=1}^n\}$
- Bootstrap samples[a]: $\mathbf{Z}^{*b} = \{(x_i^{*b}, y_i^{*b})_{i=1}^n\}$, where $b = 1 : B$
- $\hat{f}^{*b}$: classification/regression function trained by $\mathbf{Z}^{*b}$
- The bagging estimate is defined to be

$$\hat{f}_{\mathsf{bag}}(x) = \frac{1}{B} \sum_{b=1}^{B} \hat{f}^{*b}.$$

- Advantage: reduce variance. So works well for high-variance, low-bias procedures, such as trees.

[a] sample with replacement from $\mathbf{Z}$.

Random Forest

1. For $b = 1 : B$:

   (a) Draw a bs sample $\mathbf{Z}^{*b}$ from the training data.

   (b) Grow a BIG tree T_b (with some restriction).

2. Output the forest $\{T_b\}_{b=1}^{B}$.

To make a prediction at a new point x

Regression: $\frac{1}{B}\sum T_b(x)$.

Restriction when growing a tree in the forest:

- At each split, randomly select m variables from the p variables, and then pick the best split among them.
- The recommended value for m is $\sqrt{p}$ for classification and $p/3$ for regression.
- Purpose: reduce the correlation between trees in the forest.