Boosting II

Model-Based Boosting

Model-based Boosting

Boosting with regression models?

- Properties of OLS regression
  - The residuals e and X are uncorrelated
  - OLS finds the minimum SSE given a linear functional form
- There is nothing left to boost!

Component-wise/ model-based boosting

- Boosting with GLMs (linear, splines) as base learners
- Takes **small models** (components) at each iteration
- Returns an interpretable prediction function
- Performs variable selection and regularization

Model-based Boosting

Boosting with regression models?

- Properties of OLS regression
  - The residuals e and X are uncorrelated
  - OLS finds the minimum SSE given a linear functional form
- There is nothing left to boost!

Component-wise/ model-based boosting

- Boosting with GLMs (linear, splines) as base learners
- Takes **small models** (components) at each iteration
- Returns an interpretable prediction function
- Performs variable selection and regularization

Model-based Boosting

Boosting linear models[1]

1. Initialization
   - Specify a set of base learners $h_1(x_{i1}) = \beta_1 x_{i1}, \ldots, h_p(x_{i1}) = \beta_p x_{ip}$
   - Initialize model $\hat{f}^0 = \bar{y}$
2. For $m = 1$ *to* m_{stop}
   1. Compute residuals u_i^m based on the current model
   2. Fit each base-learner *separately* to the current residuals
   $$\mathbb{E}(u_i^m) = \hat{h}_1^m(x_{i1}), \ldots, \mathbb{E}(u_i^m) = \hat{h}_p^m(x_{ip})$$
   3. Select the best base learner h_{j^*}: $j^* = \text{argmin} \sum (u_i^m - \hat{h}_j^m(x_{ij}))^2$
   4. Update the current model (w. shrinkage ν): $\hat{f}^m(\mathbf{x}_i) = \hat{f}^{m-1}(\mathbf{x}_i) + \nu \hat{h}_{j^*}^m(x_{ij})$

[1] https://github.com/SocialScienceDataLab/statistical-boosting-with-mboost

Model-based Boosting

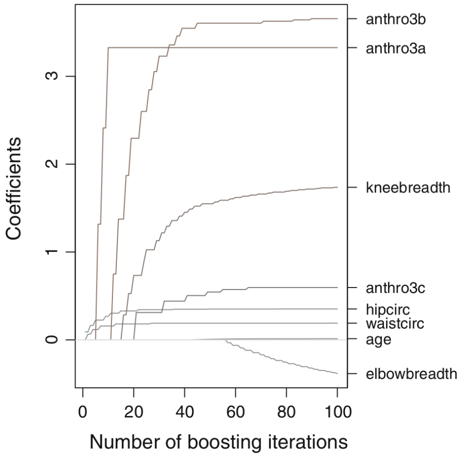
The final boosting estimate

$$\hat{f} = \hat{f}_1 + \cdots + \hat{f}_P$$

- Can be expressed as an additive function of the base learners
- Includes sums of individual estimates for each base learner
  - $\hat{f}_j$ equals sum of $\nu \hat{\mathbf{u}}^m$ over all iterations where base learner j was selected
  - Coefficients can be summed equivalently
- m_{stop} controls **variable selection** and **regularization**
  - Variables might not be selected up to m_{stop}
  - Shrinkage parameter ν causes coefficients to increase slowly

Variable selection and regularization

Figure: Coefficient paths in model-based boosting[2]



[2]Hofner et al. (2014)

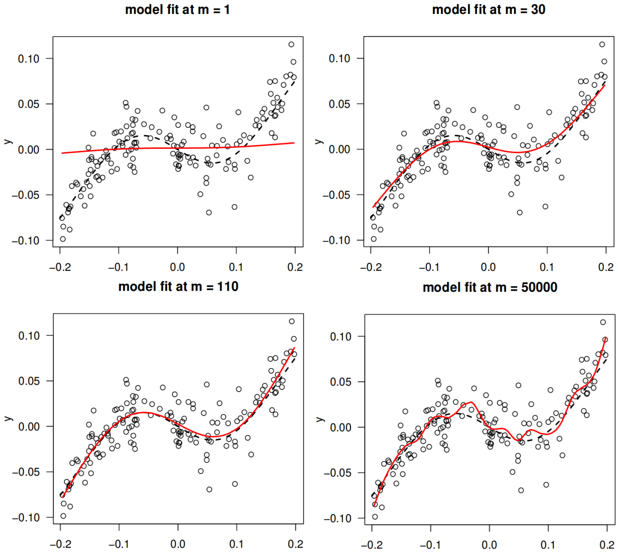
Base learners

Base learners in `mboost`

- `bols`
  - Linear base learner
- `bbs`
  - Smooth base learner: P-splines
- `bspatial`
  - Smooth base learner: bivariate P-splines
- `brandom`, `btree`, ...

Boosting splines

Figure: Boosting with a P-spline base-learner at different iterations[3]



[3] Mayr et al. (2014)

Grid search and random search

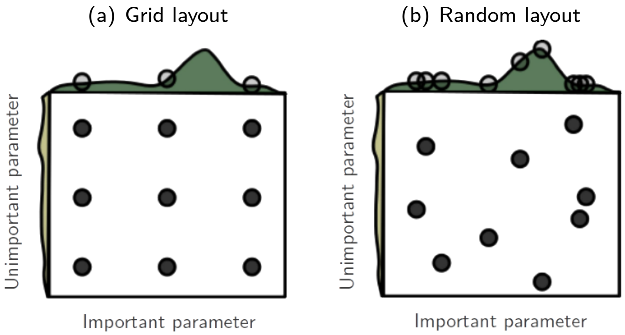
Grid search and random search

Tuning many hyperparameters

- (Exhaustive) Grid search
  - Expands a grid over all combinations of considered try-out values
  - Can become inefficient with many tuning parameters
- Random search (Bergstra & Benglio 2012)
  - Considers only a random selection of tuning parameter combinations
  - Benefit depends on method and implementation
- Adaptive search (Kuhn 2014)
  - Guided search by considering performance within the search process
  - Adaptive removal of unpromising parameter settings

Grid search and random search

Figure: Grid and random search with two tuning parameters[4]



[4] Bergstra and Benglio (2012)

Software Resources

Resources for R

- Extreme Gradient Boosting: `xgboost`
  - Competitive and scalable boosting approach (Chen & Guestrin 2016)
  - XGBoost Explainer: `https://www.rdocumentation.org/packages/xgboostExplainer/versions/0.1`
- Model-based Boosting: `mboost`
  - Implements various linear and non-linear base learners (Hofner et al. 2014)

References

Bergstra, J., Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research 13*, 281–305.

Chen, T., Guestrin, C. (2016). *XGBoost: A scalable tree boosting system.* https://arxiv.org/abs/1603.02754.

Hastie, T., Tibshirani, R., Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York, NY: Springer.

Hofner, B., Mayr, A., Robinzonov, N, Schmid, M. (2014). Model-based boosting in R: a hands-on tutorial using the R package mboost. *Computational Statistics 29(1-2)*, 3–35.

Kuhn, M. (2014). *Futility Analysis in the Cross-Validation of Machine Learning Models.* https://arxiv.org/abs/1405.6974.

Mayr, A., Binder, H., Gefeller, O., Schmid, M. (2014). *The evolution of boosting algorithms – from machine learning to statistical modeling.* https://arxiv.org/abs/1403.1452.