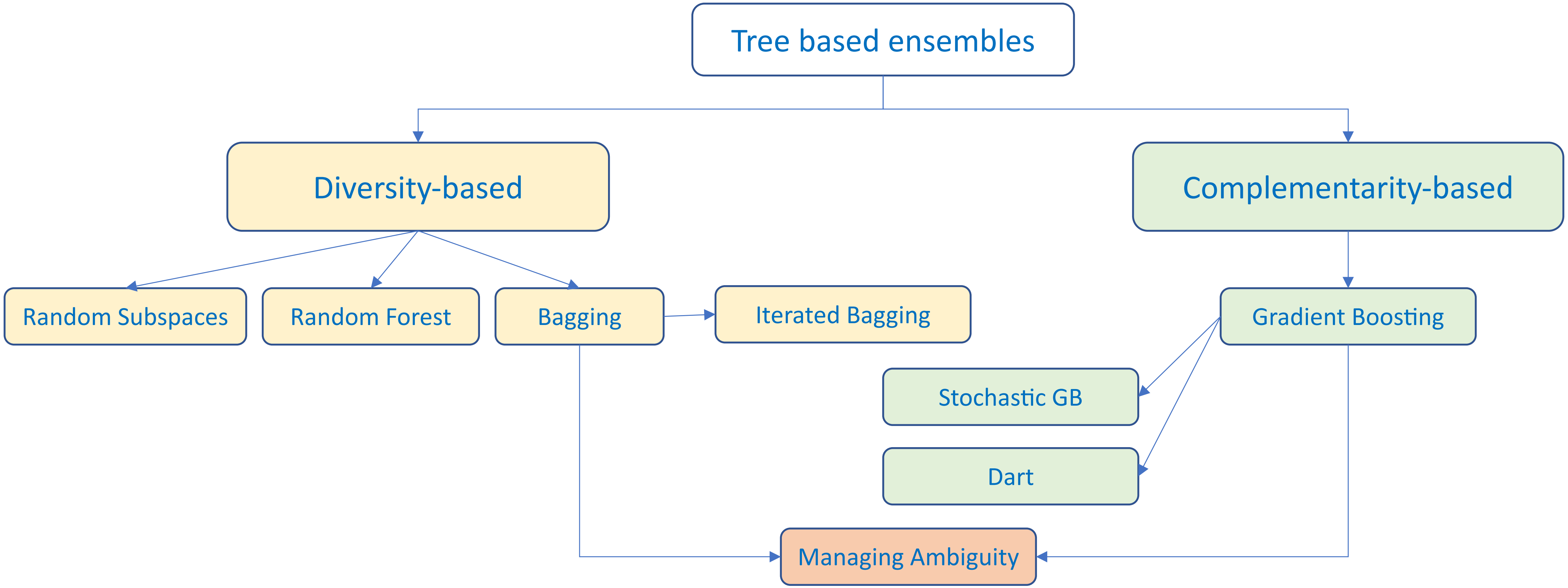
Managing Ambiguity in Regression Ensembles

Yuri Zelenkov

Graduate School of Business

HSE University

Ensemble taxonomy



Regression problem

Problem $h: X \to Y, \quad X \subset \mathbb{R}^k, \quad Y \subset \mathbb{R}$

Approximation f of h is obtained by applying a learning algorithm A with the hyperparameters θ to the training set $\mathcal{D}$, so $f = A(\mathcal{D}, \theta)$. The goodness of f is assessed by

$$L(y, f) = (y - f)^2$$

Let f is a convex ensemble $f_E = \sum_{i=1}^{M} w_i f_i \qquad \sum_{i=1}^{M} w_i = 1,\ w_i \geq 0,\ i = 1, \ldots, M$

Ambiguity Decomposition

Let f_E be an ensemble of M regressors f_i, $i = 1 \ldots M$, i.e. f_E is a convex combination of the individual estimators

$$f_E = \sum_{i=1}^{M} w_i f_i, \ \sum_{i=1}^{M} w_i = 1, \ w_i \geq 0, \ i = 1, \ldots, M.$$

According to (Krogh & Vedelsby , 1995), at an arbitrary single data point the quadratic error of the ensemble can be decomposed into two terms

$$\boxed{(f_E - y)^2 = \sum_{i=1}^{M} \underbrace{w_i (f_i - y)^2}_{\text{weighted error}} - \sum_{i=1}^{M} \underbrace{w_i (f_i - f_E)^2}_{\text{ambiguity}} \quad (1)}$$

Krogh, A. and Vedelsby, J. (1995) Neural network ensembles, cross validation, and active learning. In Advances in neural information processing systems, pp. 231–238.

Mathematical Formulation- 1

Suppose, we have an ensemble $f_E^{(M-1)}$ of $M-1$ trained estimators f_i and we want to add a new estimator f_M to minimize the total error:

$$f_M, w = \arg\min_{f,w}(y - f_E^M)^2$$

Let $w_i = 1/M$, thus $f_E^{(M)} = \frac{1}{M}\sum_{i=1}^{M} f_i = \frac{1}{M}\left(\sum_{i=1}^{M-1} f_i + f_M\right)$

So, the loss fucntion is

$$L(y, f_E^{(M)}) = \left(y - \frac{M-1}{M} f_E^{(M-1)} - \frac{1}{M} f_M\right)^2.$$

Mathematical Formulation - 2

$$L\left(y, f_E^{(M)}\right) = \left[y - \frac{M-1}{M} f_E^{(M-1)} - \frac{1}{M} f_M\right]^2$$

$$\frac{\partial L\left(y, f_E^{(M)}\right)}{\partial f_M} = -\frac{2}{M}\left(y - \frac{M-1}{M} f_E^{(M-1)} - \frac{1}{M} f_M\right) = 0$$

$$\boxed{t_M = My - (M-1) f_E^{(M-1)}}$$

As

$$f_E^{(M-1)} = \frac{1}{M-1} \sum_{i=1}^{M-1} f_i$$

Then, target to train f_M

$$\boxed{t_M = My - \sum_{i=1}^{M-1} f_i} \quad => \quad (y - t_M)^2 - \left(t_M - f_E^{(M)}\right)^2 = 0$$

Managing Ambiguity Algorithm

Algorithm 1 Managing Ambiguity Ensemble

Input: dataset $\{(x_i, y_i)\}_{i=1}^N$, number of iterations M

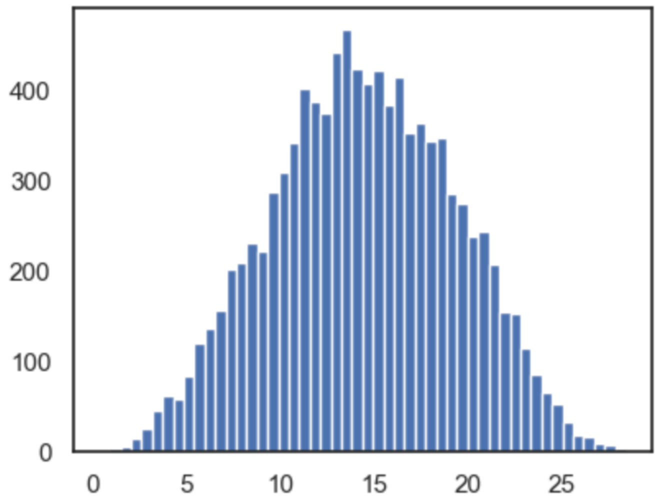
Fit an initial learner f_1 using training set $\{(x_i, y_i)\}_{i=1}^N$

for $m = 2$ **to** M **do**

1. Compute new targets $t_i^m = my_i - \sum_{j=1}^{m-1} f_{ji}$ for $i = 1, \ldots, N$.
2. Fit a base learner f_m using training set $\{(x_i, t_i^m)\}_{i=1}^N$.
3. Update the model $f_E = \frac{1}{m} \sum_{i=1}^m f_i$.

end for

Synthetic Dataset



Inputs X are independent features uniformly distributed on the interval [0, 1]. The output y is created according to the formula:

$$y = 10\sin(\pi x_1 x_2) + 20(x_3 - 0.5)^2 + 10x_4 + 5x_5 + noise * \mathbb{N}(0,1)$$

Out of the n features features, only 5 are actually used to compute y. The remaining features are independent of y.

```
sklearn.datasets.make_friedman1(n_samples=10000, n_features=20, noise=0.1, random_state=101)
```

TABLE I
OPTIMAL HYPERPARAMETERS OF THE COMPARED MODELS

MODEL	$L(y, f_E)$	M	l_r	d	S	F
GB	0.201 (0.011)	163	0.113	5	-	-
SGB	0.165 (0.007)	200	0.102	7	0.872	0.845
DART	0.138 (0.007)	166	0.249	9	-	-
SDART	0.115 (0.009)	164	0.297	6	0.883	0.834
BR	1.348 (0.071)	192	-	-	0.904	0.963
RF	1.302 (0.070)	157	-	-	0.999	0.804
MA	0.064 (0.004)	190	-	5	-	-

Comparison with Gradient Boosting

$$RLR_m = \frac{L(y, f_E^{m-1}) - L(y, f_E^m)}{L(y, f_E^{m-1})}$$

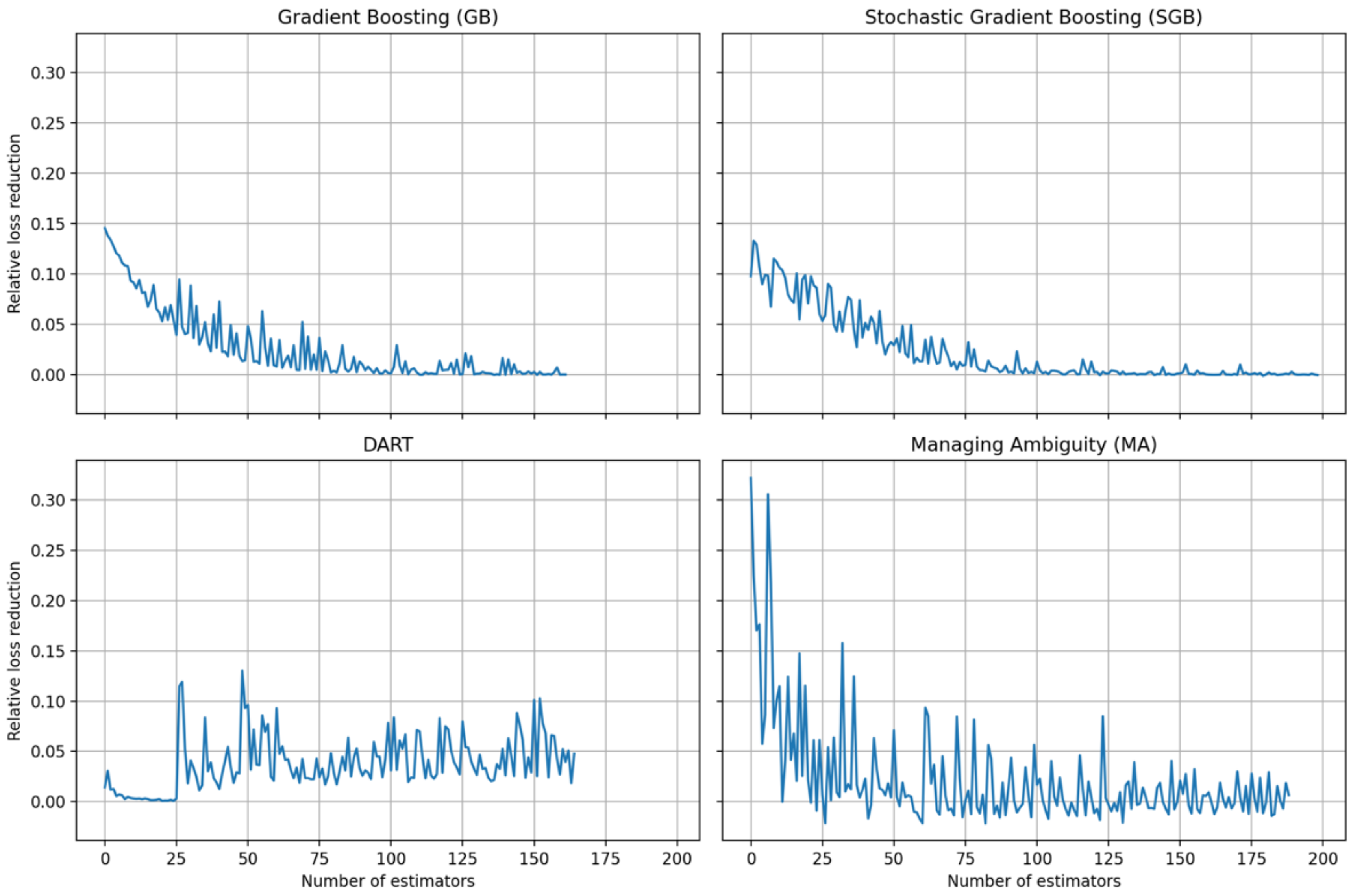


Fig. 1. The contribution of estimtors for different GB algorithms

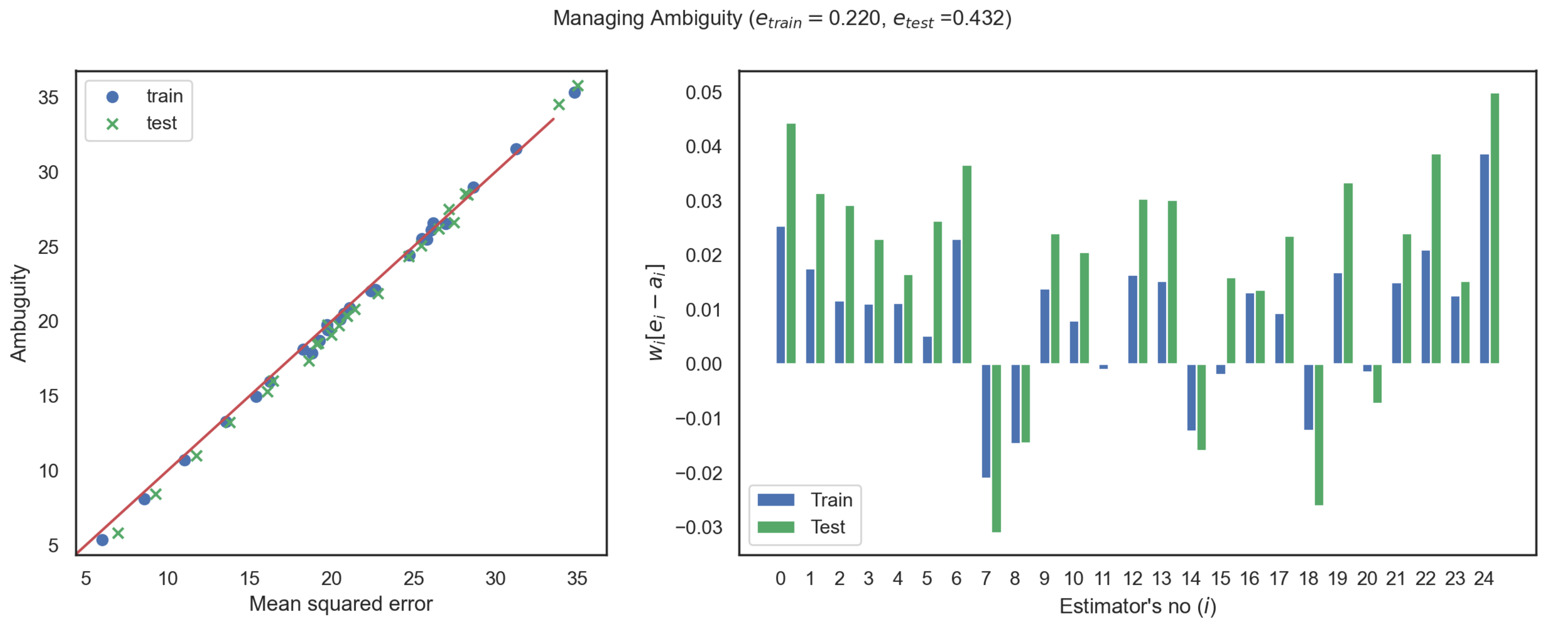
Comparison with Random Forest

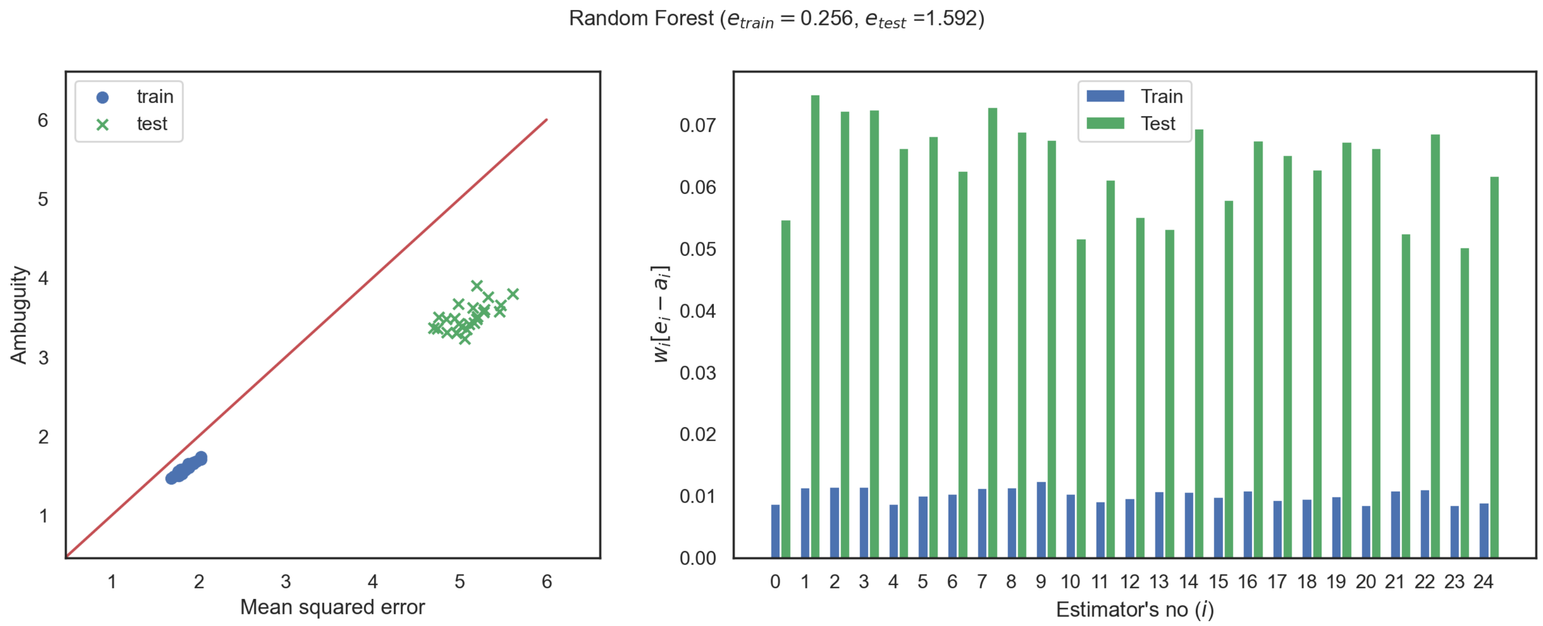
$$\frac{1}{N}\sum_{j=1}^{N}(f_{Ej}-y_j)^2=\frac{1}{N}\sum_{j=1}^{N}\sum_{i=1}^{M}w_i(f_{ij}-y_j)^2-\frac{1}{N}\sum_{j=1}^{N}\sum_{i=1}^{M}w_i(f_{ij}-f_{Ej})^2=$$

$$=\sum_{i=1}^{M}\left[\frac{1}{N}\sum_{j=1}^{N}(f_{ij}-y_j)^2-\frac{1}{N}\sum_{j=1}^{N}(f_{ij}-f_{Ej})^2\right].$$

$$\boxed{\begin{aligned} e_i &= \textstyle\sum_{j=1}^{N}(f_{ij}-y_i)^2/N \\ a_i &= \textstyle\sum_{j=1}^{N}(f_{ij}-f_{Ej})^2/N \end{aligned}}$$

```
train_test_split(test_size= 0.25)
```





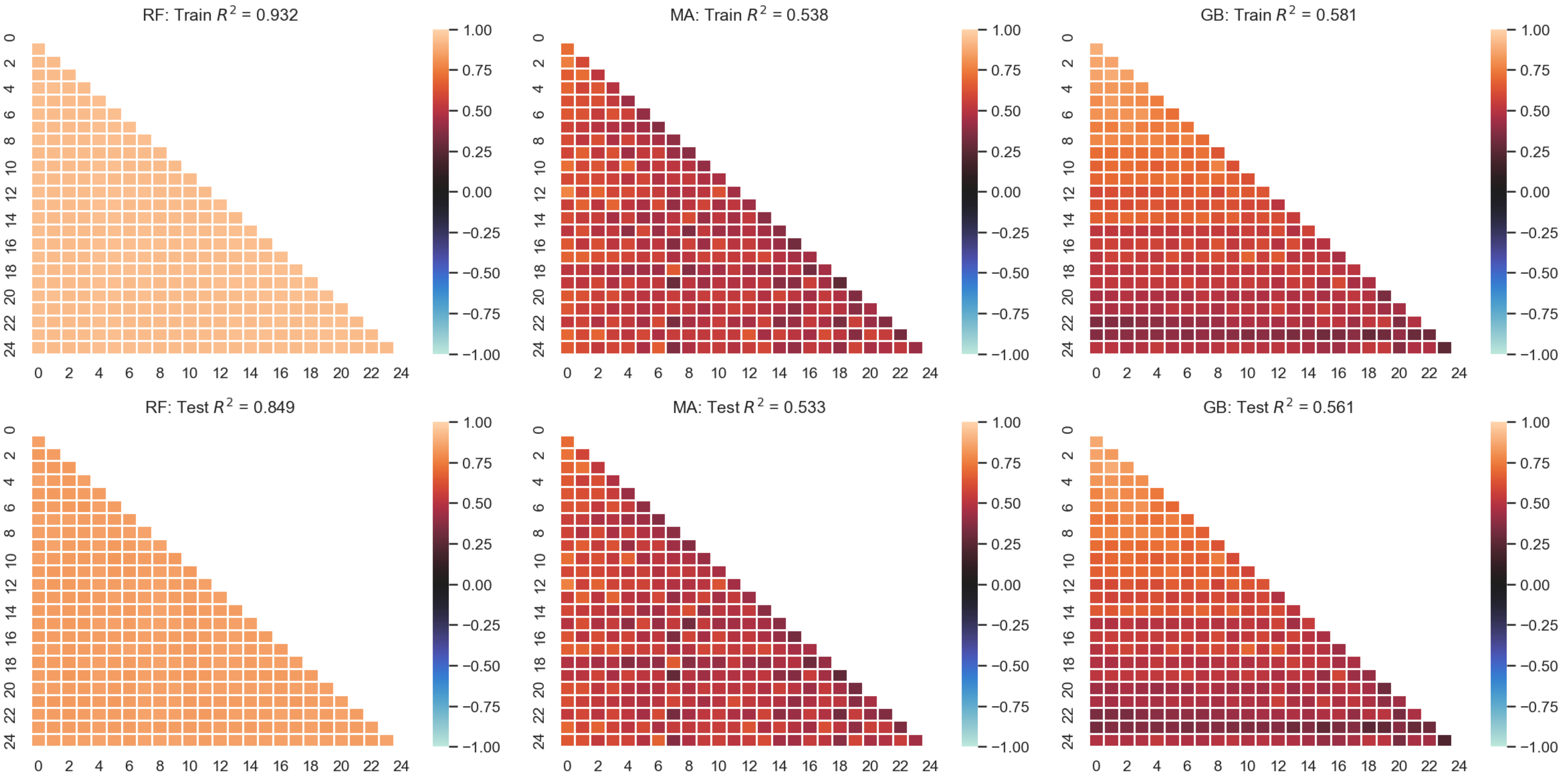

RF: Train R^2 = 0.932
MA: Train R^2 = 0.538
GB: Train R^2 = 0.581
RF: Test R^2 = 0.849
MA: Test R^2 = 0.533
GB: Test R^2 = 0.561

TABLE III

EXPERIMENT RESULTS.

DATASET	SAMPLES	FEATURES	SCALING	BR	GB		MA	
AIRFOIL	1503	3	1.0E+00	12.920(8.185)	11.361(7.240)	*	**10.612(6.773)**	*
AUTO	392	7	1.0E+00	8.280(5.776)	8.240(5.228)		**8.148(5.230)**	
BANK8FM	8192	8	1.0E+01	0.096(0.007)	**0.088(0.005)**	*◇	0.093(0.005)	
BIKE	17379	12	1.0E-02	0.382(0.178)	**0.284(0.109)**	*	0.300(0.135)	
BOSTON	506	13	1.0E+00	21.750(25.761)	**17.858(19.881)**		18.302(23.422)	
CADATA	20640	8	1.0E-05	0.494(0.222)	**0.400(0.165)**	*	0.420(0.153)	*
CART	40768	10	1.0E+00	1.293(0.026)	1.002(0.021)	*	**0.996(0.020)**	*◇
CARSEATS	400	10	1.0E+00	2.390(0.534)	1.700(0.419)	*	**1.391(0.281)**	*◇
CCPP	9568	4	1.0E+00	10.645(1.312)	**9.087(1.417)**	*◇	9.540(1.634)	*
CONCRETE	1030	8	1.0E-01	0.222(0.053)	0.157(0.054)	*	**0.138(0.055)**	*◇
EGRID	10000	12	1.0E+03	134.331(10.363)	73.125(5.553)	*	**47.264(2.691)**	*◇
ELEVATORS	16599	18	1.0E+03	8.224(2.187)	5.010(0.858)	*	**4.629(0.764)**	*◇
FACEBOOK	40949	53	1.0E+00	478.411(168.689)	477.693(201.241)		**471.570(199.857)**	
HOUSE	22784	16	1.0E-03	1009.607(108.110)	**999.840(91.138)**		1013.261(88.251)	
KIN8NM	8192	8	1.0E+01	1.919(0.096)	1.420(0.086)	*	**1.180(0.071)**	*◇
LASER	933	4	1.0E+00	67.709(93.933)	68.342(73.960)		**54.882(77.498)**	◇
SMARKET	1250	7	1.0E+01	138.414(108.073)	**129.260(103.732)**	*◇	140.175(107.545)	
STOCK	950	9	1.0E+00	8.797(8.449)	7.259(7.308)		**5.904(4.963)**	
TREASURY	1049	15	1.0E+00	0.056(0.030)	0.051(0.028)		**0.042(0.023)**	*
WANKARA	1609	9	1.0E+00	1.893(0.221)	1.836(0.185)		**1.655(0.224)**	*◇

Python Code

https://github.com/yzelenkov/Managing-Ambiguity

QUESTIONS?