Boosting

Hedibert F. Lopes & Paulo Marques
INSPER Institute of Education and Research
São Paulo, Brazil

The following slides are based on i) Hastie, Tibshirani and Friedman (2009) *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Second Edition, and ii) `http://jessica2.msri.org/attachments/10778/10778-boost.pdf`.

Using classification trees as a modeling structure

Bagging: Averaging trees

Random Forests: Cleverer averaging of trees

Boosting: Cleverest averaging of trees

Methods for improving the performance of weak learners such as Trees.

Classification trees are adaptive and robust, but do not generalize well.

The techniques discussed here enhance their performance considerably.

Properties of trees

✓ Can handle huge datasets

✓ Can handle mixed predictors?quantitative and qualitative

✓ Easily ignore redundant variables

✓ Handle missing data elegantly

✓ Small trees are easy to interpret

✗ large trees are hard to interpret

✗ Often prediction performance is poor

Model averaging

Classification trees can be simple, but often produce noisy (bushy) or weak (stunted) classifiers.

- Bagging (Breiman, 1996)
  Fit many large trees to bootstrap-resampled versions of the training data, and classify by majority vote.
- Boosting (Freund & Schapire, 1996)
  Fit many large or small trees to reweighted versions of the training data. Classify by weighted majority vote.
- Random Forests (Breiman 1999)
  Fancier version of bagging.

In general Boosting $\succ$ Random Forests $\succ$ Bagging $\succ$ Single Tree.

Boosting Methods

- *It was originally designed for classification problems, but . . . it can profitably be extended to regression as well.*
- *The motivation for boosting was a procedure that combines the outputs of many "weak" classifiers to produce a powerful "committee."*
- *From this perspective boosting bears a resemblance to bagging and other committee-based approaches.*

Weak classifier: error rate is only slightly better than random guessing.

Boosting: sequentially apply the weak classification algorithm to repeatedly modified versions of the data, thereby producing a sequence of weak classifiers $G_m(x),\ m = 1, \ldots, M$.

AdaBoost (Freund and Schapire, 1996)

1. Initialize the observation weights $w_i = 1/N,\ i = 1, 2, \ldots, N$.
2. For $m = 1$ to M repeat steps $(a) - (d)$:
   - (a) Fit a classifier $G_m(x)$ to the training data using weights w_i:
   - (b) Compute weighted error of newest tree

$$\text{err}_m = \frac{\sum_{i=1}^{N} w_i I(y_i \neq G_m(x_i))}{\sum_{i=1}^{N} w_i}.$$

   - (c) Compute $\alpha_m = \log[(1 - \text{err}_m)/\text{err}_m]$.
   - (d) Update weights for $i = 1, \ldots, N$:

$$w_i \leftarrow w_i \exp[\alpha_m I(y_i \neq G_m(x_i))]$$

3. The predictions from all of them are then combined through a weighted majority vote to produce the final prediction:

$$G(x) = \text{sign}\left\{\sum_{m=1}^{M} \alpha_m G_m(x)\right\}$$

At step m, those observations that were misclassified by the classifier $G_{m-1}(x)$ induced at the previous step have their weights increased, whereas the weights are decreased for those that were classified correctly. Thus as iterations proceed, observations that are difficult to classify correctly receive ever-increasing influence. Each successive classifier is thereby forced to concentrate on those training observations that are missed by previous ones in the sequence.

Schematic of AdaBoost

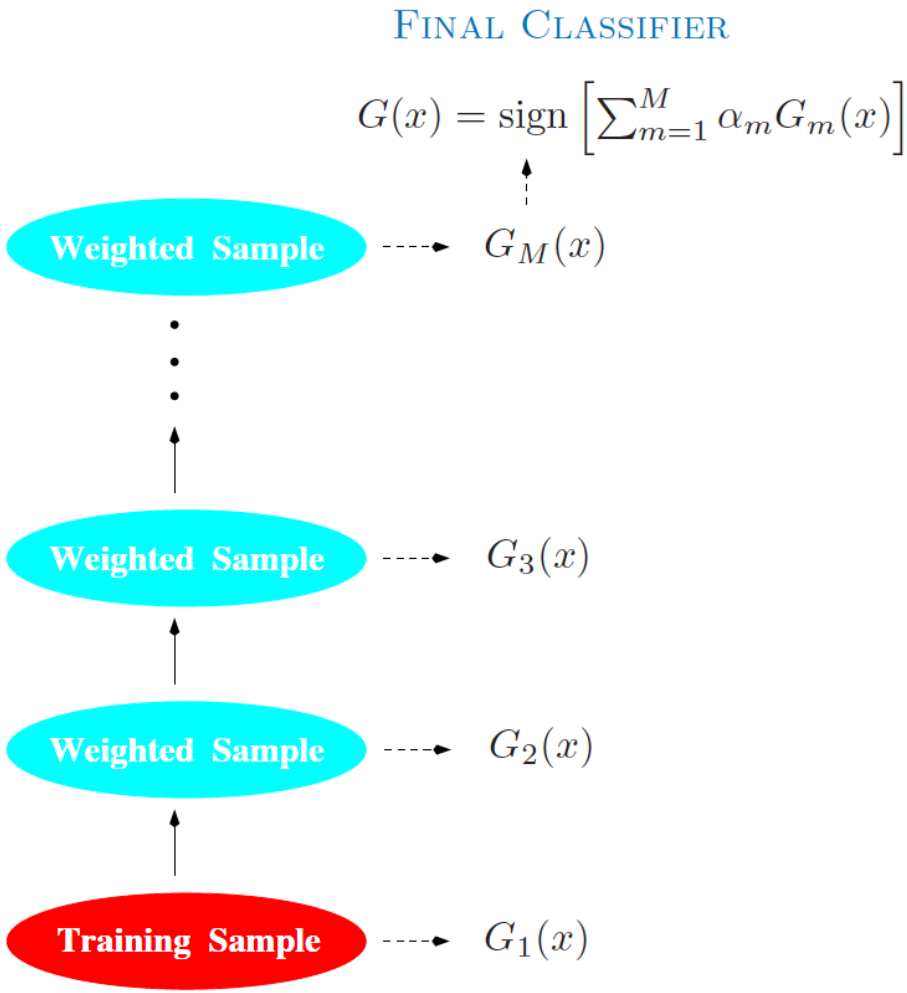
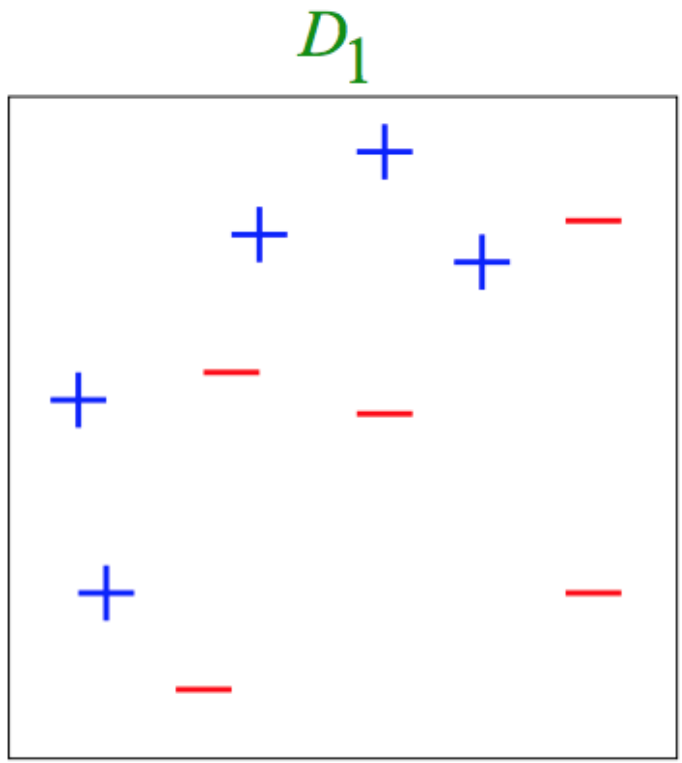


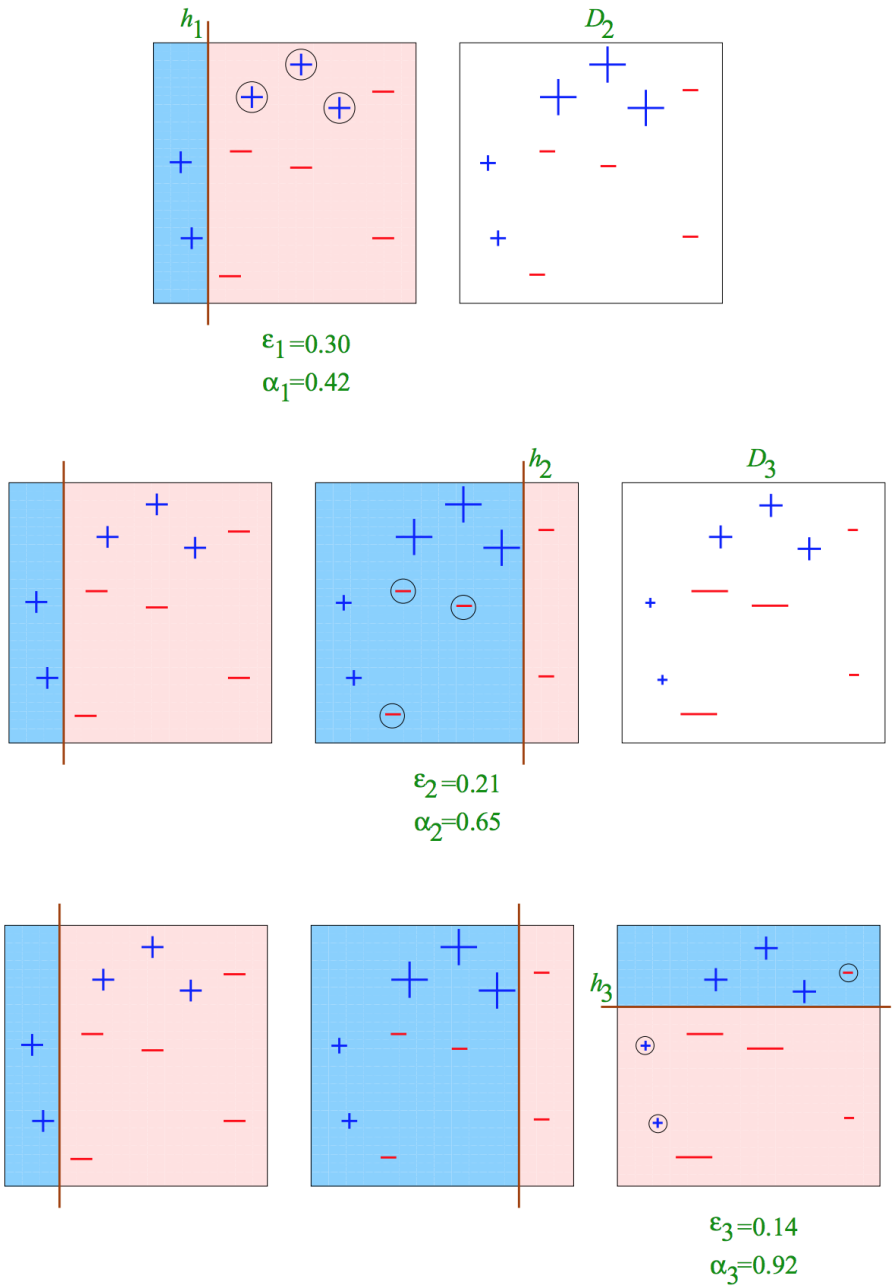
FIGURE 10.1. *Schematic of AdaBoost. Classifiers are trained on weighted versions of the dataset, and then combined to produce a final prediction.*

Toy example[1]

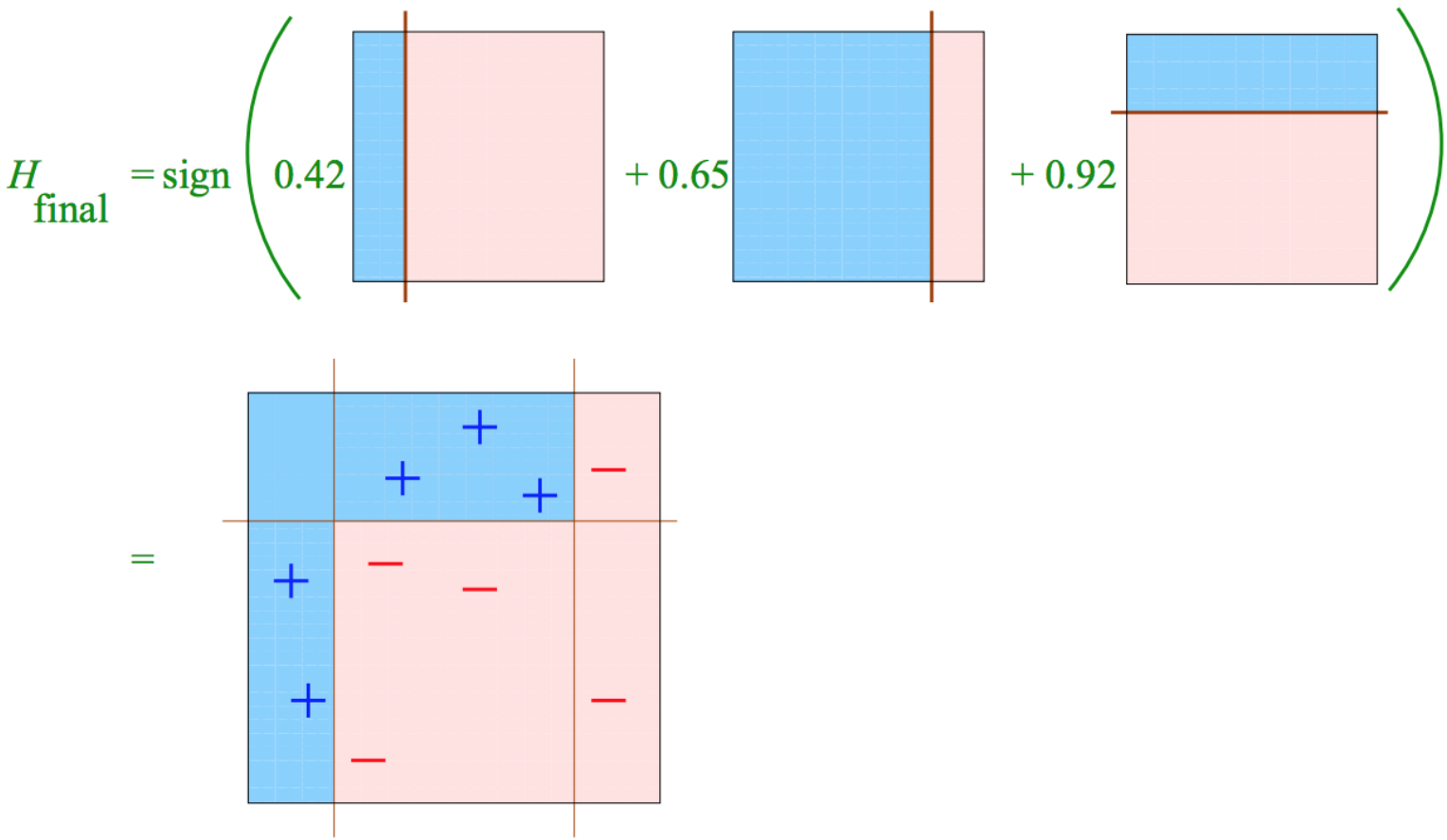


[1] https://www.cs.princeton.edu/courses/archive/spring12/cos598A/slides/intro.pdf

Rounds 1, 2 and 3



Final classifier



Simulation exercise

The features $x_1, \ldots, x_{10}$ are standard independent Gaussian, and the deterministic target y is defined

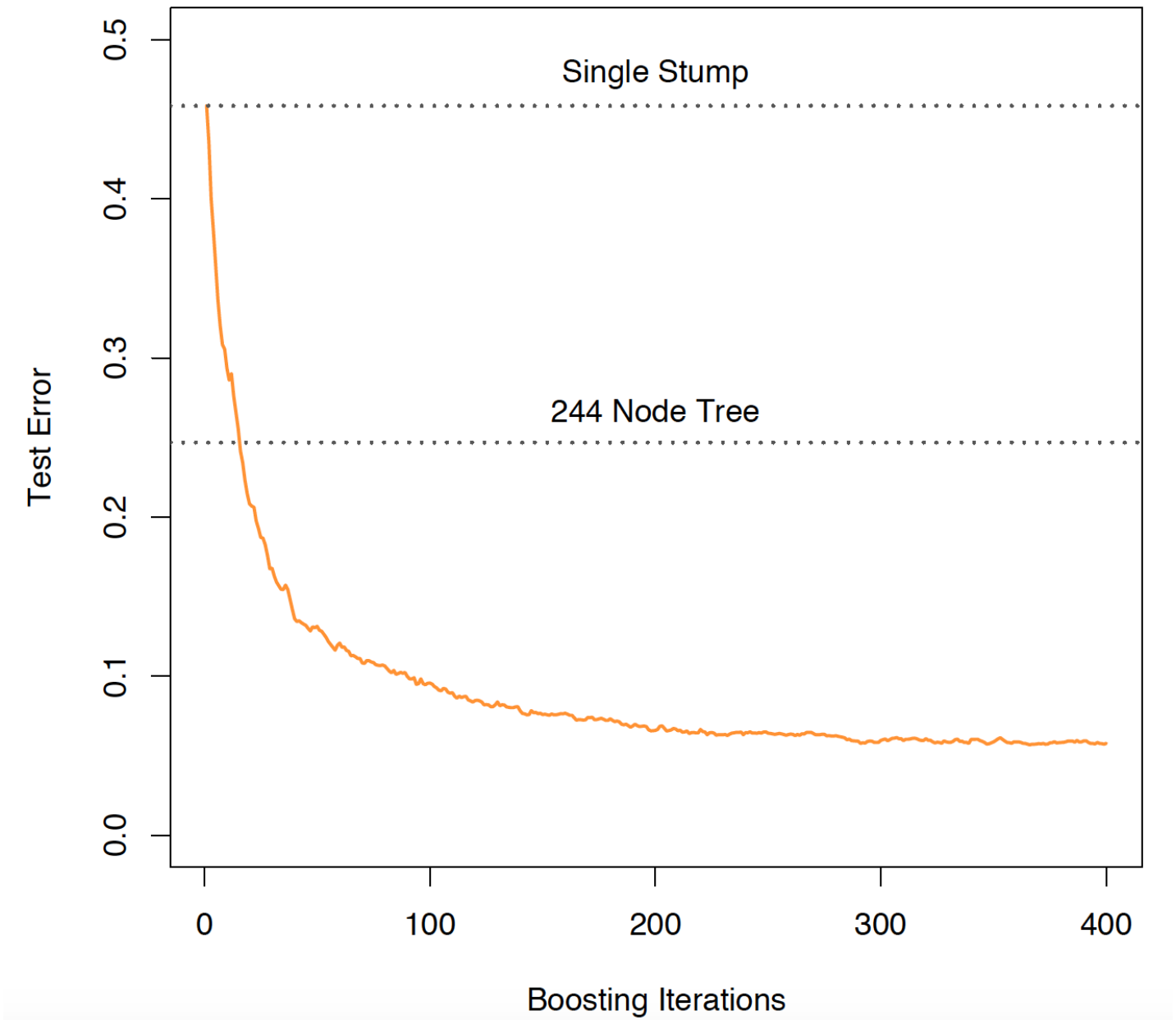
$$y = \begin{cases} 1 & \text{if } \sum_{j=1}^{10} x_j^2 > \chi^2_{10}(0.5), \\ -1 & \text{otherwise} \end{cases}$$

Here $\chi^2_{10}(0.5) = 9.34$ is the median of a chi-squared random variable with 10 degrees of freedom (sum of squares of 10 standard Gaussians). There are 2000 training cases, with approximately 1000 cases in each class, and 10,000 test observations.

Here the weak classifier is just a "stump": a two terminal-node classification tree.

Applying this classifier alone to the training data set yields a very poor test set error rate of 45.8%, compared to 50% for random guessing. However, as boosting iterations proceed the error rate steadily decreases, reaching 5.8% after 400 iterations. It also outperforms a single large classification tree (244-node & error rate 24.7%).

Figure 10.2


Single Stump
244 Node Tree
Test Error
0.0
0.1
0.2
0.3
0.4
0.5
0
100
200
300
400
Boosting Iterations

Boosting and additive models

The success of boosting is really not very mysterious.

Boosting is a way of fitting an additive expansion in a set of elementary "basis" functions.

Here the basis functions are the individual classifiers $G_m(x) \in \{-1, 1\}$.

More generally, basis function expansions take the form

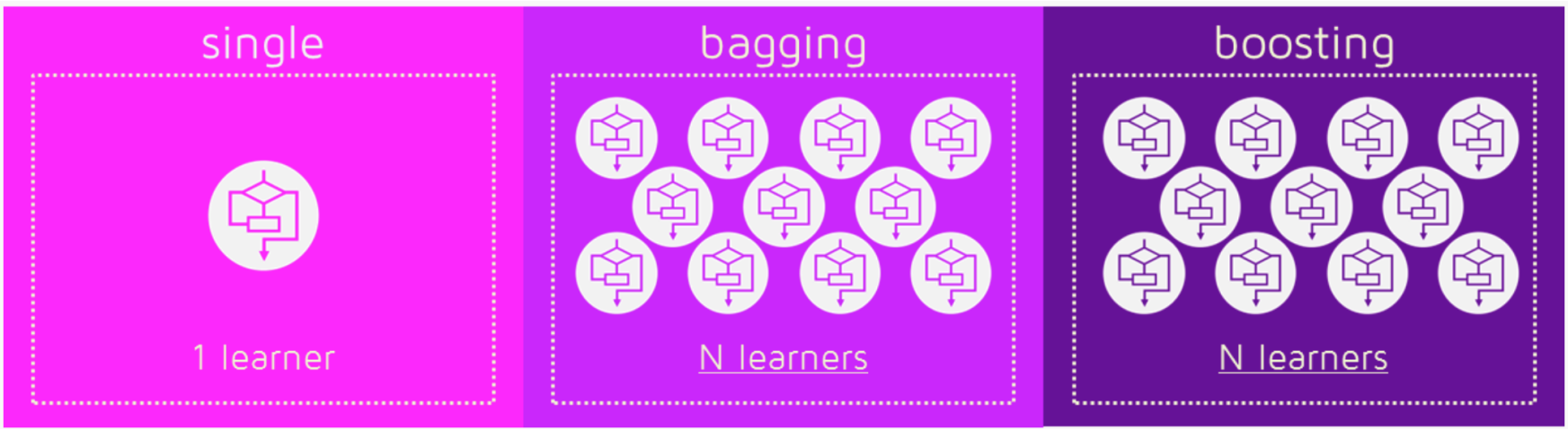
$$f(x) = \sum_{m=1}^{M} \beta_m b(x; \gamma_m),$$

where $\beta_m,\ m = 1, \ldots, M$ are the expansion coefficients, and $b(x; \gamma) \in \mathbb{R}$ are usually simple functions of the multivariate argument x, characterized by a set of parameters γ.

Additive expansions: single-hidden-layer neural networks, wavelets, multivariate adaptive regression splines (MARS) and trees.

Bagging or boosting?[2]

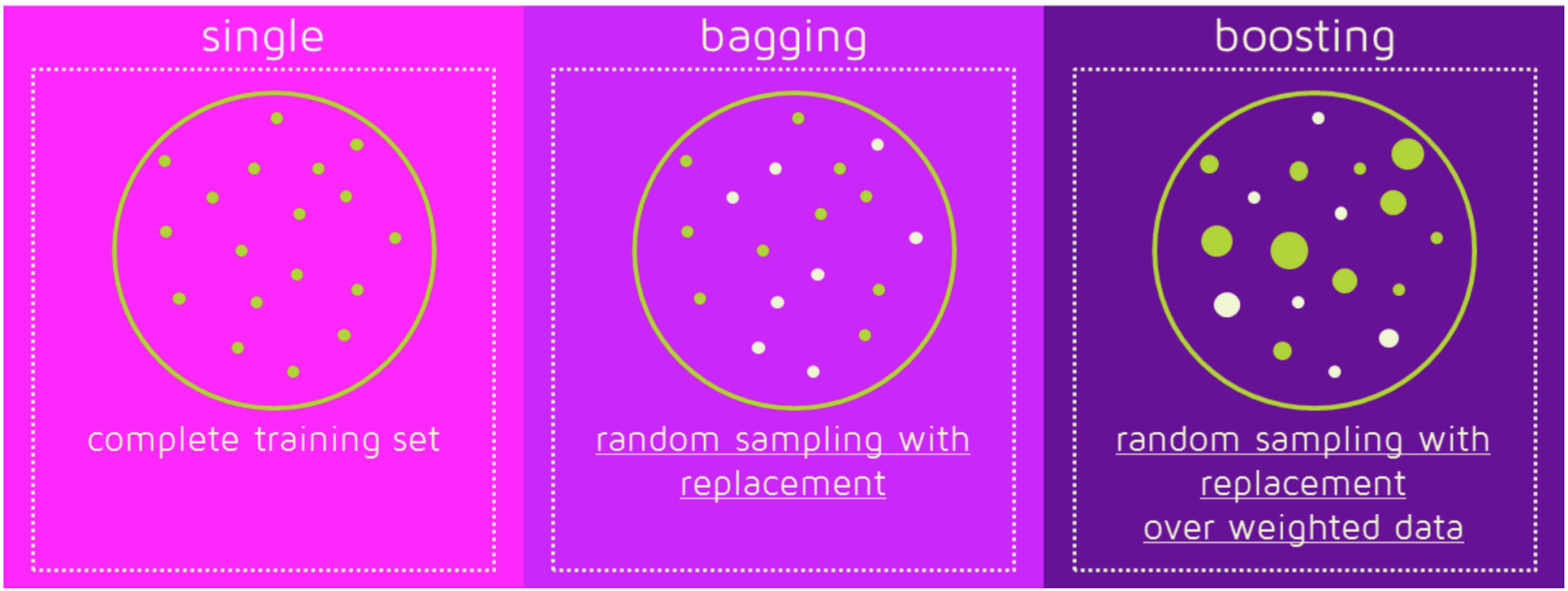
Bagging: N learners
Boosting: N learners



[2] https://quantdare.com/what-is-the-difference-between-bagging-and-boosting

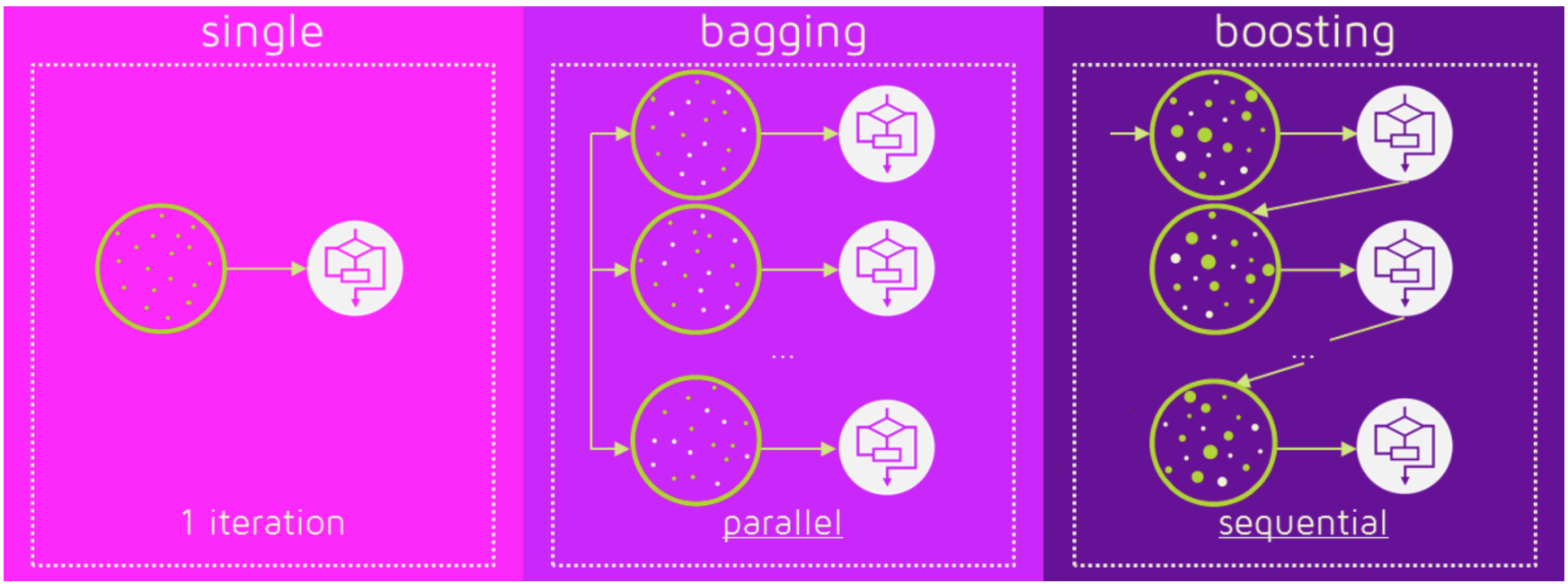
Bagging or boosting?

Bagging: simple random sampling with replacement
Boosting: weighted random sampling with replacement



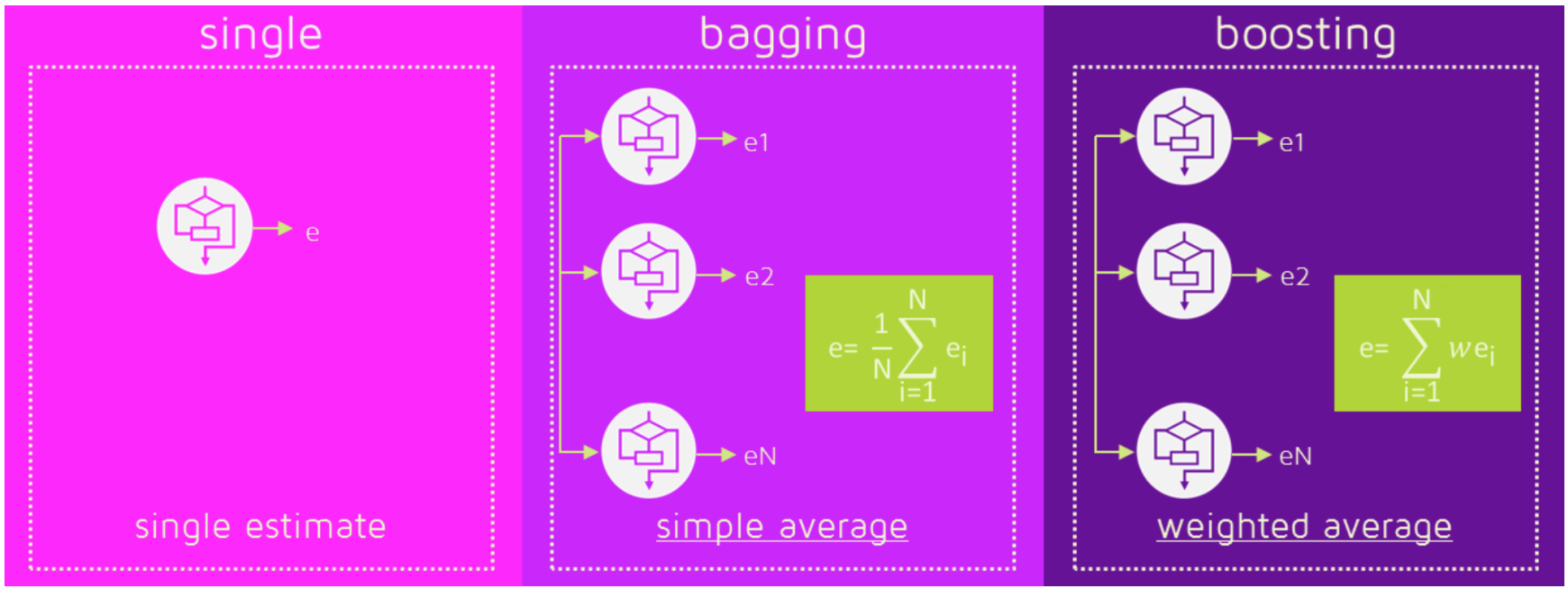
Bagging or boosting?

Bagging: parallel learners
Boosting: sequential learners



Bagging or boosting?

Bagging: simple average of errors
Boosting: weighted average of errors



Bagging or boosting?

Similarities	Differences
Both are ensemble methods to get N learners from 1 learner...	... but, while they are built independently for Bagging, Boosting tries to add new models that do well where previous models fail.
Both generate several training data sets by random sampling...	... but only Boosting determines weights for the data to tip the scales in favor of the most difficult cases.
Both make the final decision by averaging the N learners (or taking the majority of them)...	... but it is an equally weighted average for Bagging and a weighted average for Boosting, more weight to those with better performance on training data.
Both are good at reducing variance and provide higher stability...	... but only Boosting tries to reduce bias. On the other hand, Bagging may solve the over-fitting problem, while Boosting can increase it.

Comparison of learning methods

Some characteristics of different learning methods.
Key: 🟢= good, 🟡=fair, and 🔴=poor.

Characteristic	Neural Nets	SVM	CART	GAM	KNN, Kernel	Gradient Boost
Natural handling of data of "mixed" type	🔴	🔴	🟢	🟢	🔴	🟢
Handling of missing values	🔴	🔴	🟢	🔴	🟢	🟢
Robustness to outliers in input space	🔴	🔴	🟢	🟡	🟢	🟢
Insensitive to monotone transformations of inputs	🔴	🔴	🟢	🟢	🔴	🟢
Computational scalability (large N)	🔴	🔴	🟢	🟢	🔴	🟢
Ability to deal with irrelevant inputs	🔴	🔴	🟢	🟡	🔴	🟢
Ability to extract linear combinations of features	🟢	🟢	🔴	🔴	🟡	🟡
Interpretability	🔴	🔴	🟡	🟢	🔴	🟢
Predictive power	🟢	🟢	🔴	🔴	🟢	🟢

References

- Breiman (1996) Bagging predictors. *Machine Learning*, 24, 123-140.
- Breiman, L., Friedman, J., Olshen, R. and Stone, C. (1984). Classification and Regression Trees. Wadsworth, Belmont, CA.
- Breiman (2001) Random forests. *Machine Learning*, 45, 5-32.
- Buja, Hastie and Tibshirani (1989) Linear smoothers and additive models. *Annals of Statistics*, 17, 453-555.
- Freund (1995) Boosting a weak learning algorithm by majority. *Inform. and Comput.*, 121, 256-285.
- Freund and Schapire (1996a) Game theory, on-line prediction and boosting. In *Proceedings of the Ninth Annual Conference on Computational Learning Theory*, 325-332.
- Freund and Schapire (1996b) Experiments with a new boosting algorithm. In *Machine Learning: Proceedings of the Thirteenth International Conference*, 148- 156.
- Freund and Schapire (1997) A decision-theoretic generalization of online learning and an application to boosting. *J. Comput. System Sciences*, 55.
- Friedman (1991) Multivariate adaptive regression splines. *Annals of Statistics*, 19, 1-141.
- Schapire (1997) Using output codes to boost multiclass learning problems. In *Proceedings of the Fourteenth International Conference on Machine Learning*, 313-321.
- Schapire (1990) The strength of weak learnability. *Machine Learning*, 5, 197-227.
- Schapire and Freund (2012) *Boosting: Foundations and Algorithms*. MIT Press.
- Schapire, Freund, Bartlett and Lee (1998) Boosting the margin: a new explanation for the effectiveness of voting methods. *Annals of Statistics*, 26, 1651-1686.
- Schapire and Singer (1998) Improved boosting algorithms using confidence-rated predictions. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*.