Online GentleAdaBoost - Technical Report

A Preprint

Chapman Siu
Faculty of Engineering and Information Technology
University of Technology Sydney

chapman.siu@student.uts.edu.au

August 27, 2023

Abstract

We study the online variant of GentleAdaboost, where we combine a weak learner to a strong learner in an online fashion. We provide an approach to extend the batch approach to an online approach with theoretical justifications through application of line search. Finally we compare our online boosting approach with other online approaches across a variety of benchmark datasets.

***K*eywords** online boosting • adaboost • gentleadaboost • machine learning • classification

1 Introduction

Boosting algorithms belong to a class of ensemble classification approaches which use weak assumptions on the learner to efficient manner to improve performance. GentleBoost is an algorithm which was first introduced as an alternative Adaboost approach which uses Newton steps rather than exact optimization on each step (see Friedman, Hastie, and Tibshirani 2000, p353). Unlike other AdaBoost variants, GentleBoost has not received as much attention as it yields empirically inferior performance compared with other Adaboost algorithms when used on a wide range of benchmark datasets.

In machine learning, the ability to extend algorithms from a batch setting to an online setting is an important topic. Online approaches can operate on streams and use datasets which are too large to fit in memory. In this technical report we provide an approach to extend GentleBoost to the online setting through using line search. In addition we perform experiments to demonstrate that the algorithm is theoretically sound and has practical usecases.

2 Online Gentleboost

To describe the Online Gentleboost algorithm, we first describe the Gentleboost algorithm for the two-class classification scenario. The fitting procedure uses training data $(x_1, y_1), \dots, (x_n, y_n)$ where x_i is a training instance vector and $y_i \in \{-1, 1\}$. Then define $F(x) = \sum_1^M f_m(x)$ where every $f_m(x)$ is some weak classifier. Then the corresponding prediction is provided by $\text{sign}(F(x))$. For Gentleboost, it uses the *exponential criterion*, $J(F) = E(\exp^{-yF(x)})$ for estimation of $F(x)$.

Then if we use Newton steps for minimizing $J(F)$

$$\frac{\partial J(F(x) + f(x))}{\partial f(x)}|_{f(x)=0} = -E(\exp^{-yF(x)} y|x)$$

$$\frac{\partial^2 J(F(x) + f(x))}{\partial f(x)^2}|_{f(x)=0} = E(\exp^{-yF(x)}|x) \text{ , since } y^2 = 1$$

The corresponding Newton update is

$$F(x) \leftarrow F(x) + \frac{E(\exp^{-yF(x)} y|x)}{E(\exp^{-yF(x)} |x)}$$

The GentleBoost algorithm is then summarised in Table 1 shown below

Table 1: GentleBoost algorithm which is a modified version of AdaBoost that uses Newton stepping rather than exact optimization at each step

GentleBoost (see Friedman, Hastie, and Tibshirani 2000, p353) |

1. Start with weights $w_i = 1/N, i = 1,2, . . . , N, F(x) =0 $
2. Repeat for $m = 1, 2, \dots, M$:
   a. Fit the regression function $f_m(x)$ by weighted least-squares of y_i to x_i with weights w_i.
   b. Update $F(x) \leftarrow F(x) + f_m(x)$.
   c. Update $w_i \leftarrow w_i \exp(-y_i f_m(x_i))$ and renormalize
3. Output the classifier sign$(F(x)) =$ sign$(\sum_{m=1}^{M} f_m(x))$.

In our online boosting framework, the instances (x_i, y_i) only become available one at a time and the boosting algorithm must operate in an online fashion as well. As such it is not possible for the algorithm to determine the precise Newton Step at every instance. Instead, we perform line search over Newton steps, which is known to converge to the optimal Newton Step solution with sufficient small step. The choice of the step size becomes a hyperparameter related to the model, and removes the need to renormalize. The step size is chosen based on the observation it needs to be proportional to $\exp(-y_i f_m(x_i))$ and bounded by the range of $\exp(-y_i f_m(x_i))$ to meet the Lipschwitz condition (Armijo 1966). Since $-1 \leq -y_i f_m(x_i) \leq 1$ then with the choice of hyperparamter $\alpha \in (0, \exp(1) - 1)$ the step size $\hat{\alpha}$ is constructed as

$$\hat{\alpha} = \begin{cases} \frac{1}{1+\alpha}, & \text{if sign}(-y_i f_m(x_i)) > 0 \\ 1 + \alpha, & \text{otherwise} \end{cases}$$

As this approach uses a line search, any update function which directionally moves the weight in the correct direction will be suitable. The modified Online GentleBoost algorithm is summarised in Table 2 shown below

Table 2: Online GentleBoost algorithm which is a modified version of GentleBoost to allow for online learning

Online GentleBoost

1. Start $F(x) =0 $, with hyperparamter $\alpha \in (0, \exp(1))$
2. For incoming instance x_i, y_i, reset weight $w_i = 1$:
3. Repeat for $m = 1, 2, \dots, M$:
   a. Fit the regression function $f_m(x)$ by weighted least-squares of y_i to x_i with weights w_i.
   b. Update $F(x) \leftarrow F(x) + f_m(x)$.
   c. Update $w_i \leftarrow \hat{\alpha} w_i$ and renormalize |
4. Go back to 2. if there are additional instances
5. Finally output the classifier sign$(F(x)) =$ sign$(\sum_{m=1}^{M} f_m(x))$.

3 Results

We use the benchmark datasets and approaches in the River (Montiel et al. 2021) library to demonstrate the efficacy of our approach.

The model configuration uses the default settings and Hoeffding Trees Hulten, Spencer, and Domingos (2001) as the ensemble approach for AdaBoost (Oza and Russell 2001), Bagging (Oza and Russell 2001), GentleBoost algorithms. We also compare our approach with ADWIN Bagging Oza and Russell (2001), ALMA (Gentile 2000), Adaptive Random Forest (Gomes et al. 2017), Aggregated Mondrian Forest (Mourtada, Gaiffas, and Scornet 2019), Naive Bayes and Logistic Regression.

Table 3: Performance of Online GentleBoost compared with other algorithms in River

	Bananas	Elec2	Phishing	SMTP
ADWIN Bagging	0.625967	0.823773	0.893515	0.999685
ALMA	0.506415	0.906404	0.8264	0.764986
AdaBoost	0.677864	0.880581	0.878303	0.999443
Adaptive Random Forest	0.88696	0.876608	0.907926	0.999685
Aggregated Mondrian Forest	0.884318	0.854517	0.888711	0.999874
Bagging	0.634082	0.840436	0.893515	0.999685
GentleBoost	0.619362	0.804352	0.883106	0.999685
Hoeffding Tree	0.642197	0.795635	0.879904	0.999685
Logistic regression	0.543019	0.822166	0.888	0.999769
Naive Bayes	0.61521	0.728741	0.884708	0.993484

From the results above, we observe that GentleBoost generally performs worse across all datasets except for the Phishing dataset, however it demonstrates measureable uplift compared with the base weak learner (i.e. Hoeffding Tree).

More empirical evidence is required to verify this claim, though we note this inferior results is consistent with the batch GentleBoost empirical results which have been previously reported (see Friedman, Hastie, and Tibshirani 2000, p365).

4 Conclusion

We have introduced Online Gentleboost, an extension of the original batch Gentleboost approach via line search. We have justified our approach theoretically and demonstrated empirically that Gentleboost does indeed improve upon the weak learner.

References

Armijo, Larry. 1966. “Minimization of functions having Lipschitz continuous first partial derivatives.” *Pacific Journal of Mathematics* 16 (1): 1–3.

Bifet, Albert, Geoff Holmes, Richard Kirkby, and Bernhard Pfahringer. 2010. “MOA: Massive Online Analysis.” *J. Mach. Learn. Res.* 11 (August): 1601–4.

Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. 2000. “Special Invited Paper. Additive Logistic Regression: A Statistical View of Boosting.” *The Annals of Statistics* 28 (2): 337–74. http://www.jstor.org/stable/2674028.

Gentile, Claudio. 2000. “A New Approximate Maximal Margin Classification Algorithm.” In *Advances in Neural Information Processing Systems*, edited by T. Leen, T. Dietterich, and V. Tresp. Vol. 13. MIT Press. https://proceedings.neurips.cc/paper_files/paper/2000/file/d072677d210ac4c03ba046120f0802ec-Paper.pdf.

Gomes, Heitor M, Albert Bifet, Jesse Read, Jean Paul Barddal, Fabricio Enembreck, Bernhard Pfharinger, Geoff Holmes, and Talel Abdessalem. 2017. “Adaptive Random Forests for Evolving Data Stream Classification.” *Machine Learning* 106: 1469–95.

Hulten, Geoff, Laurie Spencer, and Pedro Domingos. 2001. “Mining Time-Changing Data Streams.” In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 97–106. KDD ’01. New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/502512.502529.

Montiel, Jacob, Max Halford, Saulo Martiello Mastelini, Geoffrey Bolmier, Raphael Sourty, Robin Vaysse, Adil Zouitine, et al. 2021. “River: Machine Learning for Streaming Data in Python.”

Mourtada, Jaouad, Stephane Gaiffas, and Erwan Scornet. 2019. “Amf: Aggregated Mondrian Forests for Online Learning.” *arXiv Preprint arXiv:1906.10529*.

Oza, Nikunj C., and Stuart J. Russell. 2001. “Online Bagging and Boosting.” In *Proceedings of the Eighth International Workshop on Artificial Intelligence and Statistics*, edited by Thomas S. Richardson and Tommi S. Jaakkola, R3:229–36. Proceedings of Machine Learning Research. PMLR. https://proceedings.mlr.press/r3/oza01a.html.