Article AMT-2020-78 – Response to Referee 2's comments

General (Major) Comments

1. When training the supervised ML algorithm, the estimation of the accuracy of ADABL on the validation ensemble (by the cross-validation technique) is presented in line 171: 99.5%. I think it is important to present also the accuracy on some testing ensemble, at least on the case study of April 19, 2017. It will justify the generalization ability of the applied AdaBoost algorithm showing the algorithm performance in an independent dataset. In my opinion, the training dataset could be insufficient.

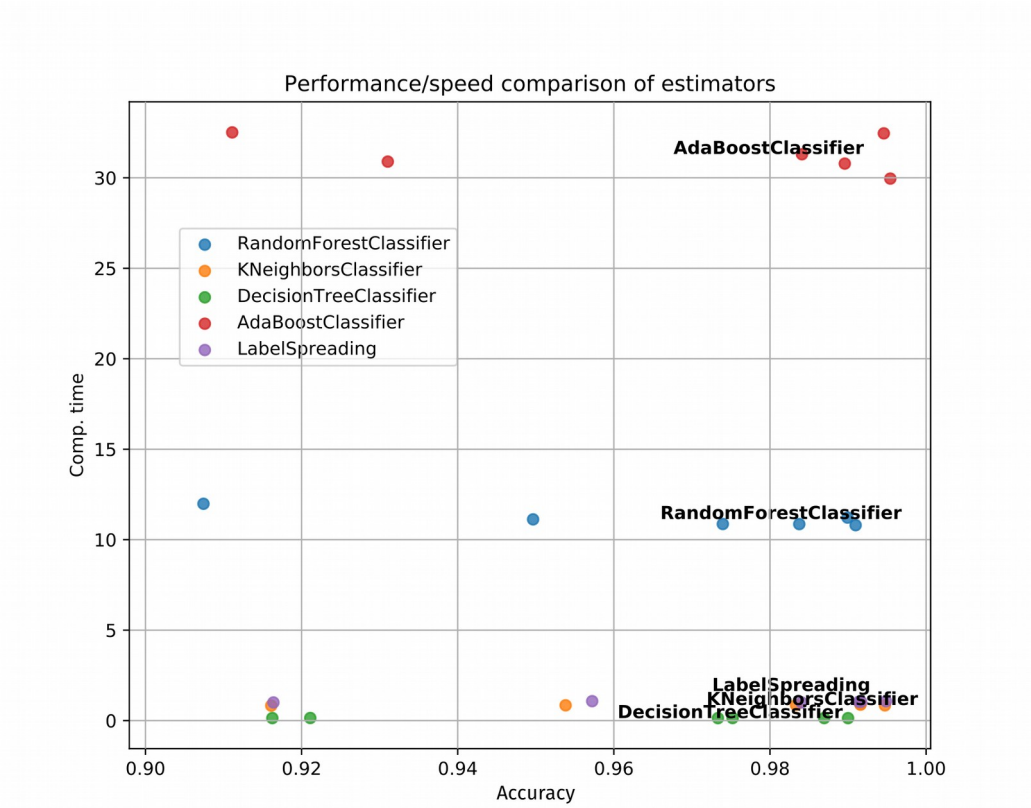
We acknowledge that the training dataset is insufficient, as it is commented in the results section. The point of this study is to show that, despite few training, ADABL has reasonably good results. As data labeling and algorithm training are time-consuming tasks, we would like to demonstrate first doing such tasks is of interest.
Time-consumption was also the reason why the case study of April 19 was not labeled, but the referee makes a very good point arguing that this would enable us to make a more accurate evaluation of ADABL (and KABL as well) on an brand new day. It will be added in the short term prospects.

In order to give a lower bound for the accuracy, we trained ADABL on one day and used the second day as validation set. The resulting accuracy is 87.6%. It is noticeably lower than the previous figure, but it suffers from a half-less rich training set.
In any case, the accuracy value is not meant to be a finding of this study. We extent this opinion in the next comment which is closely related.

2. Another reason why the accuracy of ADABL on validation ensemble is unrealistically high is the application of the random cross-validation split for time-correlated data (line 171). Using random selection from correlated datasets can lead to loss of generalization ability of the algorithm. The proposed ML method should be applied to new (independent) measurements. It means that AdaBoost should be trained-validated on uncorrelated parts of the dataset. If the data points were selected randomly from the whole dataset by the cross-validation procedure, it is highly probable that the similar neighbouring time points would be placed in both training and validation ensembles, which gives unrealistically good accuracy estimate on training-calibration datasets, but the worse result on another independent (test) dataset. I suggest the application of the block cross-validation.

We agree with the reviewer that the accuracy as it is estimated is too optimistic. Our effort were not directed towards having the best estimation of accuracy because it was only used to discriminate different supervised algorithms. The result of the comparison is not shown in the manuscript but can be found here: https://presentations.copernicus.org/EGU2020/EGU2020-19807_presentation.pdf
To meet with the referee's concern, the same study has been repeated with a block cross-validation. It was performed with group K-fold where groups were 4 hours chunks. The code to generate the figure is online on the Github repository (examples/perform_block_cv.py). The results are in the following figure. We can conclude that AdaBoost is still the most accurate among the one tested and its accuracy is rather 0.96 than

0.99.



To conclude, the accuracy value will be changed in the manuscript. The value of 0.96, obtained with block cross-validation is the most relevant in our opinion because it is more realistic than 0.99 and it was compared to the accuracy obtained by other algorithms with the same calculation. The lack of validation set will be highlighted as a limitation of the study.

3. If I understand right, the final configuration of the unsupervised ML algorithm (KABL) produces the classification using just one parameter - RCS0. In this case, the phrase in the conclusion (line 435) is misleading – "Both take the same input: one day of data generated by raw2l1 routine; . . ."

The final configuration of KABL does use only one parameter. But both algorithms still take the same input: a daily file generated by raw2l1 that contains all information needed for both algorithms, each algorithms extract only what it needs.

4. Line 272: "number of invalid values (NaN or Inf) are recorded." - Please explain why algorithms return these kinds of values. Another question is how algorithms deal with undefined values in Lidar measurements.

Algorithms return NaNs when all the points of the profile are assigned to the same cluster. For ADABL, it happens when the profile is very different than the ones in the training set (not that rare). For KABL, it happens when we specified the initial centroids (it the case in the retained configuration) and only one of these points gather a cluster around it (very rare). When lidar has few undefined values in the profile, they are

just ignored and the estimation is made on the available points.

Specific (Minor) comments

5. Line 15: ". . . boundary layer height (BLH). . ." – please give somewhere a definition of the BHL.

BHL is probably a typo error for BLH, as a research for "BHL" inside the manuscript returned no results. The BLH definition we used here is in the next sentence (lines 15-16) as "the depth of atmosphere where all pollutants emitted from the ground will remain".

6. Line 105: ". . . we chose to derive boundary layer height with parcel method for the 11:15 sounding and bulk Richardson number for the 23:15 one." Please justify why two different methods were used for morning and evening radiosounding.

We followed the recommendations in Seibert et al. (2000), figure 10, assuming that morning launch is in unstable atmosphere and evening launch is in stable atmosphere.

As the other referee had a major comment about the method used for RS, a more complete answer was given. It might be of interest here if any additional question arise.

7. Line 113: Does false positives on cloud detection perturb a BLH detection? Please explain.

Cloud screening with the collocated CL31 was only used to exclude cases for the comparison with the radiosoundings. Therefore, false positives in cloud detection would have for only effect to improperly reduce the comparison sample. The MiniMPL detection of clouds was found to reduce too much the comparison sample, while CL31 detection of clouds looked more reliable.

8. Line 113: In the following text, some basic ML concepts are introduced for readers, who are not familiar with the scope of ML. In this case, the "false positives" should also be explained or referenced.

The false positives refer here to the detection of clouds, not to the BLH detection.

To avoid confusion, the following sentences "Although the MiniMPL (...) to make some false positives." will be replaced by "Although the MiniMPL is perfectly capable of detecting clouds, we relied more on the cloud detection with the CL31, because MiniMPL's cloud detection was detecting cloud where there was not."

After the correction of the misleading use of words "false positives" and "algorithms" line 113, we do not think it is necessary to introduce false positive in section 3.

9. Line 127: As the number of seconds, since midnight is a periodical function, the 'classical' distance could not take it in consideration correctly this variable. It means

that the classical distance between one 00:01 and 23:59 will be nearly 24 hours. Please make sure, that ADABL algorithm works as expected in this case.

This remark is very important for the next stages of the algorithms development. However, at the moment, both algorithms have been used only on 24 hours chunks, so that the periodicity was not an issue.

10. Line 142: I do not see any subsampling in figure 3. Is it a five-forks weak learner creation part? Please specify.

It is correct to see no subsampling in the figure 3: AdaBoost do not perform subsampling, as the referee points out in the next comment.

Figure 3 is only meant to illustrate the boosting algorithm (as figure 2 is only meant to illustrate decision tree). They do not reflect the actual settings of the algorithms, which would needlessly blur the pedagogical effort. The decision trees used as weak learner are five-forks trees or less.

11. Line 142: How these shallow decision trees are fitted? I have never heard about resampling in the classical AdaBoost. Is it Bagging? Please give a reference or explain the algorithm in detail.

The referee raises an error here: indeed the weak learners are not trained on a subsample but on the whole dataset. Although, the weight put on each sample changes from one weak learner to another. Weak learners are trained with CART algorithm. A very good description of the algorithm can be found in Hastie et al. (2009) page 307.

12. Line 143: ". . .the error of the classifier is the number of misclassified points." - I am not sure that error is defined like that. Please explain or give a reference.

We acknowledge this sentence is incorrect: the error is the weighted average of misclassified points. For more details, the formula is given in Hastie et al. (2009) page 339, algorithm 10.1. The actual implementation can be checked in the Scikit-learn source code: sklearn.ensemble._weight_boosting.py: line 528.

13. Line 146: The explanation is not sufficient. I propose to present here a reference to any popular textbook on AdaBoost or carefully introduce the algorithm. For example, in the expression, the performance was not introduced, the upper limit in the sum should be capitalized, etc.

The reference to Hastie et al. (2009), which is a popular textbook explaining the algorithm with many details and well written, was made at the section 3.1. It is completed by the reference to Freund & Schapire (1997), which is the original publication of AdaBoost and contains many theoretical results. The authors will complete theses references with Schapire, R. E. (2013). Explaining adaboost. In *Empirical inference* (pp. 37-52). Springer, Berlin, Heidelberg.

This paragraph was kept short to avoid overwhelming the reader with technical details. We will reformulate

in order to keep the main idea and refer to the literature for the details.

14. Line 170: “trade-off between accuracy and computing time” - I do not think that the limiting factor for this problem is the computing time. Normally this kind of problem could be sufficiently well resolved by parallel computing.

It is true that computing time is not critical here because it is always low, even without parallel computing. However, we do not want a classifier needlessly complex. For example, fully-grow trees would be long to train and test for very little extra-performance.

As computing time is not a major factor, we will change the sentence line 170 by “It was chosen because more complex classifier do not show greater performance.”

15. Line 169: “RCSco, RCScr” – please make sure that these names for copolarized and crosspolarized range-corrected backscatter signals persist in the following text (notably in tables 2 and 3).

RCSco and RCScr are named respectively RCS1 and RCS2 in the source code. We chose to use RCSco and RCScr in the text because it is less ambiguous and to use RCS1 and RCS2 in the table so that readers going through the code would not be confused.
As this choice seems to be confusing for reader, we will keep only RCSco/cr in the paper and make the correspondence within the code

16. Line 175: “It is possible to quantify the relative importance of the predictors (Breiman et al., 1984; Hastie et al., 2009). After the training, the time accounts for 30.3%, RCSco for 28.4%, RCScr for 26.5% and the altitude for 14.8%.” – I have not found this information; could you please specify the corresponding page numbers?

In Hastie et al. (2009), it can be found page 368, equation 10.43.
In Breiman et al. (1984), it can be found page 147, definition 5.9.

17. Line 184: “distances from all points to all centroids” – Are these the Euclidian distances?

Yes, K-means usually implies Euclidean distance. Although it is technically possible to use any distance, most of implementations do not provide this option.

18. Line 196: “If we assume all Gaussian have the same fixed variance and that this variance tends to zero, EM and K-means algorithms are the same.” – Could you provide a reference or explain the statement?

In Hastie et al. (2009), exercice 14.2, page 580.
Let's consider a sample generated by a mixture of two Gaussian with the same fixed variance. When the variance tends to zero, the "expectation" step is the same as the attribution to the closest centroid, and the "maximisation" step is the same as updating the centroids.

19. Line 203: "Then the data they contain are normalized. . ." – if time and height are used in the KABL algorithm, are these variables also normalized?

Time and height are not used in KABL algorithm. However, all the data are normalized in order to avoid any unit comparison problem. The normalization consists in removing the mean and divide by the standard deviation. It is done in kabl.core.py: line 152 (release 1.0.0. Current commit: line 167).

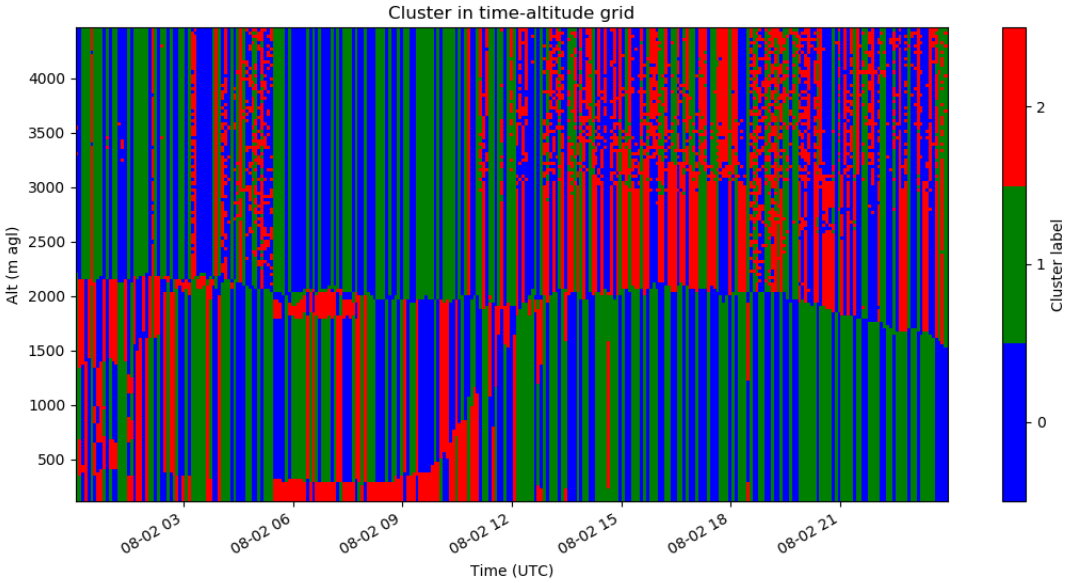
20. Line 205: What values are included in predictors? If X matrix contains only signals RCSco and RCScr, it should be stated somewhere.

Predictors can be either RCSco all day, either RCSco and RCScr all day or RCSco for daytime and both RCSco and RCScr for nighttime, as precised in Table 2.
This question arises because of the ill-suited organisation of the text, as pointed out by comment 22, but this will be corrected.

21. Line 209: "Finally, we look for the first change in clusters attribution, starting from the ground. This gives us the BLH for this profile." I am not sure that this algorithm is optimal, as it could lead to oscillations of BLH. To understand how it could be improved I suggest presenting and analyzing the altitude-time plot with pixels representing the results of the classification (like Fig 11 but with classes). Probably it is better not to take the first change, but a height above which the class is not changing, e.g. for three levels. Alternatively, a value of height could be selected that persist in time. These kinds of parameters could be optimally selected by the scores optimization. Another option is to modify the 'distance' definition.

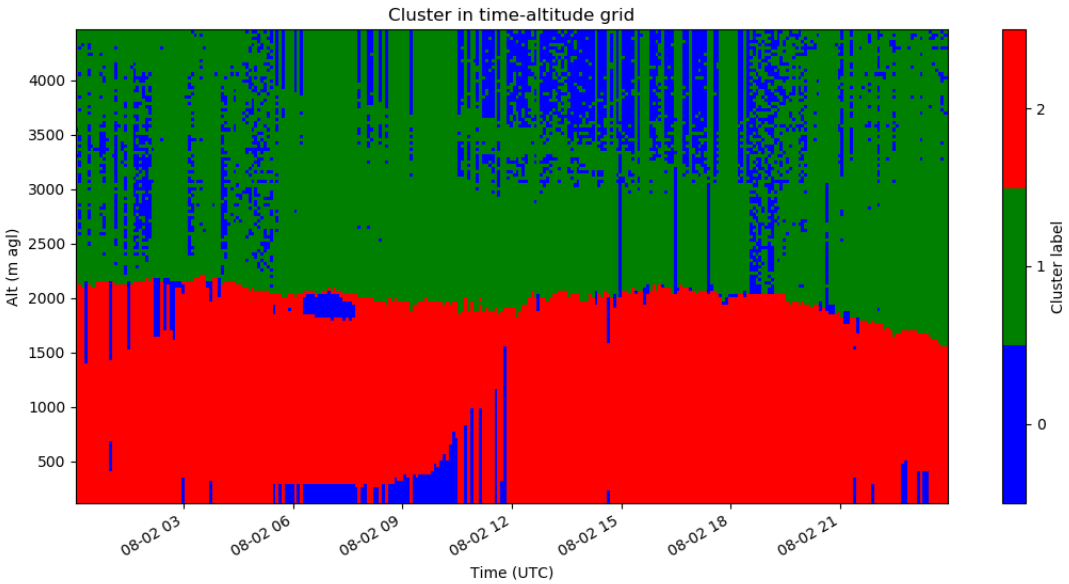
We do observe some oscillations as the referee describes (e.g. see figure 11). The proposition made by the referee, to enforce vertical persistence of the clusters, is very relevant and will be added as a prospect. The time continuity is also very relevant, and existing methods already use such criteria, thus are a source of inspiration to implement it here. The distance definition is probably the "smartest" way, as it can help us learn about what really matters to distinguish boundary layer from the rest, but it is also the less intuitive. The optimization of parameters was done here thanks to global sensitivity analysis, but it would greatly benefit from being repeated after such new features are added.

The following figure shows the altitude-time plot of the classes attributed by KABL, with random initialization (K-means "vanilla"). It makes very visible the random attribution of the classes numbers in unsupervised classification: only borders matter.



A way to avoid such random attribution is to specify the initial centroids: the resulting plot is next. Initial centroids were put at typical backscatter values (modes of the histogram). The blue cluster has a very high backscatter (it detects cloud and shallow morning BL), the red cluster has high backscatter (it detects mixed layer or residual layer), the green cluster has low backscatter (it detects free atmosphere). We can see patches of blue in the free atmosphere: they are not realistic. They occur in profiles were there is no strong backscatter corresponding to this cluster. As we still ask for 3 clusters, the blue cluster sticks to noisy points.

The BL top defined as the height where cluster stop changing, as suggested by the referee, would be affected by such noisy points.



22. Line 213: “The parameters of this computer code. . .” these parameters should be introduced at the beginning of the section 3.3, before they are referenced.

The organisation of this paragraph will be changed accordingly.

23. Line 298: “. . . figure 8 the distribution (violin plots) of the relevant output conditionally to the parameter value.” – I suggest adding here a reference on the construction of this kind of plot.

Here is a reference describing the plot and its use:
Hintze, J. L., & Nelson, R. D. (1998). *Violin Plots: A Box Plot-Density Trace Synergism. The American Statistician, 52(2), 181–184.*

24. Line 304: “Parameters values are chosen to give the most optimal value for the metrics they have influence on.” - The selection of locally optimal combination of parameters does not provide the globally optimal solution. How can you be sure that this combination gives the best precision?

This is a good remark: we cannot be sure. However, a sensitivity analysis on the 2-year long dataset would be computationally too expensive. The sensitivity analysis on a single day has the advantage to be more thorough than anything we could have done on the whole dataset. It helped us to selected few configurations to test on the whole dataset, including the configuration described in Table 3, which was elected because it had the best results. A sentence will be added to make clear that the sensitivity analysis on one day was used to select few configurations to be tried on two years.

25. Line 316: “As the average gap E1 and the RMSE E2 are very similar. . .” – I suggest excluding the average gap E1 from the article for the sake of simplification.

Yes, this is something we will do in the next version of the manuscript.

26. Line 326: “Nighttime (launch of 23:15 UTC)” - If nighttime radiosounding was not used, why to present this dataset in “2.2 Radiosonde data”? Probably it was used in supervised ML? Please specify.

Nighttime launches were not used in the supervised algorithm.
They were only used for the comparison to the diurnal BL cycles presented in Fig. 10.

27. Figure 9: Adding the confidence intervals for RMSE and Correlation in Fig. 9 could be quite useful.

Yes it would. We will add bootstrap estimations of confidence intervals in the next version of the manuscript.

28. Line 335 and Line 438: I think it would be advantages to understand how works the lidar manufacturer's software and to give some interpretations.

We asked the manufacturer for more details about their algorithm: a modified wavelet transform method described in Brooks, 2003 is used. In that regard, we will expand on the interpretations of the results for the revised manuscript.

29. Line 399: "A method to filter these oscillations will be needed, but it can also divest the "real-time" property." – Instead of filtering, the criteria of the lowest transition of the class for KABL could be somehow modified, as I proposed in my comment for line 209. The filtering could be of the "real-time" if it is done relatively the past classifications.

Yes, the comment 21 was full of good ideas to be put in the prospects. We will add the use of past classification to our answer to this comment.
As a matter of fact, the sensitivity analysis revealed that concatenating previous profiles do not solve these oscillations. Therefore, it is really the previous output of the classification that should be used in future filtering.

30. Line 422: "5.6 KABL is "trainingless"" – I suggest that KABL could be used also by an expert to simplify the learning stage of supervised ML.

As we understand this comment, the referee suggests to make first an unsupervised classification (with KABL), correct it manually and then use it as a reference to train a supervised classifier (as ADABL). This a very interesting strategy to reduce the burden of supervision in ML methods, even beyond the only question of boundary layer height estimation. For example (still close to the topic), this method is currently under experimentation to make boundary layer classification: https://github.com/ThomasRieutord/bl-classification
However we would like to emphasis that the manual correction between unsupervised and supervised classification can hardly be by-passed. First, if the result of unsupervised learning can be used as a reference, why use a supervised model after? Second, unsupervised learning tells which classes are different but not which class is what. The identification of the classes must be done by a human expert or a reliable (physically-based) strategy.

31. Line 417: ". . .strategies . . . for the training of ADABL. . ." – To decrease the sensitivity to "idealized" diurnal cycle of the BLH, I suggest trying to exclude the time predictor in ADABL.

Yes, this is an idea. Furthermore, a sensitivity analysis, as was done for KABL, would be very helpful to

know better how to correctly set ADABL.

Technical corrections

Thanks, will be corrected in the submitted version

32. Line 13: “Atmospheric boundary layer concentrates many scientific challenges (small scale flows, turbulence...) and with high impacts due to its position of the interface between ground and atmosphere.” - awkward English, please correct.

33. Line 27: “(clouds, residual layers..)” -> “(clouds, residual layers. . .)”.

34. Line 88: “SIRTA” - Please decrypt the abbreviation.

35. Line 92: “at 11:15 AM and PM” – Please utilize the same notation for the time here and further. I suggest the UTC format.

36. Line 94: Please explain what the theta is. Is it the potential temperature?

37. Line 127: I suggest inserting a comma after “height above ground”.

38. Line 129: “[[1,N]]”- What does double brackets means? Please explain.

39. Line 145: “tree” -> “trees”.

40. Line 146: “m=200” -> “M=200” + upper case in sum limit

41. Line 154, 157, fig. 4: “top”->”left”, “bottom”->”right”.

42. Line 208: Init parameter is not defined.

43. Line 208: “specified in algo” -> “specified by algo parameter”

44. Line 270: “(0.20)” Is it the software version? Please specify.

45. Figure 10: Please introduce the INDUS abbreviation.