Atmos. Chem. Phys. Discuss.,
https://doi.org/10.5194/acp-2020-509-AC2, 2020

Atmospheric Chemistry and Physics Discussions

Interactive comment on "Using machine learning to derive cloud condensation nuclei number concentrations from commonly available measurements" *by* Arshad Arjunan Nair and Fangqun Yu

Arshad Arjunan Nair and Fangqun Yu

aanair@albany.edu

Received and published: 23 August 2020

Printer-friendly version

Discussion paper

Response to Anonymous Referee #2

Referee Point P 2.1 — This study demonstrates the use of a random forest regressor (RFR) to predict cloud condensation nuclei (CCN) concentrations at 0.4% supersaturation from features derived from chemical-transport model output (GEOS-Chem-APM) and measurements from a surface monitor. The authors train a RFR using GEOS-Chem-APM output and compare these results to observations at the SGP monitoring site. They then re-train models based on measurements available at this site. While I found the approach promising and the subject matter (using a combination of modeling techniques to predict CCN) fits in well with the subject matter at ACP, I feel that the scientific goal of the manuscript in its current form is not sufficient to recommend publication.

Reply: We are grateful to Anonymous Referee #2 for reviewing this manuscript and providing constructive comments and suggestions, which have helped us revise and improve the clarity of this manuscript. We have addressed the referee's concerns about the scientific goal of this manuscript, as detailed below. Our replies addressing the referee comments follow; corresponding revisions (boxed) are in the revised manuscript and marked in a separate tracked-changes version.

General Comments

Referee Point P 2.2 — My main concern with this manuscript is in framing the results as a "proof of concept". Given the current state of literature for CCN and machine learning techniques, I do not think a proof of concept is sufficient for publication (for instance, there are a number of previous studies cited here that make use of machine learning techniques for atmospheric science). Further, as a proof of concept, the comparison to observations is limited to one surface site. Could the comparison to observations

Printer-friendly version

Discussion paper

be expanded to include more sites? This paper could be strengthened by applying this technique further than what the authors did in the present study. The suggestions the authors made in Section 4.2 sounded promising. I especially thought the suggestion at Line 396 (using this approach in models with only bulk aerosol schemes) was a good idea. For instance this paper could be strengthened by: training the model with GEOS-ChemAPM, predicting the CCN based on the standard (bulk microphysics) GEOS-Chem, and then comparing to a larger suite of measurements (perhaps including ATom or sites with an SMPS in Europe). This way the authors could demonstrate if the random forest model provided an improved estimate in CCN over the approach used with the standard GEOS-Chem model. (Note, I am not suggesting the authors follow this suggestion exactly but I am trying to demonstrate how I feel the argument of this paper could be strengthened). Overall, the authors did a detailed job in training the RFR but the use of it is limited to comparing to only one surface site. How robust is the model if we only see it compared to one site?

Reply: We agree. A novelty of this work compared to the literature we cited in the introductory section is that we leverage a multi-modeling approach, i.e. a machine learning model that learns from a global size-resolved particle microphysics model for effective application to resolve the absence of measurements. Additionally, no previous study has examined this approach for CCN, the quantification of which is significant for estimating the effective radiative forcing through aerosol–cloud interactions. Also, Anonymous Referee #1 notes (doi:10.5194/acp-2020-509-RC1): “CCN number concentrations is an essential task for environment evaluation and can be used for many different applications” and that “experimental results demonstrate robustness of the proposed method.” The points and suggestions made in P 2.2 are good; we have been working on these concurrently. However, we feel that inclusion of these results are beyond the scope of the present work. The present manuscript is a demonstration of random forest regression modeling (RFRM) for resolving the absence of [CCN] measurements with a note in the concluding section about the potential application

of this work to improve their simulated values using bulk microphysical models. With regards to the former, it is indeed possible to extend the approach to other sites; however, measurements are highly varied in the parameters quantified (which are used as 'predictors' in the RFRM) and we feel inclusion of results for other sites and corresponding discussion of site-specific machine learning model development will only be repetitive and take away attention from the crux of this paper. Regarding the latter, we believe that the necessary optimizations to be made of the machine learning model for balance between speed and accuracy in modified chemical transport/earth system models and subsequent analysis of the modified model performance are a topic that deems a separate discussion in itself.

Referee Point P 2.3 — Similarly, the discussion in Section 3.2.2 (Lines 316 and onwards) about training new models with different features feels somewhat unclear as to what the goal is. This is related to my comment above about the framing of the study. Is the goal to use actual observed pollutant and meteorological measurements to predict CCN or to use the same values used in GEOS-Chem and provide an alternate GEOS-Chem estimate? It seems problematic to have one model trained on global GEOS-Chem-APM output and a second based on observations at one site (unless the goal here is to compare why the two predictions are different?).

Reply: In Section 3.2.2, we examine the ARM SGP site in detail with model–observation comparison. Lines 316–322 describe the re-training of the RFRM with fewer variables (due to absence/paucity of observations; discussed in preceding Lines 310–315, Table 4) to create RF-ShortVars. Lines 323–332 discuss how RF-ShortVars performance changes compared to the better-optimized RFRM and how this truncated model compares to observations. Lines 333–346 discuss RFRM training using measurement data (rather than the typically large amount of training data that models can provide; here measurement data size is $\approx 0.2\%$ that of modeled) to create an observation-based RFRM (ORF) and corresponding evaluation of the ORF. This ex-

ercise: (a) demonstrates that it is unlikely that we have neglected to consider some important 'predictor', (b) provides an additional check on the importance of the 'predictors' learned in the GEOS-Chem-APM-based RFRM, (c) validates the consideration of fewer (based on availability of measurements from $19 \rightarrow 9$) variables in RF-ShortVars, and (d) shows that if large observational data sets of 'predictors' are available, the RFRM for [CCN] derivation can be directly developed on measurements rather than having to learn from the model. So, yes, one of the goals here is to compare why the two predictions are different. However, we note (also in Lines 345–347) that presently we were unable to find 'large-enough' (somewhat vague term: dependent basically on how good the RFRM performs, which is reduction of incorrect generation of feature space due to a small bootstrapped sample showing spurious association to the randomly sampled subset of predictors during node-splitting) measurement data for this; and that the exercise should only be considered academic. We have added the following lines to reflect the present discussion:

Lines 384–387

Further, this exercise is insightful in demonstrating the unlikelihood of missing any important predictor in the RFRM, providing an additional check on the importance of the RFRM predictors, and the potential utility of this machine learning approach being trained directly, without a physicochemically informed model, on atmospheric state and composition measurements to derive [CCN0.4].

Lines 62–63: "The goal of the present study is to explore the possibility of deriving the number concentrations of CCN at 0.4% supersaturation ([CCN0.4]) through more ubiquitous measurements of atmospheric state and composition."

Specific Comments

Referee Point P 2.4 — Section 3.2.2: I think it would be useful to compare the performance of the RFR model relative to the observations at SGP and GEOS-Chem-APM at SGP. Does the RFR perform as well at predicting CCN as GEOS-Chem-APM? Or is the decrease in accuracy reasonable given the improvement in computation time of RFR over a microphysical model?

Reply: We agree. While these have already been discussed in some detail in Section 3.2.2, you make a great point of warranted further discussion focusing on the relative performance of RFRM and GCAPM. We have added Fig. 1 (on Page C14) as Figure 12 (b) with associated discussion in the revised manuscript as follows:

> **Lines 344–349**
>
> Fig. 12 (b) compares GCAPM and RFRM performance in quantifying [CCN0.4] for SGP with respect to its measurements corresponding to Fig. 12 (a). In the left-hand panel, GCAPM simulated [CCN0.4] shows fair correlation ($\tau \approx 0.27$) with SGP measurements and 65% within the good-agreement range. The general tendency is overestimation ($\text{median MFB} \approx 0.25$), seen as higher density above the perfect agreement line (dashed black). RFRM-ShortVars derives [CCN0.4] (Fig. 12 (b): center) to a greater degree of agreement than GCAPM does, with $\tau \approx 0.37$ and $\approx 75\%$ with $|\text{MFB}| < 0.6$ and $\text{median MFB} \approx -0.04$ indicating a slight tendency to underestimate.

and continued:

Lines 350–367

It is to be recalled that the GCAPM [CCN0.4] is more reflective of regional tendencies, simulating [CCN0.4] for a 2×2.5 gridbox around the SGP site. The RFRM is trained on GCAPM, from where the associations were learned between atmospheric state and composition variables and [CCN0.4], thus implicitly imbibing the effect of physical and chemical processes that control particle number concentrations. That RFRM-ShortVars-derived [CCN0.4] is better-representative is a demonstration that these processes are well-represented within GCAPM. Leveraging this aspect as well as utilising localised conditions (actual measurements) of atmospheric state and composition, the RFRM performs significantly better than GCAPM. These results and the ability to capture the variability of [CCN0.4] across temporal scales demonstrates the derivation of [CCN0.4] through the more commonly available measurements of meteorology, atmospheric chemical species including speciation of particulate matter.

The right-hand panel of Fig. 12 (b) shows how RFRM-ShortVars performs when using GCAPM simulated values of the input predictor variables. $\tau \approx 0.24$ and 71% with $|\mathrm{MFB}| < 0.6$ indicates that the RFRM model performance is comparable to GCAPM for the SGP site in alignment with our observations in Section 3.2.1. This is encouraging towards further development of this machine learning approach for potential application in Earth system models (ESMs). The Random Forest technique discussed here has two key virtues: (1) its computational advantages as discussed in Section 2.2 and (2) its learning from a state-of-the-science chemical transport model coupled with size-resolved microphysics. In ESMs, where the demand for computational efficiency results in using simplified bulk microphysical treatment, the RFRM can provide a more accurate representation of particle numbers, especially those that mediate aerosol-cloud interactions, while remaining computationally efficient.

Regarding how reasonable is the accuracy vs. speed: a deliberation on this is beyond the scope of the present paper as discussed in our reply to P 2.2.

Referee Point P 2.5 — Hyperparameter tuning (Line 215): Why were these three hyperparameters chosen to be tuned as opposed to the other possible hyperparameters (such as max tree depth)? This section is often framed as balancing accuracy with computational cost, but isn't overfitting also a large concern here?

Reply: Preliminary work (not discussed) involved k-fold cross validation with random and grid searches of the hyperparameter space and determining which are important with smaller samples of data. The three parameters were chosen based on this as well as literature (a comprehensive review by Probst et al. (2019) covers these). True, *max tree depth* is important to consider in optimization of the RFRM, we have implicitly considered this through *min.node.size* (mentioned in Lines 215–216, 249); setting the minimum number of observations in the terminal nodes determines how deep each decision tree in the forest is. Lines 106–109 discussed overfitting; random forests are expected to not suffer from the overfitting issues of decision trees due to the ensemble of random individual models (Breiman, 2001). Further, the model optimization process discussed in Section 3.1 provides further guarantee that overfitting is not a concern by evaluating the model not just with data it is trained on, but also data it has never been exposed to, by additionally optimizing for IQR(MFB) (Lines 217–219), and thus ensuring that performance metrics are consistent. For these reasons, compensatory techniques such as pruning decision trees (that make up the random forest) need not be applied here. To reflect this discussion, the following modified text is in the revised manuscript:

Lines 212–224

Once the data has been selected to train the RFRM, we tune the hyperparameters, which govern the training of the machine learning model. The default implementation of Wright and Zeigler (2017) comes with reasonable (balancing speed and accuracy) choices for these. Based on literature review (Probst et al. (2019) and the references therein) as well as our preliminary examination of RFRMs with varying hyperparameters, we identify the following as most important to optimize—*numtrees*: number of trees in the forest, *mtry*: the minimum number of variables to consider for each split, and *min.node.size*: the minimum node size, i.e. the minimum size of homogeneous data to prevent overfitting. By setting the minimum number of training examples in the terminal nodes of the component trees of the RF, the individual tree depth is controlled, which further mitigates the overfitting associated with decision tree algorithms (discussed in Section 2.2). The default hyperparameter values in Wright and Zeigler (2017) are: *numtrees* = 500, *mtry* = rounded-down square root of the number of variables, and *min.node.size* = 5. We verify if these hyperparameter choices are optimal by performing a grid-search of the hyperparameters and training multiple Random Forest models and not just examining their performance with the training set, but also additionally with the test set. By evaluating the RFRMs with the test set (data that the machine learning algorithm was not exposed to during its training), additional mitigation of possible overfitting is achieved.

Referee Point P 2.6 — The writing in Section 2.2 can be improved. First, the paragraph on the random forest technique should be better cited. In addition, while a random forest can improve overfitting issues over a single decision tree, it certainly does not completely correct overfitting. Finally, I do not think the paragraph at Line 100 (listing different techniques and supervised/unsupervised learning) is necessary.

Reply: Discussed ‘overfitting’ above in P 2.5. Modified accordingly as follows:

Lines 97–117

In the present study, we choose to use the Random Forest (RF) technique (Breiman, 2001) from the large suite of machine learning techniques, for the following reasons: (a) our objective of predicting (regressing) values of [CCN0.4], (b) the ease-of-physical-interpretability of RF models, (c) ease-of-implementation, and (d) the ability to tune this supervised machine learning, which is learning by example.

A Random Forest (Breiman, 2001) is an ensemble of decision trees. A decision tree (Breiman et al., 1984) is a supervised machine learning algorithm that recursively splits the data into subsets based on the input variables that best split the data into homogeneous sets. This is a top-down 'greedy' approach called recursive binary splitting. Decision trees are easy to visualise, are not influenced by missing data or outliers, and are non-parametric. They can, however, overfit on the data. Random Forest modeling is an ensemble technique of growing numerous decision trees from subsets (bags) of the training data and then using all the decision trees to make an aggregated (typically mean) prediction. This approach corrects for the overfitting of single decision trees. Additionally, the bootstrap aggregating (bagging; Breiman, 1996) allows for model validation during training, by evaluating each component tree of the Random Forest with the out-of-bag training examples (training data that was not subsetted in growing the decision tree). Random Forest models are advantageous due to the component decision trees being able to resolve complex non-linear relationships between predictor variables regardless of their inter-dependencies or cross-correlations and the outcome to be predicted. Further, they are relatively easier to visualise and interpret as compared to black-box neural network or deep learning methods. For the purpose of predictions, Random Forest models are one of the most accurate machine learning models with the ability to be trained fast due to the parallelisability of the growth of decision trees. For these reasons, Random Forest is our chosen machine learning tool.

We utilise a fast implementation (Wright and Ziegler, 2017) of Random Forest models (Breiman, 2001) in R (R Core Team, 2020) trained on the GCAPM modeled [CCN0.4] detailed in Section 2.1. Further details of model development and applications are in Section 3.1.

Referee Point P 2.7 — The first two paragraphs of the introduction can be cleaned up. For instance:

- Line 23 reads “these particles or aerosols, or rather CCN”, equates aerosols to CCN which is not technically correct.
- The word “particles” should be used instead of “particulates”.
- Line 25: I suggest changing “directly get into the atmosphere” with “emitted”.
- Line 34: “Aerosol–cloud interactions are through CCN” is vague (“depend on”?). This also neglects to mention the intermediate step of cloud droplet number concentration (the authors point is still valid).

Reply: We have made the following changes (italicized) in the revised manuscript:

Lines 23–26

These *particles/aerosols, or rather CCN* (cloud condensation nuclei: aerosols capable of being imbibed in clouds and modifying their properties) have direct and indirect sources. They can *be directly emitted into the atmosphere* as sea salt, primary inorganic *particulates* such as dust and carbon, or primary organic *particulates*.

Lines 33–34

These *aerosol–cloud interactions are mediated by CCN* that affect cloud micro- and macro-physics primarily through their interaction with water vapor to modify cloud droplet size and number.

We retain "particulates" instead of the suggested "particles" as we feel its meaning of ‘matter in the form of minute separate particles’ is better-suited here.

Technical Comments

Referee Point P 2.8 — Line ∼135: The paragraph citation of Nair et al. (2019) does not seem needed here, especially since this paragraph cites two other papers.

Reply: We retain this to avoid self-plagiarism of the block quote.

Referee Point P 2.9 — Line 209: "The remaining data..": Does the word 'data' here refer to GC model output or observed data?

Reply: Clarified in the revised manuscript:

> **Lines 208–209**
>
> The remaining GCAPM data for 46 sites and 6 vertical levels each are partitioned into training (≈101 million rows) and testing sets (≈44 million rows) in a 7:3 ratio.

Referee Point P 2.10 — Line 327: A tau of 0.36 is pretty low, correct? This seems like a notable decrease in model accuracy.

Reply: Yes, $\tau \approx 0.36$ is in the fair agreement range that we define in Section 2.2. Lines 316–322 & 326–329 discussed the decrease and possible reasons for this.

Referee Point P 2.11 — Line 353: Minor typo (subscript 2.5)

Reply: Changed all instances of "PM2.5" → "$PM_{2.5}$" in the revised manuscript.

References

Breiman, L.: Bagging predictors, Machine Learning, 24, 123–140, https://doi.org/10.1007/bf00058655, 1996.

Breiman, L.: Random forests, Machine Learning, 45, 5–32, https://doi.org/10.1023/A:1010933404324, 2001.

Breiman, L.: Manual for Setting Up, Using, and Understanding Random Forest V4.0, 2003. Available online at https://www.stat.berkeley.edu/~breiman/Using_random_forests_v4.0.pdf

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J.: Classification And Regression Trees, Routledge, https://doi.org/10.1201/9781315139470, 1984.

Probst, P., Wright, M. N., and Boulesteix, A.-L.: Hyperparameters and tuning strategies for random forest, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 9, https://doi.org/10.1002/widm.1301, 2019.

R Core Team: R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria, https://www.R-project.org/, 2020.

Wright, M. N. and Ziegler, A.: ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R, Journal of Statistical Software, 77, https://doi.org/10.18637/jss.v077.i01, 2017.

Interactive comment on Atmos. Chem. Phys. Discuss., https://doi.org/10.5194/acp-2020-509, 2020.

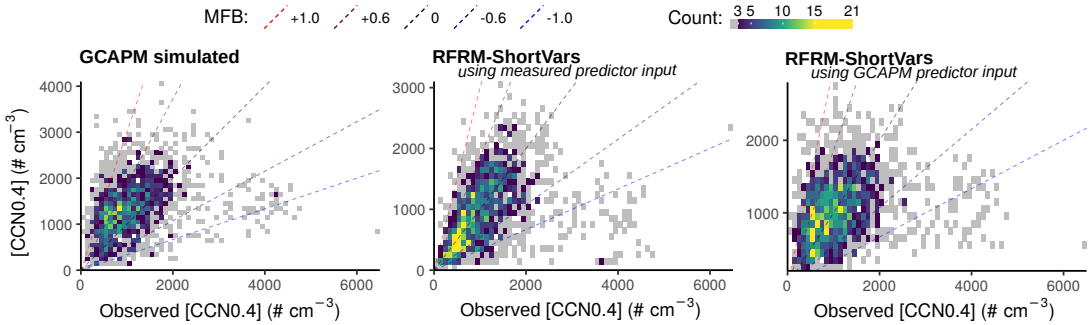

Fig. 1. Fig 12 (b) in the revised manuscript: Performance comparison of the models in quantifying [CCN0.4] compared to its measured values for the SGP site.

Printer-friendly version

Discussion paper