





Article

Using Machine Learning Methods to Identify Particle Types from Doppler Lidar Measurements in Iceland

Shu Yang ^{1,*} , Fengchao Peng ², Sibylle von Löwis ³ , Guðrún Nína Petersen ³  and David Christian Finger ^{1,4,5} 

¹ School of Technology, Reykjavik University, 102 Reykjavik, Iceland; davidf@ru.is

² Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong 999077, China; fpengaa@connect.ust.hk

³ Icelandic Meteorological Office, 105 Reykjavik, Iceland; sibylle@vedur.is (S.v.L.); gnp@vedur.is (G.N.P.)

⁴ Sustainability Institute and Forum (SIF), Reykjavik University, 102 Reykjavik, Iceland

⁵ Energy Institute, Johannes Kepler University, 4040 Linz, Austria

* Correspondence: yangshu.nju09@gmail.com or shu16@ru.is

Abstract: Doppler lidars are used worldwide for wind monitoring and recently also for the detection of aerosols. Automatic algorithms that classify the lidar signals retrieved from lidar measurements are very useful for the users. In this study, we explore the value of machine learning to classify backscattered signals from Doppler lidars using data from Iceland. We combined supervised and unsupervised machine learning algorithms with conventional lidar data processing methods and trained two models to filter noise signals and classify Doppler lidar observations into different classes, including clouds, aerosols and rain. The results reveal a high accuracy for noise identification and aerosols and clouds classification. However, precipitation detection is underestimated. The method was tested on data sets from two instruments during different weather conditions, including three dust storms during the summer of 2019. Our results reveal that this method can provide an efficient, accurate and real-time classification of lidar measurements. Accordingly, we conclude that machine learning can open new opportunities for lidar data end-users, such as aviation safety operators, to monitor dust in the vicinity of airports.

Keywords: lidar; Iceland; machine learning; aerosols



Citation: Yang, S.; Peng, F.; von Löwis, S.; Petersen, G.N.; Finger, D.C. Using Machine Learning Methods to Identify Particle Types from Doppler Lidar Measurements in Iceland.

Remote Sens. **2021**, *13*, 2433.

<https://doi.org/10.3390/rs13132433>

Academic Editor: Simone Lolli

Received: 31 May 2021

Accepted: 20 June 2021

Published: 22 June 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The light detection and ranging system (lidar) is an active remote sensing instrument that is widely used for various purposes: from automatic driving [1] and forestry [2] to aviation safety [3,4]. In the meteorological sector, it is mainly used for wind measurements [3,5] and aerosol detection [6]. While conventional meteorological measurements are either continuous in time but at a single height (e.g., measurements in meteorological masts) or give a profile at a certain time only a few times a day (e.g., radiosondes), a lidar provides a continuous profile measurement with a high temporal and spatial resolution at the same time.

A lidar emits laser beams and receives the backscattered signal, with the detected scatterers being aerosols, cloud droplets and ice particles. A Doppler lidar can also retrieve the radial wind speed based on the Doppler effect by assessing the frequency change of the received signal [3]. The lidar signals can be weakened by high noise levels. The lidar signal becomes attenuated by molecular absorption, which weakens the signal, and subsequently, the noise backscatter signal becomes increasingly dominant with longer distances from the lidar system. A commonly used method to identify noise is to define a threshold of the carrier to noise ratio (CNR) for filtering the data; in some cases, the signal to noise ratio (SNR) is used instead [7–9]. However, in the daily use of lidars, we found this method to be too coarse. We found that the distortion of the lidar signal can

be large, and the noise exceeded the threshold in some cases, or non-noise signals could not reach the thresholds, especially in the free troposphere, where the concentration of scatterers is in general low. Therefore, the lidar needs to be calibrated, which is complex and time-consuming. Foremost, the CNR threshold needs to be optimized and validated to identify the noise. Thus, it is ideal to find a way to identify and filter the background noise observed by a lidar, other than the conventional CNR filtering approach, which could perform better, especially when the intensity of the scatterers is low.

After the background noise is identified, lidar observations of different target classes, such as aerosols and clouds need to be interpreted. Frequently, lookup tables (LUT) are used to classify lidar measurements [10]. LUTs typically contain a combination of variables measured by the lidar, e.g., the lidar ratio, the backscatter coefficients, the color ratio, the temperature and the relative humidity, as well as their respective thresholds [6,11–13]. Some lidar manufacturers, i.e., Leosphere Inc (Vaisala Group), can provide a built-in automatic algorithm to classify the lidar signals [14]. However, based on our experience of using the WindCube lidar from Leosphere, the accuracy of this built-in classification function is rather low. Considering the environmental conditions are varied at different locations, validating this built-in algorithm is also time-consuming and requires high-skill. However, an automatic algorithm that classifies the lidar measurements in real-time, and applies to data from different instruments, is valuable to lidar end users.

Machine learning is a widely used method to classify data [15] and has been used on lidar measurements. Zeng et al. [16] applied a k-means clustering method to classify cloud and aerosols from CALIPSO satellite-based lidar. Brakhasi et al. [17] used the Random Forest (RF) and Support Vector Machine (SVM) algorithm to discriminate CATS-ISS measured aerosols from clouds. Farhani et al. [18] applied both supervised and unsupervised machine learning methods to classify lidar measurements over the UK and identify the anomalies, which were considered as biomass burning aerosols, from other measurements. The machine learning method is considered capable of handling high-dimensional data and map classes with complex characteristics [19]. Although machine learning was used widely in remote sensing research, including lidar data analyses, it has not been applied on the lidar systems in Iceland, partly because this is a rather new approach, and also the performance of Doppler lidars in Iceland was not studied until recently [4,20]. Whether the machine learning method can perform well on the unique lidar systems in Iceland under the subpolar and volcanic environment is still unknown.

Here, we used machine learning algorithms to classify lidar backscatter signals from two Leosphere Doppler wind lidars, with the purpose of aerosol monitoring in Iceland. The sub-polar conditions in Iceland imply several challenges, such as a relatively clean atmosphere with low aerosol concentrations compared to the mid-latitude region, which leads to high noise levels, but also events of intense dust due to wind erosion, volcanic ash clouds, frequent ice clouds and heavy precipitation events [4,21]. This would be the first time that artificial intelligence is applied on the lidars in Iceland and provides an automatic measurement classification. In this study, we propose two separate steps to solve these problems. In the first step, we used the density-based spatial clustering of applications with noise (DBSCAN) clustering algorithm, combined with a conventional method, to identify the noise data. In the second step, we used the Random Forest (RF) method, a well-established machine learning model, to classify the lidar data. Based on our extended experiments in the summer of 2019 (June to August), we show that our method achieves high accuracy and can classify the lidar backscatter in real-time. We will describe the data and methods used in Section 2, present the results in Section 3, discuss the results and give suggestions for future work in Section 4 and summarize the conclusions in Section 5.

2. Methodology

2.1. Instrument and Data Description

Two Leosphere WindCube 200S Doppler lidars are in use in Iceland; both were equipped with a depolarization module during this study period. One is located at Keflavik

International airport, while the other is on a mobile trailer, ready for rapid deployment during volcanic eruptions. During this study, the mobile lidar was located in Reykjavik, the capital city of Iceland. The basic specifications can be found in Table 1. The lidars were scheduled with two scan strategies: (1) Velocity-Azimuth-Display (VAD) scans, the conic-shaped scans with fixed elevation angles for calibration and wind monitoring, and (2) vertical stare for aerosol monitoring. In this study, we only used vertical stare measurements [20]. Here, data from the lidar located in Reykjavik are used to train and test the models, and subsequently, data from the lidar located at Keflavik airport for testing the trained models. In the following section, if not specified otherwise, the lidar observations were collected by the mobile lidar in Reykjavik.

Table 1. The main specifications of the lidars in Iceland.

Model	Windcube 200S
manufacturer	Leosphere Inc.
wavelength (μm)	1.54
detection range (m)	50–14,000
range resolution (m)	25, 50, 75, 100
elevation angle ($^{\circ}$)	–10–90
azimuth angle ($^{\circ}$)	0–360

The primary output of the lidar is the two-dimensional CNR. The relative backscatter coefficients (β) and depolarization ratio (δ) can be retrieved from CNR. The detailed methods to retrieve β and δ are described by Yang et al. [20].

The data we use to train the models were collected from four days in 2019: 14 June, 15 June, 31 July and 1 August. During these days, two dust storms were observed by weather observers, as well as other weather conditions, including low clouds, high clouds and rainfall. A detailed description of these two dust events and the lidar measurements can be found in Yang et al.'s study [20].

A complete lidar profile contains 139 range gates with a resolution of 100 m each. The time difference between the two profiles varied, from around 1.6 to 6 s. To shorten the processing time, 10% of the data were randomly selected for model training.

2.2. Machine Learning Algorithms

Machine learning is a popular method used to achieve some tasks which are relatively easy to perform subjectively for a small data set but are difficult to achieve by conventional algorithms for large data sets, such as the classification of a large number of images. Several algorithms were developed based on Artificial Intelligence and Statistics [15]. In general, there are two approaches, supervised and unsupervised learning. The main difference is that for supervised learning, the data need to be labeled.

The Random Forest (RF) classifier is a supervised ensemble classifier that uses a combination of decision trees to classify the target data set [22]. It is also called the bagged tree method since the data are split into the samples used for training the trees (in-bag samples) and the samples used for internal cross-validation (out-bag samples). The final prediction is made by a majority vote of the trees. Belgiu and Drăguț [23] reviewed the use of RF in remote sensing and revealed that the RF classifier could successfully handle high dimensional data, and at the same time, be fast and insensitive to over-fitting. RF was also used for lidar measurements specifically. Brakhasi et al. [17] used two algorithms, including RF, to discriminate aerosols from clouds in satellite-based lidar measurements, and they found the method had higher accuracy compared to probability distribution function-based algorithms. Liu et al. [24] compared several machine learning algorithms on air pollution (SO_2 and NO_2) classification and found that RF can give a good result. RF

is well-known for its competence in solving data imbalance problems and robustness to data noises [25]. In this study, we use an RF classifier to classify the lidar measurements.

Unlike RF, DBSCAN clustering is an unsupervised algorithm used for unlabeled clustering. It is a popular density-based data clustering algorithm proposed by Ester et al. [26]. The algorithm uses a simple minimum density level estimation, and the objects with a density exceeding a threshold for a number of neighbored data (minPts), within a radius ϵ , will be considered as a core point of a group [27]. The algorithm can be used to separate data points with different densities. Farhani et al. [18] used the DBSCAN method and successfully identified the bio-mass burning aerosols from a lidar data set. We used the DBSCAN with a conventional CNR filter to identify and label the lidar noise data.

We divided this study into two tasks, two steps, and two models—Task 1: to identify noise; Task 2: to classify non-noise measurements. The two steps are step 1: label the training dataset, using DBSCAN algorithm, conventional CNR threshold and conventional LUT method with some manual corrections on obvious odd values; step 2: train two models, noise discrimination model and classification model, with the prepared dataset using the RF algorithm. These two models can fulfill the tasks raised above accordingly, and with two of them, we can classify the lidar data. Besides different labeling methods, another reason for classifying the lidar data with two models is the fact that the amount of noise data points is normally much larger than non-noise data points. This results in a decreased performance of the lidar signal classification model because the noise would dominate the classification. The details of the two models are explained in the following sub-sections.

a. Noise Discrimination Model

The noise of the lidar data is normally identified by CNR thresholds [7–9]. In Iceland, the data quality within the boundary layer is mostly good, and the noise dominates the measurements above the boundary layer, except for the measurements of clouds and precipitation. However, in operation, we found the method of CNR filtering not optimal [4] as the data quality and CNR value are not strictly linearly correlated, i.e., low CNR value does not always represent noise signals. In some cases, the CNR value increases with height in the troposphere, which is unrealistic since fewer aerosols are expected at higher altitudes. Moreover, there are cases when the noise signals spike an unexpectedly high CNR value, which will be kept even after the CNR filtering. The CNR value of the cloud measurements, especially high clouds, are normally lower than the signal within the boundary layer, which makes it difficult to find the “boundary” between clouds and noises. Here, we introduce a new method to optimize noise identification.

Figure 1 illustrates an example of labeling the noise and non-noise data on 31 July and 1 August 2019. We first apply a CNR filter with a threshold of -28 dB, see Figure 1a. The background is not clean, which means the filtered data still contain many data points that should be classified as noise. Subsequently, we apply the DBSCAN method to cluster the remaining data (Figure 1b). The algorithm divides the data points into groups based on their density. Since most of the noise is filtered already, the noise data have a significantly lower density compared to non-noise data. In this way, we can remove the noise (blue dots in Figure 1b) from the data. The two separate groups (yellow and purple in Figure 1b,c) are considered as clouds that are far apart from most non-noise data points within and close to the boundary layer (the orange group Figure 1b,c). By removing all noise data (blue dots), the resulting filtered data have quite a clean background (Figure 1c). The separated cloud groups (yellow and purple) are labeled as high clouds. Here, the definition of “high clouds” is different from the traditional meteorological definition, that is, clouds that are significantly apart from the clouds within and right above the boundary layer. In this case, there are some clouds at 7:00 UTC 31 July, with an altitude of 6 to 8 km, but they can not distinctly be separated from the low clouds within or at the top of the boundary layer (e.g., the clouds at around 2 km in the afternoon of 31 July) and not separated by the DBSCAN cluster, we will not treat them as “high clouds” in this study. These orange data points will be further labeled in the next section and be used as classes to train the classification

model. All the data points are now labeled as noise and non-noise. Since the noise is mainly relevant to CNR, we use CNR as the only feature and the noise/non-noise data as the classes to train an RF model. In this way, a noise discrimination model is trained, and the results are shown together with the classification model in Section 3.

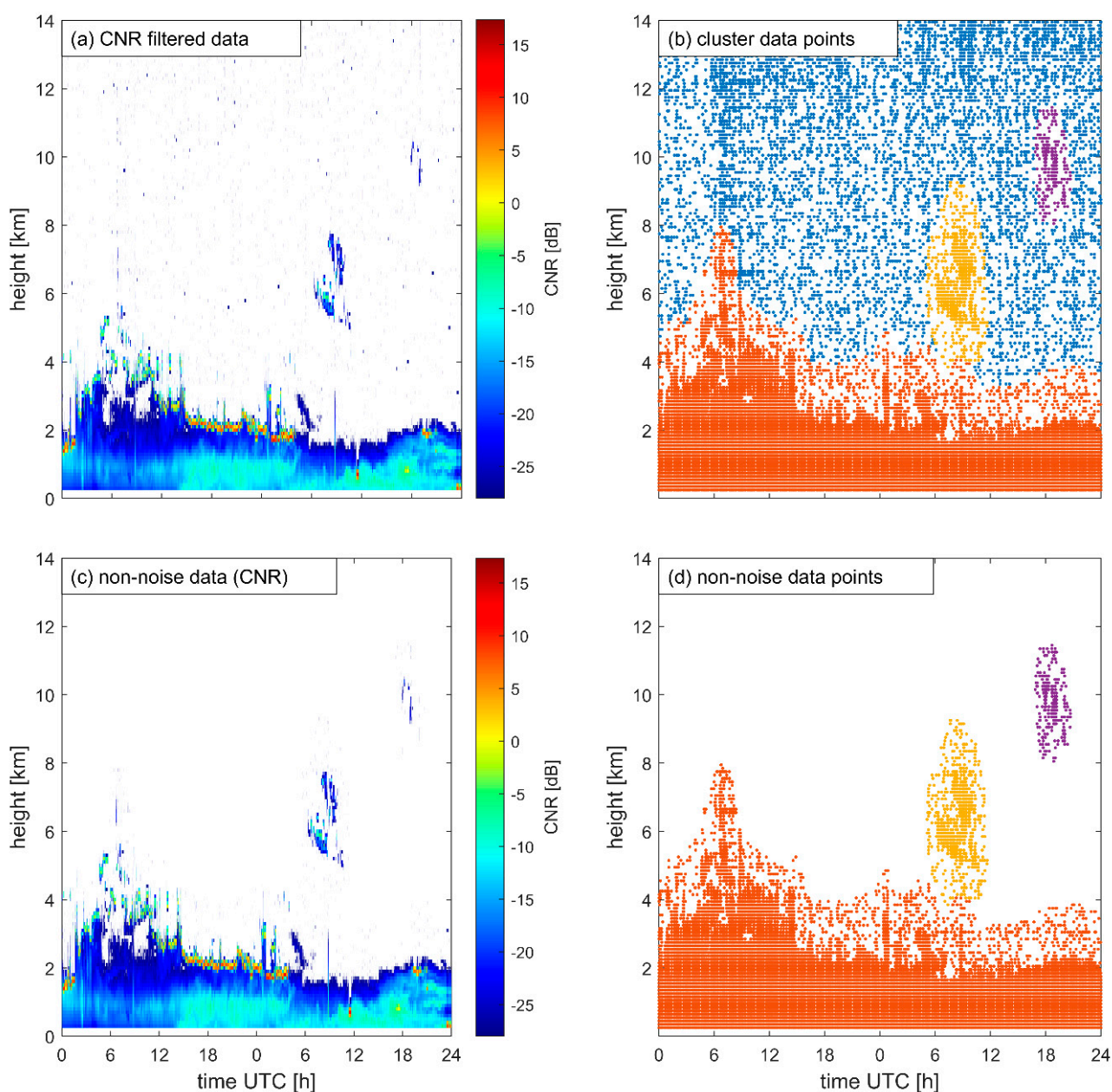


Figure 1. Preparing data for training the noise discrimination model with the DBSCAN clustering algorithm. The left panel (a,c) is the CNR data, while the right panel (b,d) is the clustered data points. All data were collected on 31 July and 1 August 2019. (a) CNR > −28 dB; (b) same data as in (a), but the colors indicate the data groups clustered by the DBSCAN algorithm. Omitting the considered noise data (blue dots in (b)), the filtered data are shown in (c,d). The yellow and purple groups are considered as clouds far above the boundary layer, labeled as high clouds for training the classification model; the orange group is clustered as the same group and considered as the non-noise measurements, which will be further classified later. The x-axis shows the time of 31 July and 1 August 2019.

b. Classification Model

In the frame of a supervised machine learning algorithm, RF needs a labeled data set for training. The four-day data used here contain more than 7.8 million data points, which makes it impossible to label all the data points manually. Based on the weather conditions of these four days [20], we divide the lidar data into eight groups: low clouds (group 1); high clouds (group 2); rain (group 3); aerosol type I (group 4); aerosol type II (group 5); aerosol type III (group 6); others (non-noise, group 7); noise (group 8). The noise is determined by CNR and DBSCAN cluster, as described above, and the high clouds are separated as well during the DBSCAN clustering. Low clouds are the easiest to distinguish. The training dates were all collected during summer 2019. The low clouds are all water clouds, with strong backscatter signals and a low depolarization ratio, usually found at the top of the boundary layer or close to the surface. They are separated by β and δ thresholds from the red dots in Figure 1d. We simply label the clouds that are discriminated from noise by DBSCAN methods above the boundary layer as high clouds (Figure 1d, purple dots). According to [20], three types of aerosols were detected during these four days. All three types are considered to be dust aerosols. We divided the aerosols into three different classes because they have different dust origins, and the water vapor in the atmosphere turned out to have a significant impact on the lidar measurements of dust aerosols, and these three types correspond to three situations [20]. Type I was observed on 14 June and 15 June 2019 when the atmosphere was relatively dry, and lidar measurements showed high β and high δ values. On the morning of 31 July, the atmosphere was dry, and the δ values were high, but β values were relatively low. This is probably because the aerosol load was not heavy at the beginning of the dust event, and the particle size was smaller than type I, and we classify this kind of aerosol as type II. In the afternoon of 31 July, the relative humidity increased, resulting in a lower δ value as the particles absorbed water vapor, and β values were high because of the high aerosol concentration, and we label this kind of aerosol as type III. The type I aerosols are relatively easy to discriminate since both β and δ values are larger than the background, while the type II and III are not, thus there could be some mislabeled type II and type III aerosols.

Rain is the most difficult group to label. Because of its wavelength, the Doppler lidar used here is not sensitive to rain. We classify the rain data based on the low depolarization ratio and a descending movement. If the background depolarization ratio is high, such as in the afternoon of 15 June (see Figure 7), it is easier to identify rain but more difficult when the background depolarization is low as well, such as in the morning of 31 July. Even in the afternoon of 15 June, there were low clouds after 18:00 UTC, and it is difficult to distinguish between the low clouds and precipitation. The remaining data points were labeled as others. Some mislabeled data, e.g., the data points that were labeled as rain without descending movement or weather observation reports with no precipitation, were corrected manually. All these eight classes, the labeling method and the physical explanations can be found in Table 2, and the time-height cross-section of the labeled data is shown in Figure 2.

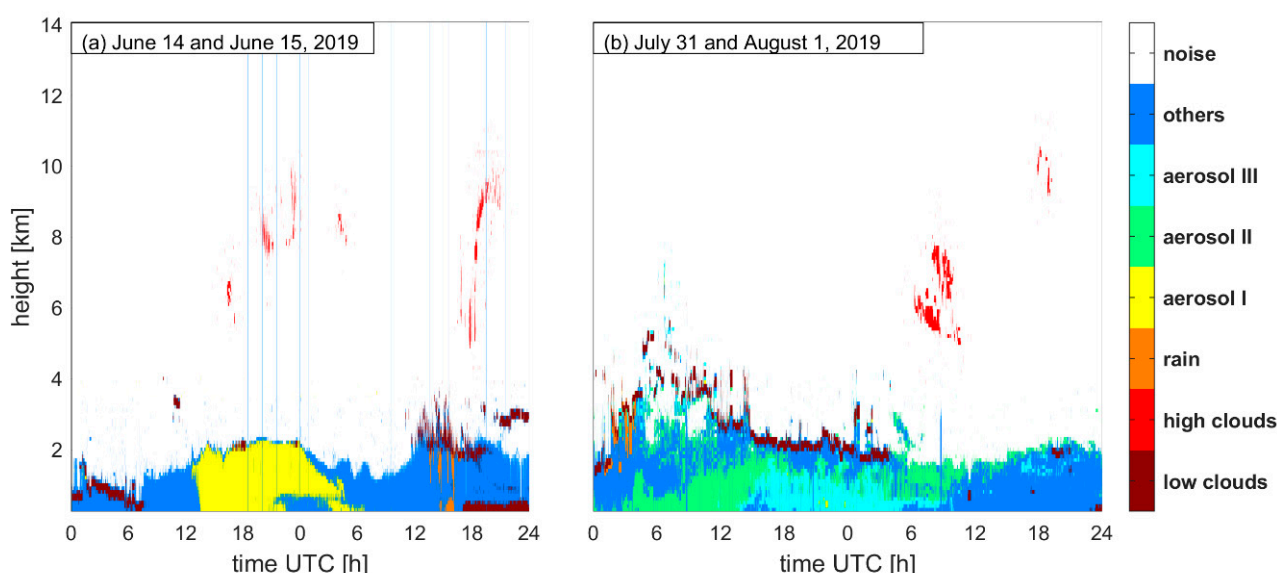
The labeled non-noise data contain the target classes to train, and we select four training features: (i) height, (ii) CNR, (iii) backscatter coefficient (β) and (iv) depolarization ratio (δ). The lidar system broke down and was rebooted then re-calibrated in July 2019, which leads to a slight difference in the absolute measured value between June and July. Here, we normalize each day's lidar data (CNR, β , δ) accordingly to the values with center 0 and standard deviation 1 (same units). Due to the large data size, 10% of randomly selected data points were used to train the model. Figure 3 demonstrates the workflow, from the lidar output to the trained models, including the data clustering, labeling and features used for training.

With the trained noise discrimination model and the classification model, the lidar data are processed in two steps: (1) CNR is used as the input of noise discrimination model to separate the noise and non-noise data points; (2) the height and normalized CNR, β and δ , are used as the input of classification data, to classify the non-noise data points from step one.

Table 2. The eight classes used in this study and their labeling methods and physical explanation.

Group	Class Name	Labeling Method *	Physical Explanation
1	Low clouds	Highest backscatter coefficient; low depolarization ratio	Clouds within or close to the boundary layer; mostly water clouds in the training data set
2	High clouds	CNR threshold; DBSCAN clustering	Clouds in the free troposphere and at higher altitudes; separated from other non-noise data; mostly ice clouds, but could be water clouds
3	Rain	Descending movement; low depolarization ratio	Rain
4	Aerosol type I	High backscatter coefficient; high depolarization ratio **	Dust particles in dry atmosphere; relative large size, high concentration with a non-spherical shape
5	Aerosol type II	Medium-high backscatter coefficient; high depolarization ratio	Dust particles in dry atmosphere; relative small size, low concentration, with a non-spherical shape
6	Aerosol type III	High backscatter coefficient; medium to high depolarization ratio	Dust particles in a humid atmosphere; relative high concentration, the shape is more spherical
7	Others	Unclassified data points	Other data points except for noise data
8	Noise	CNR threshold; DBSCAN clustering	Noise signals

* In addition, some data points are manually corrected. ** High/medium/low values used for labeling are relative values with manual correction, which could be varied from case to case.

**Figure 2.** Labeled lidar data classes on (a) June 14–15, (b) 31 July–1 August 2019. The vertical blue lines are introduced due to measurement gaps. This data will be used to train and validate the models.

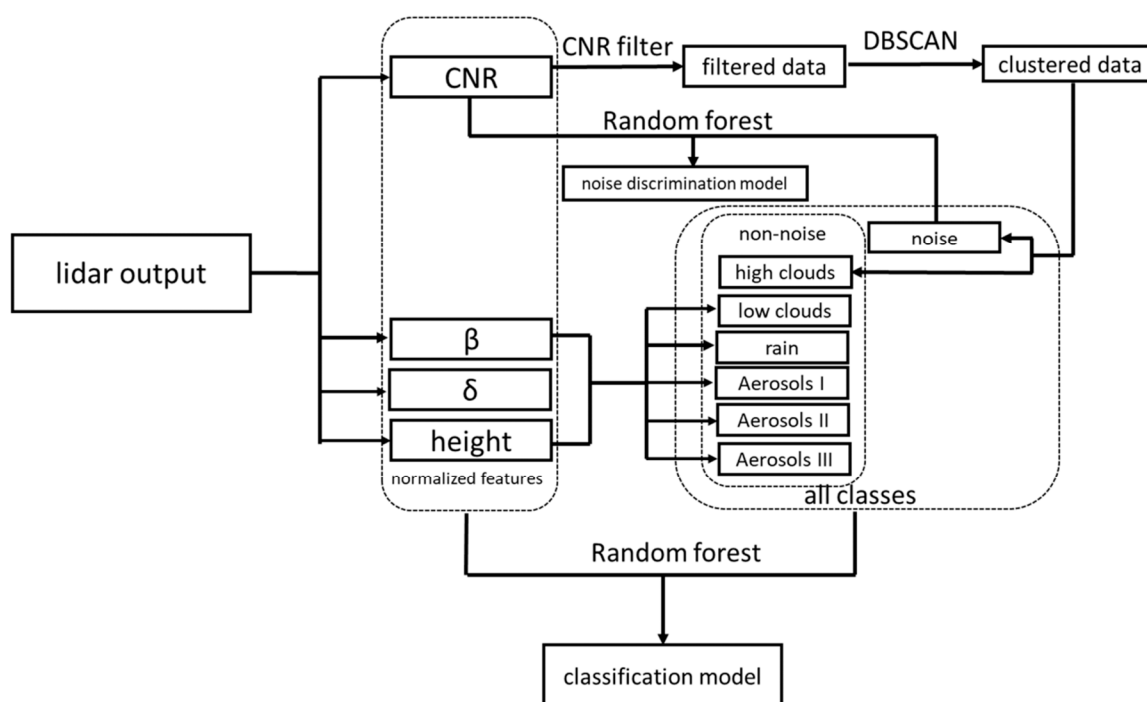


Figure 3. Overview of the process of the method. The text above the arrows is the methods used for that procedure. The noise discrimination model and the classification model are the two trained models.

2.3. Model Performance Evaluation

One way to evaluate the performance of trained models is to use the confusion matrix [28]. For instance, one data set consists of a group of data with two predefined classes: positive (P) and negative (N), and they are the true classes. A classifier will classify this data set as two predicted classes as well. If one data point is positive and it is classified as positive, it is counted as a true positive (TP), but if it is classified as negative, it is counted as a false negative (FN). Similarly, if one data point is negative and classified as negative, it is counted as a true negative (TN), but if it is classified as positive, it is counted as a false positive (FP). In this way, a two-by-two confusion matrix (Figure 4) can be used to evaluate the performance of a classification model. More TPs and TNs mean better performance of the classifier. The accuracy is the rate of all data, how much is correctly classified as Equation (1):

$$\text{accuracy} = \frac{TP + TN}{P + N} \quad (1)$$

For a certain class, the true positive rate (TPR, also called hit rate and recall) of a model [28] is Equation (2):

$$TPR = \frac{\text{correctly classified positives}}{\text{total positives}} = \frac{TP}{P} \quad (2)$$

Similarly, the false-positive rate (FPR, also called false alarm rate) of a model [28] is Equation (3):

$$FPR = \frac{\text{incorrectly classified positives}}{\text{total negatives}} = \frac{FP}{N} \quad (3)$$

For the negative class, the true negative rate (TNR, also called specificity) and false negative rate are given as $TNR = \frac{TN}{N}$ and $FNR = \frac{FN}{P}$. These ratios, TPR, TNR, FPR and FNR, describe the percentages of correctly and incorrectly classified data for each true class. For each predicted class, there are positive predictive values ($PPV = \frac{TP}{TP+FP}$), negative predictive values ($NPV = \frac{TN}{TN+FN}$), false discovery rates ($FDR = \frac{FP}{TP+FP}$) and false omission rates ($FOR = \frac{FN}{TN+FN}$) used to indicate the correctly and incorrectly classification

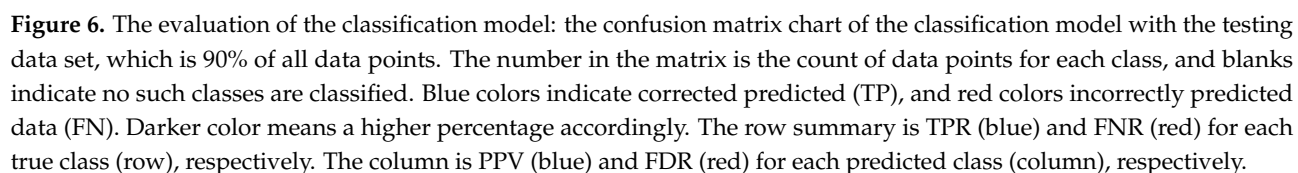
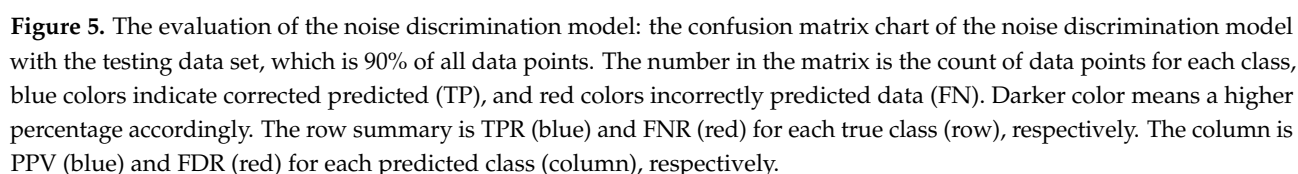
of each predicted class. In Figure 4, these ratios are shown in the row/column summary in percentage. For the data set with more than two classes, we can treat each class as a “positive” class to evaluate, and then the row summary will be TPR vs. FNR, and the column summary will be PPV vs. FDR.

		Predicted class		Row summary	
		positive	negative		
True class	Positive (P)	True Positive (TP)	False Negative (FN)	$TPR = \frac{TP}{P}$	$FNR = \frac{FN}{P}$
	Negative (N)	False Positive (FP)	True Negative (TN)	$TNR = \frac{TN}{N}$	$FPR = \frac{FP}{N}$
		$PPV = \frac{TP}{TP + FP}$	$NPR = \frac{TN}{TN + FN}$		
		$FDR = \frac{FP}{TP + FP}$	$FOR = \frac{FN}{TN + FN}$		
		Column summary			

Figure 4. A concept diagram of a confusion matrix can be used for evaluating the model prediction. True class is the original labeled class, and the predicted class is the model predicted class. Row summary displays the percentages of correctly and incorrectly classified data for each true class. The column summary displays the percentages of correctly and incorrectly classified data for each predicted class.

3. Results

As mentioned in Section 2, randomly selected 10% of the data points were used for the model training. The remaining 90% were used as a testing data set to examine the trained models. The overall accuracy of the trained classification model is 97.3% and 99.3% for the trained noise discrimination model. Figures 5 and 6 show the confusion matrix of test data and predicted results using the trained noise discrimination model and classification model, respectively. The noise discrimination model worked relatively well on separating the noise and non-noise data with high accuracy. Only a very few noise data ($FNR = 0.3\%$) were wrongly classified as non-noise data, and most of them were classified as noise again by the classification model. For the non-noise class, 3.1% of them are classified as noise, but they might not all be wrongly classified. In Figure 1d, there are some points with a lower density, which might be noise but mislabeled.



The performance of the classification model varies, depending on the classes. As mentioned above, a small portion of noise data is not identified by the noise discrimination model, but most of them (TPR = 94.6%) are classified as noise by the classification model. The TPR and PPV value of most classes are larger than 93%, which means the model performs quite well in classifying these classes. The rain class is the worst predicted one, with a TPR of 56% but a higher PPV with a value of 70.9%. In other words, the model may underestimate the rain data by around 40%, but for each predicted rain data point, we have around 70% confidence that it is the rain class.

Since the training and testing data set could be mislabeled, the results by specific cases need to be assessed. Figure 7 shows the predicted classes of 15 June 2019, along with CNR, β and δ values. In this case, the trained model performed well in predicting the classes of clouds and aerosol type I (high β and high δ) and the rain that fell around 15:00 UTC. Some data points are classified as aerosol type III (high β and lower δ) but are not labeled in Figure 2. We cannot say if these points are wrongly predicted. The depolarization ratio was high in the afternoon, and therefore, these could be some remaining aerosols not cleaned out by the precipitation. However, these data points stayed at a similar height, around 600 m, which is an artifact. That is, the lidar was designed to focus at a certain range to maximize the detection distance, which could lead to an artifact high backscatter coefficient layer.

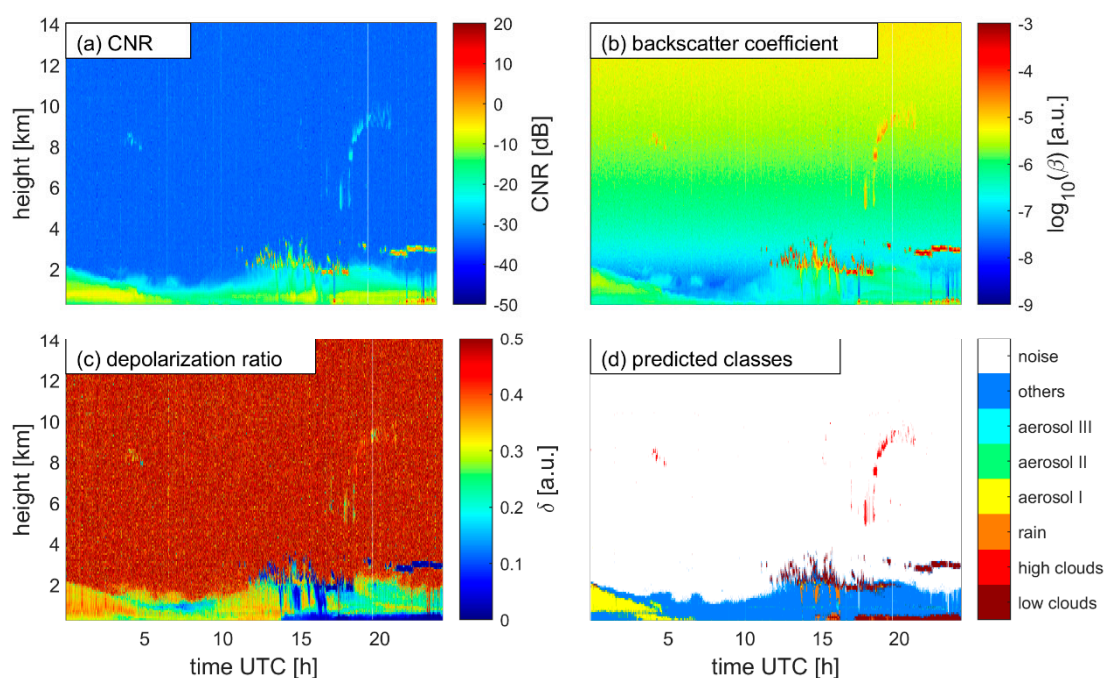


Figure 7. The lidar measurements on 15 June 2019, at Reykjavik (a) CNR, (b) backscatter coefficient, (c) depolarization ratio and the machine learning classification result (d) predicted classes.

As mentioned before, the lidar was re-calibrated and rebooted in July 2019. Accordingly, the observations and classification might be influenced by the re-calibration. However, with a normalized input data set, the trained model can classify the measurements well. Figure 8 shows the results of 31 July. The physical properties of the aerosols are different in this case, compared to those on June 14 and 15. We can find a clear transition from the type II aerosols to the type III aerosols at around 15:00 UTC. The rain in the morning around 03:00 UTC was also classified but is underestimated since the rain data points do not reach the surface. In both examples (15 June and 31 July), we find that there are some missing data due to measurement gaps, which are labeled as others in the training data set but are classified as noise signals here.

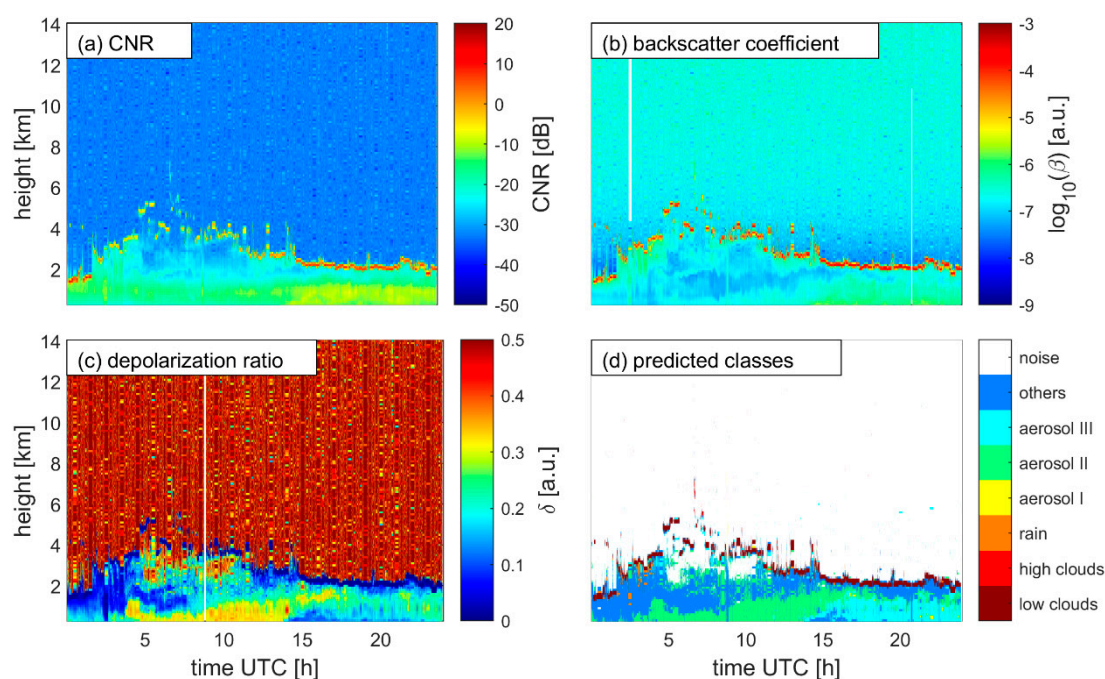


Figure 8. The lidar measurements on 31 July 2019, at Reykjavik (a) CNR, (b) backscatter coefficient, (c) depolarization ratio and the machine learning classification result (d) predicted classes.

These trained models were applied to other data sets. Another dust event was observed from 9 to 10 July 2019. Figure 9 presents the results of the two models based on the measurements on 10 July. It can be seen that there is a mixture of three types of aerosols in the morning but dominated by type II (high δ and lower β). The dust event lasted until the end of the day.

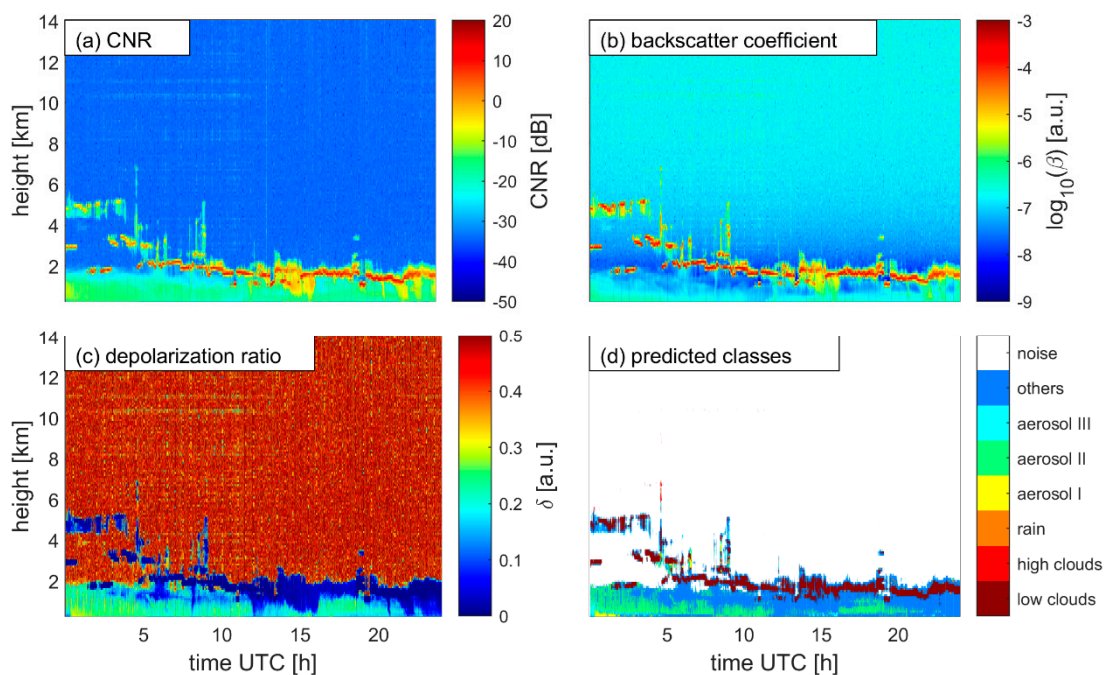


Figure 9. The case with the data not used for model training: the lidar measurements on 10 July 2019, at Reykjavik (a) CNR, (b) backscatter coefficient, (c) depolarization ratio and the machine learning classification result (d) predicted classes.

The same models were applied to the data collected from the lidar at Keflavik airport, which is identical to the lidar in Reykjavik, but the original calibration performed by the manufacturer is expected to be slightly different. Figure 10 shows an example from 31 July 2019. The CNR values of noise data points are slightly lower than from the lidar at Reykjavik, but the models still work well, except for an artifact layer of the “other” class around 600 m above the surface, which, as mentioned earlier, is a result of the focal effect.

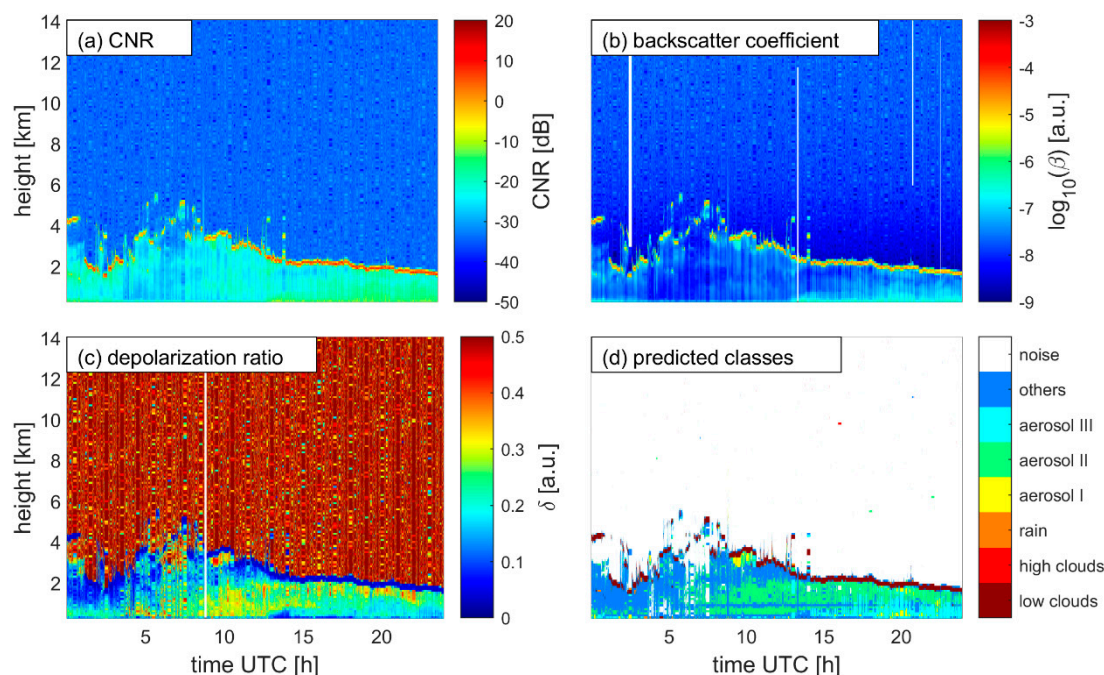


Figure 10. The case with the data collected by the Keflavik lidar: the lidar measurement on 31 July 2019 (a) CNR, (b) backscatter coefficient, (c) depolarization ratio and the machine learning classification result (d) predicted classes.

What should also be noted is that the input data were normalized over four days, but Figures 7–10 used the data normalized over those single days accordingly. In this case, the difference between instruments and calibration parameters can be ignored. More discussion regarding the normalization can be found in the next section.

4. Discussion and Suggestions

One of the well-known features of the machine learning method is that the model itself is a “black-box” to the user. The models trained here can identify the noise signals very well and performs better than the conventional CNR filtering method. However, how did the models achieve that? Did the models apply a similar CNR filtering but with a more precise threshold than the conventional CNR filtering? Table 3 shows the statistics of the data points that are classified as noise by the models from different data sets. If there is a CNR threshold, that would be the maximum value. As we can see, the maximum value varied between data sets. An exception is for the cases of 15 June and 31 July, where they have the same maximum value because they were used for model training together. Besides, these maximums are around -13 to -10 dB, which are much higher than the thresholds normally used (i.e., -32 dB or -28 dB) in previous studies [4,7]. To further explain this, a good example is during the morning of 10 July, which results can be found in Figure 9. There are some horizontal stripes caused by instrumental factors with higher CNR values that should be classified as noise. With a conventional CNR threshold approach, these signals may exceed the threshold and thus be kept after filtering. The trained models classify them as noise signals successfully. Figure 11 presents the model classified data points distribution over different CNR values. Figure 11a shows all data points on that day, classified by the model as noise and non-noise data. There is a boundary between noise and non-noise

data at -28 dB due to the noise labeling method we use (first filter the data with -28 dB then apply DBSCAN method). However, unlike the conventional threshold-based LUT method, there is some overlap around -28 dB. Some classified noise data have CNR values larger than -28 dB, and some non-noise data have CNR values lower than that. Figure 11b presents these data points, which are different from the conventional CNR filtering results. These data have a “long tail”, which means the models do not simply filter the data with a CNR threshold, either the same or different from the value we used to apply, but use a more complicated method.

Table 3. CNR value for model classified noise data points.

Location	Date	CNR of Noise Points (dB)		
		Mean	Minimum	Maximum
Reykjavik *	06/15/2019	-34.11	-41.29	-10.75
Reykjavik	07/10/2019	-33.88	-41.28	-12.82
Reykjavik *	07/31/2019	-32.40	-40.25	-10.75
Keflavik	07/31/2019	-33.92	-49.79	-12.82

* Data used to train the model.

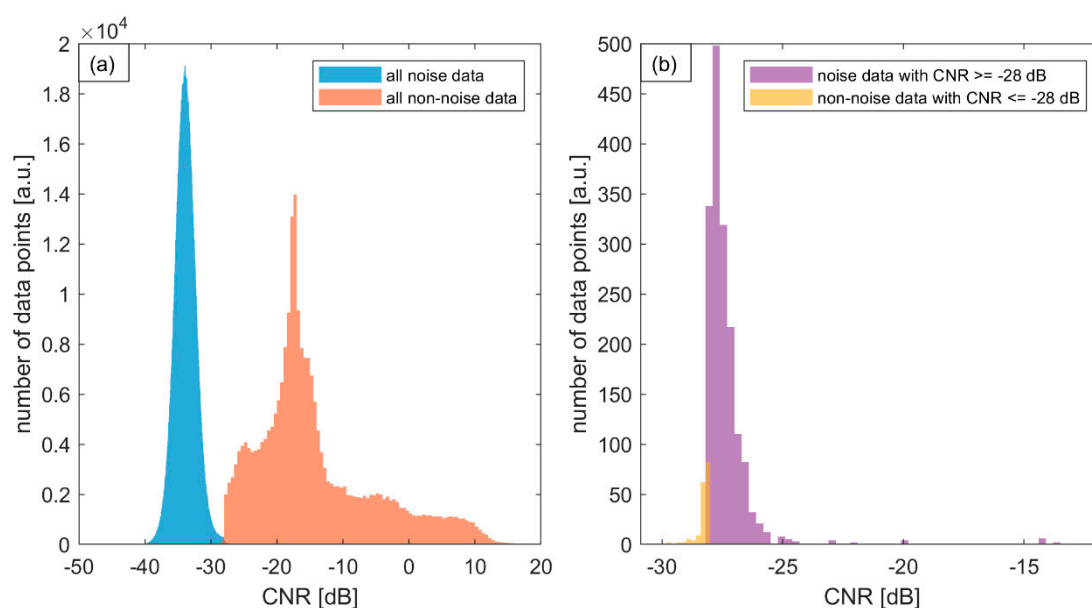


Figure 11. Model classified data points distribution as a function of CNR. Data collected on 10 July 2019 at Reykjavik. (a) All data are classified by the model as noise (blue) and non-noise (orange). (b) specifically, only the data points classified differently by the model than by the conventional CNR threshold method: the data points classified as noise but with CNR greater or equal to -28 dB (purple) and the data points classified as non-noise but with CNR lower or equal to -28 dB (yellow).

As a supervised machine learning, the performance of RF is directly related to the labeled data set used for training. The accuracy of the noise discrimination model is higher than the classification model, partly because the noise of lidar measurements is better studied and easier to label compared to other classes. For non-noise classes, the rain classification has the worst performance; one reason is that this lidar is not sensitive to rain, and the way we label the rain data is mainly by descending movements, which could be hard to distinguish if the β or δ values are similar to that of the background. Moreover, the rain data are relatively rare in the training data set we use. For rain classification, training on extended data sets with more rain events is needed. Including data from other instruments, such as the ceilometer and rain gauge next to the lidar, might also

help to label the rain data. Another way forward is to explore the possibility of detecting solid precipitation in wintertime. For cloud classification, the method in this study is quite promising. However, for now, we only distinguish the low and high clouds based on different labeling methods. A possible improvement to be implemented would be to include more data collected in wintertime with low ice clouds and snow. Hopefully, the model can distinguish water clouds and ice clouds, and it could be more meaningful for meteorological monitoring and research. For the classification of aerosols, the major challenge is having a better understanding of lidar aerosols measurements. For example, the boundary between aerosol types II and III on 31 July is not entirely clear, and we do not know the exact physical difference between these aerosol types yet. Unlike type I, type II and III are aerosols considered sharing the same origin, which is the west of the Icelandic highlands. The main difference is due to the i) weather conditions and ii) aerosols concentration: type II had a lower concentration in a relatively dry atmosphere, so the δ value was high but β value was lower, while type III had a higher concentration in a more humid atmosphere. These features are not necessary for aerosols classification, any particles with similar observations could be identified as these types of aerosols, and this can be checked in operations. For possible eruptions in Iceland, the two models can identify the signals from volcanic ash with similar properties as these three types of aerosols but may miss the ash particles that are different. In general, a larger data set with more precise labels is recommended for improving the performance of the lidar classification model. For the “others” class, which is considered as the observations when the backscattered lidar signal is strong enough (not noise), and no special phenomenon (clouds, rain, aerosols) was identified, i.e., the air should be relatively clean air within the boundary layer. As we mentioned above, the rain class was underestimated, which means some rain data were classified as others. Besides the rain, some low clouds are also difficult to distinguish and label, which means some low clouds may be mislabeled and classified as others. However, compared to the rain class, this mislabel issue is of little importance since the lidar signal is more sensitive to cloud than rain droplets. A better labeling data set could improve this.

Neither model uses measurement time as the input feature; that is, the classification is not time-dependent. This means that these models can be deployed for real-time classification. This was examined by compiling a profile-by-profile classification (not shown). The features of the classification model are normalized because the lidar calibrations are different in June and July 2019. From an operational perspective, if a lidar has stable operations, the model can be re-trained without normalization. Otherwise, the newly measured data need to be normalized with the data measured 6–24 h before the event. Using the normalized data, the models will try to identify the different classes based on the relative values, which can minimize the impact of shifting between different instruments or calibration parameters. What also needs to be kept in mind for applying this method is that the models are sensitive to lidar calibration parameters, so a well-calibrated lidar system is needed.

In this study, we trained the model with 4-days measurements. Based on our results, it is advisable to apply the presented machine learning algorithms during operations and explore the application on a larger data set, with a wider range of atmospheric conditions, including winter measurements, when the boundary layer can be shallower and the precipitation liquid and/or frozen. This would lead to a continuously improved classification of the lidar observations.

5. Conclusions

In this study, we applied machine learning methods to classify the signals retrieved from lidar measurements in Iceland. The first challenge is to label the lidar data for the supervised machine learning algorithm. We used an unsupervised data clustering method, DBSCAN, and combined it with the conventional threshold-based method and manual correction. We applied the method to classify four days of data obtained during two dust events in 2019. Subsequently, we use this labeled data set to train two models, the noise

discrimination model and the classification model with the RF algorithm. With these two trained models, we can accurately identify the noise data and classify the lidar data into eight classes, including three types of aerosols, two types of clouds and rain. For all classes except rain, the true positive rate is higher than 94%; however, the model is underpredicting rain. In most cases, the model can identify rain correctly but cannot identify all rain data points. The accuracy of model prediction is directly related to labeling accuracy. With more accurately labeled lidar data, the classification model can be further improved. A larger data set with more and different weather conditions might also improve the performance of the model. In addition, the calibration and correction of lidar data can also affect the results. We found an artifact layer at around 600 m due to the lack of focal correction of the lidar depolarization channel in some cases.

This method was tested on different data sets from different dates and events, as well as from two identical Doppler lidars in Iceland, and the results remained valid and robust. This suggests that the machine learning method can be used for the classification of lidar measurements.

Machine learning algorithms depend on the training data sets. Based on our results, we conclude that a continuous application of new training data would improve the results further. Accordingly, the presented algorithms may be useful to interpret in real-time lidar observations and provide valuable information to end-users, such as aviation service providers and air quality observers.

Author Contributions: Conceptualization, S.Y., F.P., G.N.P., S.v.L. and D.C.F.; methodology, S.Y. and F.P.; programming, S.Y.; resources, G.N.P., S.v.L. and D.C.F.; data curation, S.Y.; writing—original draft preparation, S.Y.; writing—review and editing, F.P., G.N.P., S.v.L. and D.C.F.; visualization, S.Y.; supervision, D.C.F., G.N.P. and S.v.L.; project administration, D.C.F.; funding acquisition, D.C.F. All authors have read and agreed to the published version of the manuscript.

Funding: This study is funded by ISAVIA, the aviation authority in Iceland.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Gao, H.; Cheng, B.; Wang, J.; Li, K.; Zhao, J.; Li, D. Object Classification Using CNN-Based Fusion of Vision and LIDAR in Autonomous Vehicle Environment. *IEEE Trans. Ind. Inform.* **2018**, *14*, 4224–4231. [\[CrossRef\]](#)
2. Dubayah, R.O.; Drake, J.B. Lidar Remote Sensing for Forestry. *J. For.* **2000**, *98*, 44–46. [\[CrossRef\]](#)
3. Thobois, L.; Cariou, J.P.; Gultepe, I. Review of Lidar-Based Applications for Aviation Weather. *Pure Appl. Geophys.* **2019**, *176*, 1959–1976. [\[CrossRef\]](#)
4. Yang, S.; Petersen, G.N.; von Löwis, S.; Preißler, J.; Finger, D.C. Determination of Eddy Dissipation Rate by Doppler Lidar in Reykjavik, Iceland. *Meteorol. Appl.* **2020**, *27*, e1951. [\[CrossRef\]](#)
5. Bilbro, J.; Fichtl, G.; Fitzjarrald, D.; Krause, M.; Lee, R. Airborne Doppler Lidar Wind Field Measurements. *Bull. Am. Meteorol. Soc.* **1984**, *65*, 348–359. [\[CrossRef\]](#)
6. Ansmann, A.; Müller, D. Lidar and Atmospheric Aerosol Particles. In *Lidar: Range-Resolved Optical Remote Sensing of the Atmosphere*; Weitkamp, C., Ed.; Springer Series in Optical Sciences; Springer: New York, NY, USA, 2005; pp. 105–141. ISBN 978-0-387-25101-1.
7. Boquet, M.; Royer, P.; Cariou, J.-P.; Machta, M.; Valla, M. Simulation of Doppler Lidar Measurement Range and Data Availability. *J. Atmos. Ocean. Technol.* **2016**, *33*, 977–987. [\[CrossRef\]](#)
8. Gryning, S.-E.; Floors, R.; Peña, A.; Batchvarova, E.; Brümmer, B. Weibull Wind-Speed Distribution Parameters Derived from a Combination of Wind-Lidar and Tall-Mast Measurements Over Land, Coastal and Marine Sites. *Bound. Layer Meteorol.* **2016**, *159*, 329–348. [\[CrossRef\]](#)
9. Gryning, S.-E.; Mikkelsen, T.; Baehr, C.; Dabas, A.; Gómez, P.; O'Connor, E.; Rottner, L.; Sjöholm, M.; Suomi, I.; Vasiljević, N. Measurement methodologies for wind energy based on ground-level remote sensing. In *Renewable Energy Forecasting*; Elsevier: Sawston, Cambridge, 2017; pp. 29–56. ISBN 978-0-08-100504-0.

10. Di Noia, A.; Hasekamp, O.P. Neural Networks and Support Vector Machines and Their Application to Aerosol and Cloud Remote Sensing: A Review. In *Springer Series in Light Scattering: Volume 1: Multiple Light Scattering, Radiative Transfer and Remote Sensing*; Kokhanovsky, A., Ed.; Springer Series in Light Scattering; Springer International Publishing: Cham, Germany, 2018; pp. 279–329. ISBN 978-3-319-70796-9.
11. Groß, S.; Tesche, M.; Freudenthaler, V.; Toledano, C.; Wiegner, M.; Ansmann, A.; Althausen, D.; Seefeldner, M. Characterization of Saharan Dust, Marine Aerosols and Mixtures of Biomass-Burning Aerosols and Dust by Means of Multi-Wavelength Depolarization and Raman Lidar Measurements during SAMUM 2. *Tellus B Chem. Phys. Meteorol.* **2011**, *63*, 706–724. [\[CrossRef\]](#)
12. Chouza, F.; Reitebuch, O.; Groß, S.; Rahm, S.; Freudenthaler, V.; Toledano, C.; Weinzierl, B. Retrieval of Aerosol Backscatter and Extinction from Airborne Coherent Doppler Wind Lidar Measurements. *Atmos. Meas. Tech.* **2015**, *8*, 2909–2926. [\[CrossRef\]](#)
13. Haarig, M.; Ansmann, A.; Gasteiger, J.; Kandler, K.; Althausen, D.; Baars, H.; Radenz, M.; Farrell, D.A. Dry versus Wet Marine Particle Optical Properties: RH Dependence of Depolarization Ratio, Backscatter, and Extinction from Multiwavelength Lidar Measurements during SALTRACE. *Atmos. Chem. Phys.* **2017**, *17*, 14199–14217. [\[CrossRef\]](#)
14. Leosphere, Inc. *WINDCUBE 100s-200s User Manual Version 2*; Leosphere Inc.: Orsay, France, 2013.
15. Kotsiantis, S.B.; Zaharakis, I.D.; Pintelas, P.E. Machine Learning: A Review of Classification and Combining Techniques. *Artif. Intell. Rev.* **2006**, *26*, 159–190. [\[CrossRef\]](#)
16. Zeng, S.; Vaughan, M.; Liu, Z.; Trepte, C.; Kar, J.; Omar, A.; Winker, D.; Lucker, P.; Hu, Y.; Getzewich, B.; et al. Application of High-Dimensional Fuzzy k-Means Cluster Analysis to CALIOP/CALIPSO Version 4.1 Cloud–Aerosol Discrimination. *Atmos. Meas. Tech.* **2019**, *12*, 2261–2285. [\[CrossRef\]](#)
17. Brakhasi, F.; Hajeb, M.; Fouladinejad, F. Discrimination Aerosol Form Clouds Using Cats-Iss Lidar Observations Based on Random Forest and Svm Algorithms over The Eastern Part of Middle East. *ISPRS Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2019**, *XLII-4/W18*, 235–240. [\[CrossRef\]](#)
18. Farhani, G.; Sica, R.J.; Daley, M.J. Classification of Lidar Measurements Using Supervised and Unsupervised Machine Learning Methods. *Atmos. Meas. Tech. Discuss.* **2020**, 1–18. [\[CrossRef\]](#)
19. Maxwell, A.E.; Warner, T.A.; Fang, F. Implementation of Machine-Learning Classification in Remote Sensing: An Applied Review. *Int. J. Remote Sens.* **2018**, *39*, 2784–2817. [\[CrossRef\]](#)
20. Yang, S.; Preißler, J.; Wiegner, M.; von Löwis, S.; Petersen, G.N.; Parks, M.M.; Finger, D.C. Monitoring Dust Events Using Doppler Lidar and Ceilometer in Iceland. *Atmosphere* **2020**, *11*, 1294. [\[CrossRef\]](#)
21. Ólafsson, H.; Furger, M.; Brümmer, B. The Weather and Climate of Iceland. *Meteorol. Z.* **2007**, *16*, 5–8. [\[CrossRef\]](#)
22. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [\[CrossRef\]](#)
23. Belgiu, M.; Drăguț, L. Random Forest in Remote Sensing: A Review of Applications and Future Directions. *ISPRS J. Photogramm. Remote Sens.* **2016**, *114*, 24–31. [\[CrossRef\]](#)
24. Liu, B.; Xu, Z.; Kang, Y.; Cao, Y.; Pei, L. Air Pollution Lidar Signals Classification Based on Machine Learning Methods. In Proceedings of the 2020 39th Chinese Control Conference (CCC), Shenyang, China, 27–29 July 2020; pp. 13–18.
25. Mellor, A.; Boukir, S.; Haywood, A.; Jones, S. Exploring Issues of Training Data Imbalance and Mislabelling on Random Forest Performance for Large Area Land Cover Classification Using the Ensemble Margin. *ISPRS J. Photogramm. Remote Sens.* **2015**, *105*, 155–168. [\[CrossRef\]](#)
26. Ester, M.; Kriegel, H.-P.; Sander, J.; Xu, X. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, Portland, OR, USA, 2 August 1996; pp. 226–231.
27. Schubert, E.; Sander, J.; Ester, M.; Kriegel, H.P.; Xu, X. DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN. *ACM Trans. Database Syst.* **2017**, *42*, 19:1–19:21. [\[CrossRef\]](#)
28. Fawcett, T. An Introduction to ROC Analysis. *Pattern Recognit. Lett.* **2006**, *27*, 861–874. [\[CrossRef\]](#)