atmosphere

MDPI

Article

Data Filling of Micrometeorological Variables in Complex Terrain for High-Resolution Nowcasting

Nipun Gunawardena [1,*], Pierre Durand [2], Thierry Hedde [3], Florian Dupuy [4] and Eric Pardyjak [5]

1 Lawrence Livermore National Laboratory, Livermore, CA 94550, USA
2 Laboratoire d'Aérologie, Université de Toulouse, CNRS, UPS, 31400 Toulouse, France; pierre.durand@aero.obs-mip.fr
3 Institut de Recherche sur les Systèmes Nucléaires pour la Production D'énergie bas Carbone, Commissariat à l'Energie Atomique et aux Energies Alternatives, 13108 Saint-Paul-Lez-Durance, France; thierry.hedde@cea.fr
4 Météo-France, Direction des Opérations pour la Production and Centre National de Recherches Météorologiques (CNRM-GAME), Météo-France/CNRS URA 1357, 31057 Toulouse, France; florian.dupuy@meteo.fr
5 Department of Mechanical Engineering, University of Utah, Salt Lake City, UT 84112, USA; pardyjak@eng.utah.edu
* Correspondence: gunawardena1@llnl.gov; Tel.: +1-801-664-9931

Abstract: In this paper, two different computationally inexpensive methods for nowcasting/data filling spatially varying meteorological variables (wind velocity components, specific humidity, and virtual potential temperature) covering scales ranging from 100 m to 5 km in regions marked by complex terrain are compared. Multivariable linear regression and artificial neural networks are used to predict micrometeorological variables at eight locations using the measurements from three nearby weather stations. The models are trained using data gathered from a system of eleven low-cost automated weather stations that were deployed in the Cadarache Valley of southeastern France from December 2016 to June 2017. The models are tested on two held-out periods of measurements of thermally-driven flow and synoptically forced flow. It is found that the models have statistically significant performance differences for the wind components during the synoptically driven flow period ($p = 6.6 \times 10^{-3}$ and $p = 2.0 \times 10^{-2}$ for U and V, respectively), but perform the same otherwise. These methods can be used to spatially fill gaps in micrometeorological datasets. Recommended future work should include statistically interpreting the predictive models and testing their capabilities on meteorological datasets from different locations.

Keywords: boundary layer; artificial neural networks; multiple linear regression; prediction

check for updates

Citation: Gunawardena, N.; Pardyjak, E.; Durand, P.; Hedde, T.; Dupuy, F. Data Filling of Micrometeorological Variables in Complex Terrain for High-Resolution Nowcasting. *Atmosphere* **2022**, *13*, 408. https://doi.org/10.3390/atmos13030408

Academic Editor: Stephan Havemann

Received: 28 October 2021
Accepted: 26 February 2022
Published: 2 March 2022

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

Copyright: © 2022 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

To study meteorology in regions of complex terrains, such as urban or mountainous areas, researchers often conduct field experiments. During these experiments, large amounts of data are typically collected with a wide variety of instruments, including tethered balloons, radiosondes, manned and unmanned aircraft, remote sensing instruments (LIDAR, SODAR, Radio Acoustic Sounding Systems, etc.), meteorological towers, and small, distributed weather stations [1]. Field experiments typically last from a few weeks [2] to a few months [3,4], to even a few years [5]. The instrumentation is often removed after the field experiment is completed, eliminating the ability to make more observations in that area. However, in many cases, the field experiment is conducted in an area that has permanent weather stations installed. For example, the MATERHORN experiment [6] at the U.S. Army Dugway Proving Ground in Utah, the BLLAST experiment in southern France [7], and the KASCADE experiment described herein were all field experiments where the scientific equipment used supplemented permanent operational weather stations.

In this work, we use artificial neural network and multiple linear regression methods to predict the measurements at a fixed sensor station using the measurements from other stations in the area. This type of modeling is motivated by the following needs:

1. Gap filling. If a sensor in an array becomes inoperative for some reason, the missing values from that station can be filled by the other stations in the area.
2. Greater use of limited experimental resources. At the end of a field campaign, temporarily deployed instruments are removed, while permanent weather stations remain. The data from the permanent weather stations can be used to predict observations at the locations where the stations were removed, increasing operational capacity and the amount of data available for modeling and science. Increased data, especially in complex terrain, where spatial variability is high over small length scales, can allow for a better understanding of atmospheric physics and can improve forecasting ability of phenomena that depend on weak-wind variability as well as small temperature and humidity differences, such as dispersion [8,9] and frost or fog formation [10].

There are two key points that differentiate our work from related work, which are described in Section 2. First, we demonstrate that these methods work on small temporal and spatial scales in complex terrains, a regime where there are significant measurement variabilities due to surface heterogeneity. Second, we demonstrate the effectiveness of multiple linear regression techniques, which are simple, easy to interpret, and computationally inexpensive (so much that they can be run on low-powered stations themselves). The rest of this paper is divided into the following sections: background, methods, experiments and results, future work, and conclusions. In addition, an appendix is included where supplemental tests are presented.

2. Background

There are numerous reasons for studying the meteorology of complex terrain, which have been discussed previously [6,11–13]. The atmospheric physics associated with complex terrain inherently violates many assumptions typically invoked in idealized theory. For example, the Monin–Obukhov similarity theory and its assumptions are often violated in complex terrains [14–16]. In addition, studying complex terrain is critical to improving numerical weather prediction capabilities [12] to address issues that are critical to humans. For example, over half of the world's population lives in cities [17], and air pollution is considered a serious human health risk [18]. Studying and understanding mountain-meteorological phenomena such as cold air pools [19] and valley flows [20] is necessary for developing air-pollution mitigation strategies as well as forecasting fog formation [21]. The nuclear energy community is required to consider the side-effects of a breach and how the contaminants spread, which is affected by complex terrain [3,22,23]. Improved weather prediction in complex terrain is also applicable to forecasting mountain waves [11] as well as snow and ice storms [12]. Adams et al. [24] estimate that improved snow prediction in the United States could potentially produce 1.3 billion US dollars of benefit annually, in addition to the number of lives saved due to prevented accidents. In considering all of these applications, it is clear that one of the features that complex the atmosphere terrain atmospheres exhibit is high spatial and temporal variability [25,26], which can make weather prediction difficult from micro- to mesoscales. It is common to use statistical methods to post-process weather data of all scales, and here we use two different methods to do so.

The two methods described in this paper are artificial neural networks and multiple linear regression. As outlined in the book by Shalev-Shwartz and Ben-David [27], artificial neural networks (ANNs) are a class of biologically-inspired algorithms that can be used for regression or classification tasks. While there are many types of ANNs, the simplest is the multilayer perceptron, also known as a standard feedforward neural network.

A feedforward neural network is a directed acyclic graph where the nodes are called nodes (or neurons), and the edges are called connections. Each node has an associated bias and activation function. Each connection has an associated weight. The graph is organized into layers, where each layer contains a certain amount of nodes and is connected to the

layer above and below it. The bottommost layer is called the input layer, and any gathered data are passed into the nodes of the input layer. The input layer has as many nodes as there are input variables. The topmost layer is called the output layer, and is the output of the entire feedforward network. There can be multiple outputs, and the number of nodes in the output layer is equal to the number of target variables. The layers in-between the input layer and the output layer are called the hidden layers. Therefore, data are fed into the input layer; its output is then fed into the first hidden layer, whose output goes into the second hidden layer, and so on, until the output from the last is sent to the output layer. The nodes in any given layer can be referred to by the layer name, e.g., the nodes in the hidden layer can be called hidden nodes [28–30]. The output of any given node is the weighted sum of all the nodes in the layer before it, passed through the activation function. The values of the weights and biases are found by minimizing the network's mean-squared error, in a process known as backpropagation. The process of finding the mean-squared error using known data is called training, while using the ANN to predict unknown values is called testing. There are more complicated ANNs, known as deep neural networks, but those will not be discussed in this work. More information can be found in Shalev-Shwartz and Ben-David [27].

A multiple-linear-regression (MLR) model is a statistical tool where a set of explanatory variables linearly models one or more target variables [27]. Linear regressions with a single target variable take the form of $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p$, where y is the target variable, x_p are the explanatory variables, and β_p are the regression coefficients. Since the MLR model is considered a basic statistical tool, the general topic will not be explained here any further. A more in-depth discussion of the method can be found in the book by Shalev-Shwartz and Ben-David [27].

ANNs and MLR models are general techniques that can be applied to a wide variety of topics. In our case, we are using them to perform predictions in space, though others have used these methods to predict in time and space. Since predicting values in time or space is an incredibly broad categorization, there are several terms that the literature uses to describe this process. Here we review a few of these terms, methods, and literature.

One of these terms is *nowcasting*, which is a process that describes the current and near-future (hourly scale) state of the atmosphere [31]. Nowcasting has been used in many different areas of meteorology. It has been used to nowcast storms [32–34], ice [35], ice fog [36], precipitation and flash floods [37,38], and tropical cyclones [39]. These examples, however, are quite different from ours due to the time scales, spatial scales, and equipment involved. Therefore, we will mostly review nowcasting systems in the literature that use distributed sensor stations or wireless-sensor networks. (For a background on distributed sensor stations, see Gunawardena et al. [40].)

Öztopal [41] used an ANN to predict the wind speed at one station, given the wind speeds at nine different stations. While this is similar to the work presented herein, Öztopal [41] did it on a much larger spatial scale. Their stations were distributed on a scale of hundreds of kilometers. Philippopoulos and Deligiorgi [5] also used ANNs to predict wind speeds, but on the spatial scale of tens of kilometers. They compared the ANN's performance on several spatial interpolation methods. In this paper, we compare our ANN performance to a multiple linear regression performance, which they did not do. In addition, we show that an ANN can predict variables other than wind speed on a much shorter time scale.

Benvenuto and Marani [42] use an air quality monitoring network in Mestre, Italy, and ANNs to nowcast pollution concentrations in the near future and to interpolate missing data. While their work is similar to ours, there are some key differences. They nowcast the near future (one hour ahead and three hours ahead), while we predict the present values. Benvenuto et al. also interpolate missing data using an ANN that predicts forward in time. Here, we show that the ANN does not have to predict forward in time to interpolate missing data. Finally, Benvenuto et al. used data gathered in an urban area, whereas we

used data gathered in a vegetated valley. Videnova et al. [43] also used an ANN to predict air pollution in their work, and they also predicted forward in time.

Another term that is often used to describe the work we are doing herein is gap filling. Gap filling is concerned with filling the gaps in data, often time series data. The simplest and most common method of gap filling is simply linear interpolation between two non-adjacent points in a time series. Gap filling has been used to increase environmental data in several different studies [44,45], and is often used as an intermediate step when performing complex analyses [46]. Moffat et al. [47] present an overview of many of these methods. While some of these existing gap-filling methods can be used with data from a single station, many of the same methods also require lookup tables or physical models, neither of which are required by the methods we present here. Tardivo and Berti [48], Kemp et al. [49], and Coutinho et al. [50] all present nowcasting methods that use data from multiple stations, and are based on regression techniques. However, none of them show that the methods work on a very small spatial and temporal scale, and Coutinho et al. [50] only show successful predictions for maximum temperature and relative humidity. Furthermore, the regression methods presented in Tardivo and Berti [48] are more complicated than the ones we present herein, since they use adaptive regression methods.

Data assimilation is a field in which experimental data are incorporated into forecast models to improve analyses [51]. There are many techniques used within data assimilation to do gap filling. There are two big differences between the techniques used in data assimilation and our methods. First, data assimilation methods are often very complicated, and rely on an understanding of the underlying physics. Even a simple Kalman filter depends on an underlying physical model to work properly [52]. Our methods are purely statistical, and in the case of MLR, relatively easy to run. The second main difference is the application. Our methods are meant to enhance and aid experimental data collection, while the methods used for data assimilation are meant to enhance and aid analysis and prediction. While the methods used for both applications can be generalized, they are still separate applications.

From a purely spatial point of view, one way to interpolate between stations is known as kriging. This method has been used by both Asa [53] and Friedland et al. [54] to spatially interpolate wind data. The name "kriging" is in fact somewhat unique to geospatial statistics applications, and is more generally known as a Gaussian process [55]. Osborne et al. [56] used Gaussian processes to interpolate sensor readings. Hart et al. [57] used sensor stations along with satellite imagery to spatially interpolate evapotranspiration data. Finally, both Apaydin et al. [58] and Luo et al. [59] have written spatial interpolation comparison papers, where they compare methods such as kriging, inverse distance weighting, polynomial interpolation, splines, and more.

To reiterate, we use artificial neural network and multiple linear regression models to predict data at a given micrometeorology station given the data at other, nearby stations for a given time. Both of these techniques have been widely used in the past for various applications, but our work is novel in showing that both methods work at this spatial scale.

3. Methods

3.1. Experiment Overview

The data used for this publication were gathered during the Katabatic winds and Stability over Cadarache for Dispersion of Effluents (KASCADE) experiment of 2017. KASCADE 2017 is a follow-on experiment to the KASCADE experiment conducted in 2013 [3,4] that was focused on understanding the vertical structure of the atmosphere in the Cadarache Valley at night during stable atmospheric conditions. A brief description of the KASCADE 2017 is given here, while more details may be found in Dupuy et al. [60].

KASCADE 2017 was conducted in the Cadarache Valley of the Bouches-du-Rhône department in southeastern France from December 2016 through June 2017 (See Figure 1). The Cadarache Valley contains the French Alternative Energies and Atomic Energy Commission (CEA) research center, and the International Thermonuclear Experimental Reactor (ITER) is located in the adjacent Durance Valley. The CEA performs various types of nuclear research,

including the study of contaminant dispersion in the event of an accident. To better understand and predict dispersion events, it is critical to have detailed knowledge of small-scale winds and other atmospheric variables. Therefore, increasing our understanding of these phenomena was the main objective of the experiment.

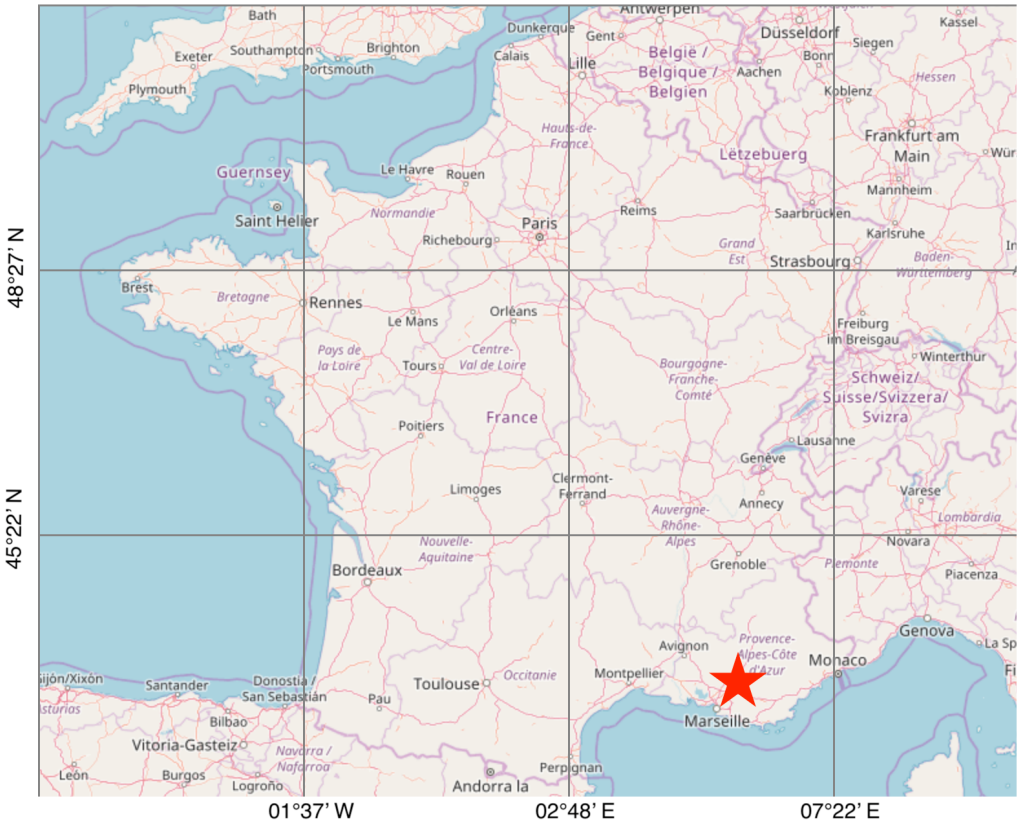

Figure 1. Map of France highlighting the Cadarache Valley experimental site (red star). © OpenStreetMap contributors [61].

As illustrated in Figure 2, the Cadarache Valley is a small 6 km long by 1 km wide valley. The elevation difference between the floor of the valley and the peaks is about 100 m. The mouth of the valley is connected to the Durance Valley, which runs approximately perpendicular to the Cadarache Valley [62]. The land cover and land use within the valley are heterogeneous, with a combination of buildings, roads, grassy areas, and light forests. A land-use map is presented in Figure 2.

During stably stratified periods, there are two main flow regimes present in the Cadarache Valley: thermally driven and synoptically forced. During thermally driven flow events, the winds blow down the valley with downslope flow components feeding into it. During these periods, the winds are typically relatively light at 2 m. There are strong diurnal patterns and spatial variability in temperature, humidity, and wind velocity. During synoptically forced flow events, larger-scale weather systems drive the winds relatively uniformly across the valley. Synoptically forced flows are typically stronger than thermally driven flows. There are minimal diurnal patterns and spatial variability in temperature, humidity, and wind velocity. A number of different synoptic situations are typical of the area and are described in [60].

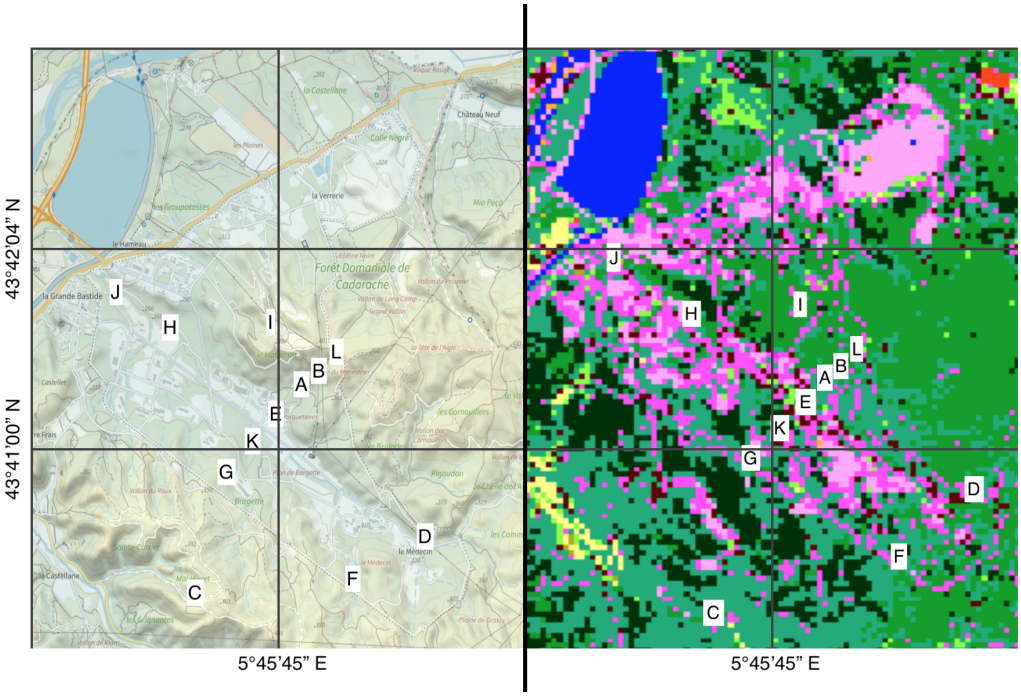

Figure 2. On the (**left**) is a map of the Cadarache Valley and on the (**right**) is the equivalent land-use map. Each contour level represents a 20 m change in elevation. On the land-use map, green shades indicate vegetation, pink shades indicate buildings and roads, and blue indicates water. The letters show the LEMS stations, which are described in Section 3. The map on the left is modified, and the original is from https://www.geoportail.gouv.fr/carte, accessed on 9 October 2021. The land-use map on the right is modified, and the original is from https://theia.cnes.fr, accessed on 9 October 2021.

For KASCADE 2017, the Cadarache Valley and the surrounding areas were heavily instrumented. Included in the deployment were: four sonic anemometer stations, one surface flux station, two SODAR stations, wind and temperature measurements from a 110 m tower, two general meteorological stations, and 12 Local energy-budget measurement stations (LEMS) (described below). In addition to these continuous observations, radiosondes were released every three hours during fourteen intensive observation periods (IOPs). In this paper, we use a subset of all the data collected. Namely, we use data collected by the LEMS from January through March 2017. Note that data from the KASCADE 2017 field campaign are available at https://kascade.sedoo.fr, accessed on 26 October 2021.

LEMS are small, low-cost meteorological stations that are capable of taking surface and subsurface measurements. The LEMS used for this experiment are the second generation of the instrument. The first generation was designed, built, and characterized in 2013 [40]. The second generation of the LEMS has a better radiation shield (the Socrima Multiplate radiation shield outlined in van der Meulen and Brandsma [63]), a better processor, and updated sensors. The LEMS are open source, and information and build instructions can be found at https://github.com/madvoid/LEMSv2, accessed on 26 October 2021.

Each LEMS deployed during KASCADE 2017 measures the following variables at approximately 2 m above ground: wind speed and direction, incoming shortwave radiation, air temperature, and air relative humidity. Barometric pressure is measured at approximately 1 m. In addition, LEMSs measure surface radiative temperature, as well as soil moisture content and temperature at two different heights (≈5 and ≈25 cm) below the surface. The heights of the sensors relative to the ground for each LEMS are approximately the same, and each LEMS has the same kind of sensor for each measurement. The wind speed and direction measurements for each LEMS were made using a cup and vane anemometer. Therefore, the data can be inaccurate at low-wind speeds, and may also demonstrate overspeed problems, as observed in the literature [64,65]. The LEMS were

deployed at 12 different locations around the Cadarache Valley. The LEMS locations are shown in Figure 2, and information about each LEMS location is presented in Table 1.

Table 1. Table of LEMS locations.

Name	Latitude	Longitude	Elevation (m)
LEMS A	43.68483	5.76803	332
LEMS B	43.68568	5.76885	347
LEMS C	43.66839	5.76142	397
LEMS D	43.67518	5.78671	328
LEMS E	43.68263	5.76568	293
LEMS F	43.66871	5.77791	383
LEMS G	43.67848	5.75763	325
LEMS H	43.69141	5.74918	276
LEMS I	43.69300	5.76253	385
LEMS J	43.69548	5.74323	262
LEMS K	43.68038	5.76003	317
LEMS L	43.68879	5.77071	368

Each LEMS station gathered data with a sampling frequency of 0.1 Hz. The data were quality controlled and averaged, with an averaging period of five minutes. These 5 min averages were used for all methods described in this paper.

As an aside, the Cadarache Valley has been the subject of several studies designed to better forecast local-scale winds. For example, Duine et al. [3] developed a simple method, based on potential temperature differences routinely observed from a 110 m tower, to nowcast the existence of down-valley winds. More recently, Dupuy et al. [60] used an ANN to downscale weather research and forecasting (WRF, [66]) model forecasts. Instead of using observational data as neural network inputs, Dupuy et al. [60] used low-resolution WRF output as inputs to an ANN. Their work effectively demonstrates that ANNs can be used to downscale physics-based weather models. In separate work, Dupuy et al. [62] used an ANN to nowcast local-scale 2 m wind speeds and directions at a point in the Cadarache Valley using the temperature gradient and velocity component data from an operational nearby 110 m tower. Their work is similar to the work presented here, but they only use one station and one variable. By using a valley-scale temperature gradient and ridge-top wind information, they successfully downscale local valley winds to a point.

3.2. ANN Details

The ANNs used in this paper were implemented using MATLAB's Neural Network Toolbox [28]. The number of hidden layers and nodes in the network, as well as the number of inputs and outputs, are presented in the results sections. The initial values for the weights and biases of the neural networks were randomly generated, and were dependent on the random seed. Since the random seed was varied across some experiments, they are presented alongside the results. While many ANNs use stochastic gradient descent (SGD) as their training algorithm [27], we use the Levenberg–Marquardt algorithm. The Levenberg–Marquardt algorithm is used for training since it is recommended by MATLAB as being the fastest converging training algorithm for a moderate amount of weights [67]. The Levenberg–Marquardt algorithm also produces the lowest mean squared error for many types of problems when compared to alternative algorithms [68]. The transfer function for the hidden layer is the hyperbolic tangent sigmoid function, while the transfer function for the output layer is the linear transfer function. The ANN performance function is the mean squared error (MSE), and normalization and regularization happens internally in MATLAB. MATLAB can preprocess data within the Neural Network Toolbox. For this application, MATLAB's preprocessing consisted of removing constant inputs, and mapping the minimum and maximum values of all inputs to −1 and +1, respectively. The inverse of these preprocessing steps were taken for the output of the network. While we created our own training and testing data for the experiments, it is important to note that MATLAB uses

the dataset to create its own internal training, testing, and validation data partitions. For all the experiments conducted, we set MATLAB to split the given training data into 75% internal training data, 20% internal validation data, and 5% internal testing data. The 5% internal testing data are only used by Matlab. The test sets we describe in Section 4 are completely held out and separate from Matlab's internal testing data. Alternate methods of determining model performance (such as cross-validation) could have been used, but we chose the held-out test sets for specific reasons, which will be explained further in Section 4. The training and testing splits we created are presented alongside the results for each experiment. Since the inputs and outputs for each experiment are different, they are also presented alongside the results for each experiment. Finally, ensemble averaging was frequently implemented. When ensemble averaging is utilized, multiple models with different initial weights are trained, and their outputs averaged [69]. Ensemble models typically have better performance than single models, and are less likely to show outlier performance [70]. In all the tests presented here, the neural networks that were ensemble-averaged varied by their weight initialization. We did not use ensemble-averaging with different inputs, outputs, or number of hidden nodes. If ensemble-averaging is used, it is specified alongside the results. All code can be accessed at https://doi.org/10.5281/zenodo.5921140, accessed on 31 January 2022.

3.3. MLR Details

As with the ANNs, the MLR models were implemented using MATLAB. In particular, the Statistics and Machine Learning Toolbox [71] was used. Since the inputs, outputs, and training/testing splits are experiment dependent, they are presented alongside the results. No extra preprocessing was done for any of the MLR models and none of the explanatory variables were transformed. That is, all MLR models are of the form $y = \beta_0 1 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p$. While ANNs have many hyperparameters, such as batch size, learning rate, loss functions, weight initialization, etc., MLR models do not. Therefore, the MLR models can be run without specifying many hyperparameters beforehand. All code can be accessed at https://doi.org/10.5281/zenodo.5921140, accessed on 31 January 2022.

4. Experiment and Results

To test the effectiveness of the ANN and MLR models in nowcasting microclimate parameters, a specific suite of tests were conducted. In these tests, virtual potential temperature, specific humidity, or individual wind speed components were predicted for LEMS A, B, D, E, F, G, H, or L using the measurements from LEMS I, J, and K. The measurements from LEMS I, J, and K used were the U and V wind components (where positive U points east and positive V points north), surface temperature, barometric pressure, and virtual potential temperature. When predicting specific humidity, the virtual potential temperature input was replaced by the specific humidity input because it worked better. These environmental parameters were chosen as outputs because they are typically measured by or can be derived from standard measurements made on other weather stations and are critical for dispersion modeling. LEMS I, J, and K were used as the input stations because they captured the different kinds of flows present in the Cadarache Valley: slope flows (LEMS K), valley flows (LEMS J), and ridge flows (LEMS I). (This was determined from a pre-analysis of the data [72].) Since these LEMSs represented the different flows present in the valley, we hypothesized that they would be the strongest predictors. These LEMSs were also present at different elevations so they would capture vertical stratification of the atmosphere, which is important for wind prediction, as shown in Duine et al. [3], Dupuy et al. [60]. LEMS C was not used in any of the tests because a complete dataset was not available. However, LEMS C was used for some of the analyses in the appendix. For example, we tested other sets of stations as inputs, and the analysis can be seen in Appendix C.

There are two "held-out" test periods for each variable (these are not Matlab's internally used test periods that were described in Section 3.2). None of the statistical models were trained on these two test periods. The test period starting 15 January 2017 at 00:00

standard local time and ending 20 January 2017 00:00 was characterized by thermally driven flows. The second test period, starting 27 January 2017 00:00 standard local time and ending 1 February 2017 00:00, was synoptically-forced. Since synoptically-forced flows break up thermally driven flows, these two test periods represent two extremes present in the Cadarache Valley, and test the range of the statistical models. The training data are identical for all runs: 5-min averages of the data from 16 December 2016 to 15 March 2017, excluding the two test periods. This training period was chosen because it is the period where there was a full deployment of sensors. Five-minute averages were chosen for the input variables to smooth turbulent fluctuations. We also tested other averaging periods (10 min, 15 min, 30 min, and 1 h), but the results are not displayed here. The models worked similarly well for those averaging periods.

An ANN ensemble average consisted of five randomly initialized neural networks. Each ANN within the ensemble-averaged model had 15 inputs (five parameters each from three LEMS), a single hidden layer, and one output. Each ANN had fourteen hidden nodes in the hidden layer, and each ensemble average was initialized using the same random seed. ANNs have several hyperparameters that can be changed to affect the model performance. While the details related to the testing of these hyperparameters can be seen in Appendix A, the main conclusion that needs to be presented here is that the ANNs do not exhibit a large change in performance when varying the number of hidden nodes.

To summarize, a five-network ANN ensemble was trained for each of the four output variables, for each of the eight target LEMSs, resulting in a total of 32 ANN ensemble averages. In addition, an MLR model was trained for each of the eight target LEMS for each of the four output variables, resulting in a total of 32 MLR models. The training data are identical for all runs within a given test period: 5 min averages of the data from 16 December 2016 to 15 March 2017, excluding the test data. The test data consisted of the periods from 15 January 2017 to 20 January 2017 and 27 January 2017 to 1 February 2017. None of the models were trained on the test data.

Since it would be difficult to view the results for 64 individual tests, summary plots were made for each of the two test periods. The results for the 15 January 2017–20 January 2017 period can be seen in Figure 3, and the results for the 27 January 2017–1 February 2017 can be seen in Figure 4. Each figure shows a box plot for each model for each variable tested. Each box represents the eight target LEMS that were predicted by input LEMS I, J, K. The abscissa shows the environmental variable predicted while the ordinate shows the normalized root-mean-squared error (NRMSE) between the model and the experimental data. The NRMSE is defined as:

$$\mathrm{NRMSE} = \frac{\sqrt{\frac{\sum_{n=1}^{N}(\hat{y}_n - y_n)^2}{N}}}{\max(y) - \min(y)}, \tag{1}$$

where $\hat{y}$ is the model prediction and y is the experimental data.

For visualization purposes, time series plots for subsets of the two periods can be seen in Figures 5 and 6. These figures also highlight the range of possibilities in the Cadarache Valley. For example, the virtual potential temperature plot of the synoptic period (Figure 6), shows the three stations having very similar virtual potential temperatures for the entire time period. However, the same plot for the thermally-driven period (Figure 5) shows the three stations having an approximately 7 °C virtual potential temperature difference at points in the time period, which is quite large, especially for such a small domain.

A visualization of the flows for a single time within both periods can be seen in Figure 7. The information presented in Figure 7 is also in Figures 5 and 6. Furthermore, scatter plots for the 15 January 2017–20 January 2017 period for both models when predicting virtual potential temperatures are shown in Appendix B.

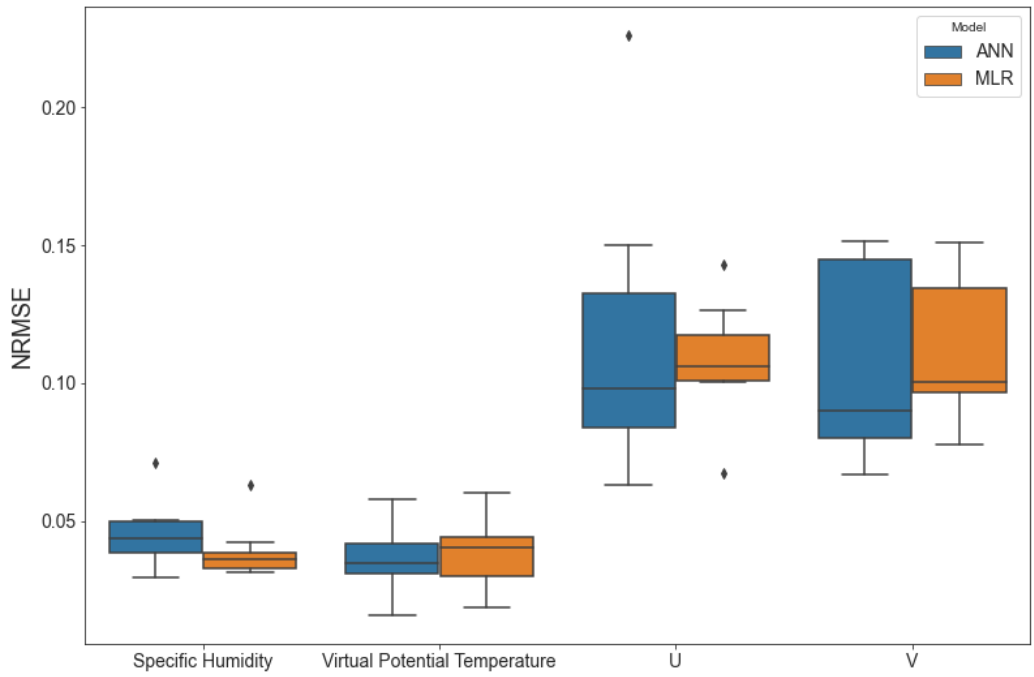

Figure 3. Summary of the performance of the 64 tests conducted for the 15 January 2017–20 January 2017 thermally-driven flow period. The abscissa represents the environmental variable being predicted, while the ordinate shows the normalized root-mean-squared error between the model and the experimental data. Any result outside of 1.5 times the interquartile range was deemed an outlier and is represented by a dot. The data represented by each "box" are the eight LEMS that were predicted by input LEMS I, J, and K.

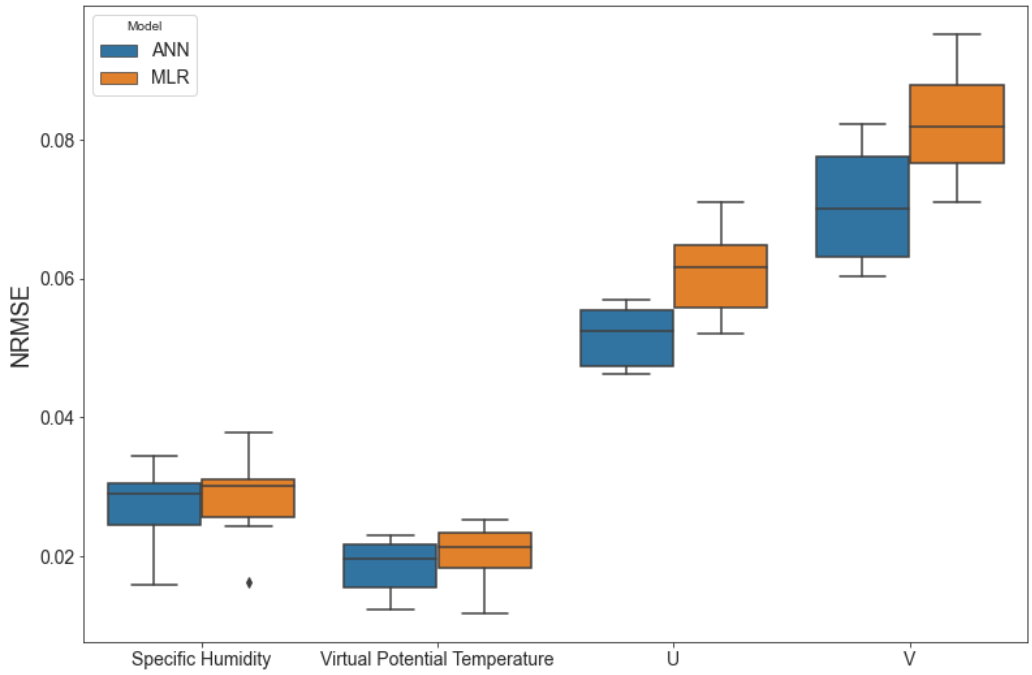

Figure 4. Same as Figure 3 but for the 27 January 2017–1 February 2017 period corresponding to synoptically-forced flows. Note that the ordinate scale is different from Figure 3.

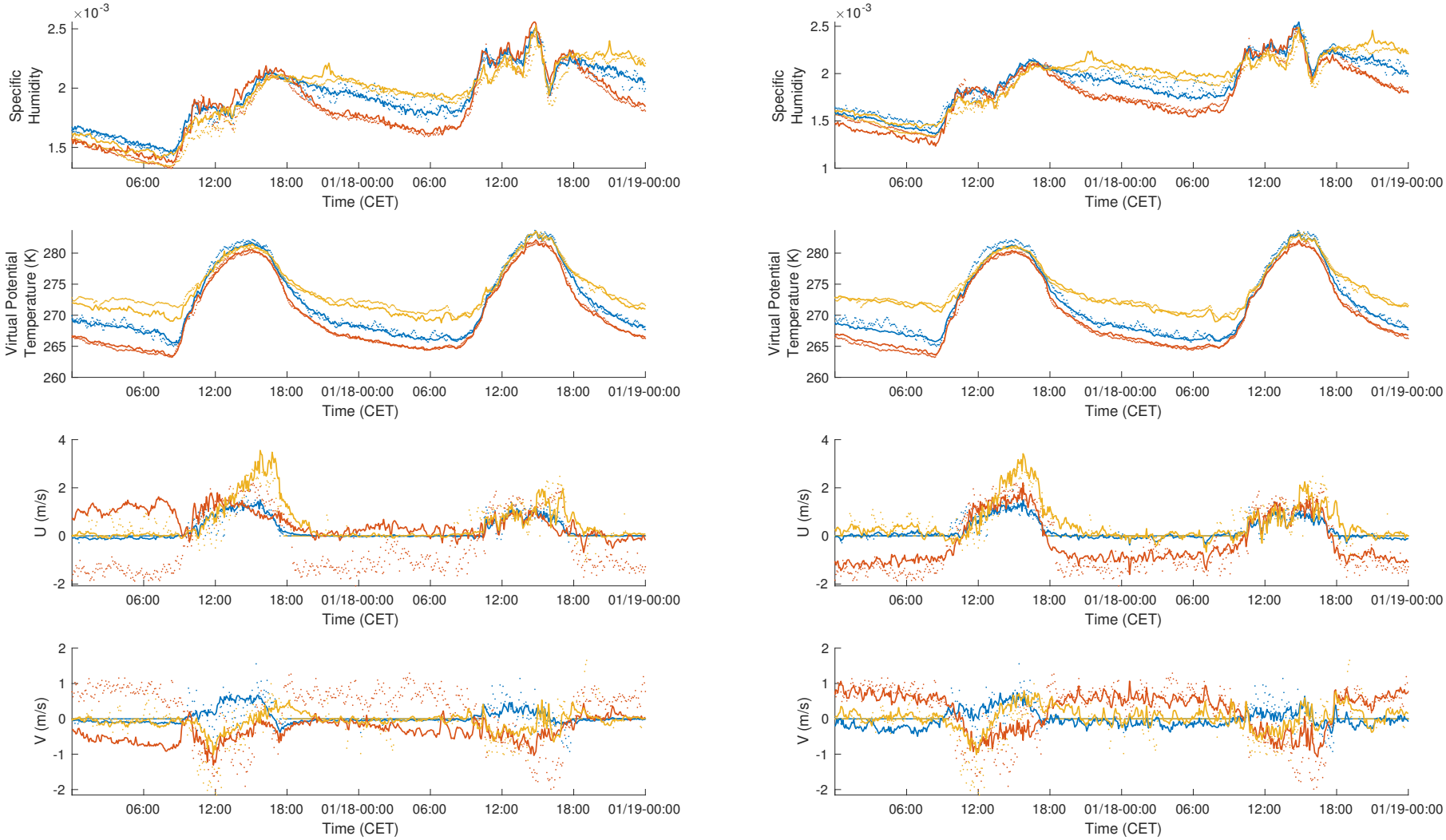

Figure 5. Time series for 17 January 2017 to 19 January 2017 in local standard time, representing a subset of the thermally-driven flow period. The (**left**) column of plots shows the ANN predictions compared to the data, and the (**right**) column of plots shows the MLR predictions compared to the data. The solid lines are predictions, while the dotted lines are measurements. Blue is LEMS A, red is LEMS E, and yellow is LEMS F.

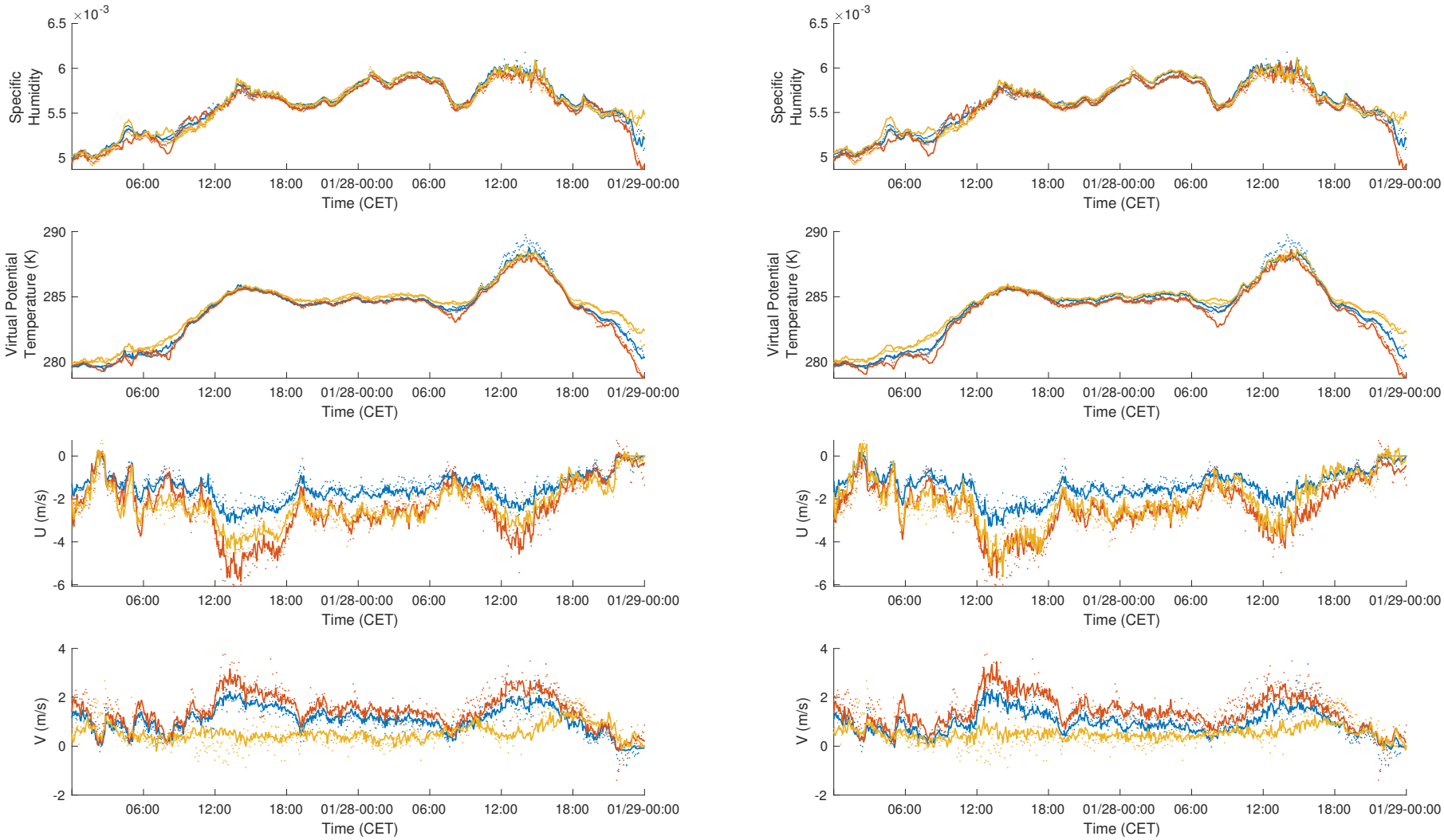

Figure 6. The same as Figure 5 but for 27 January 2017 to 29 January 2017 local standard time, representing a subset of the synoptically-forced flow period.

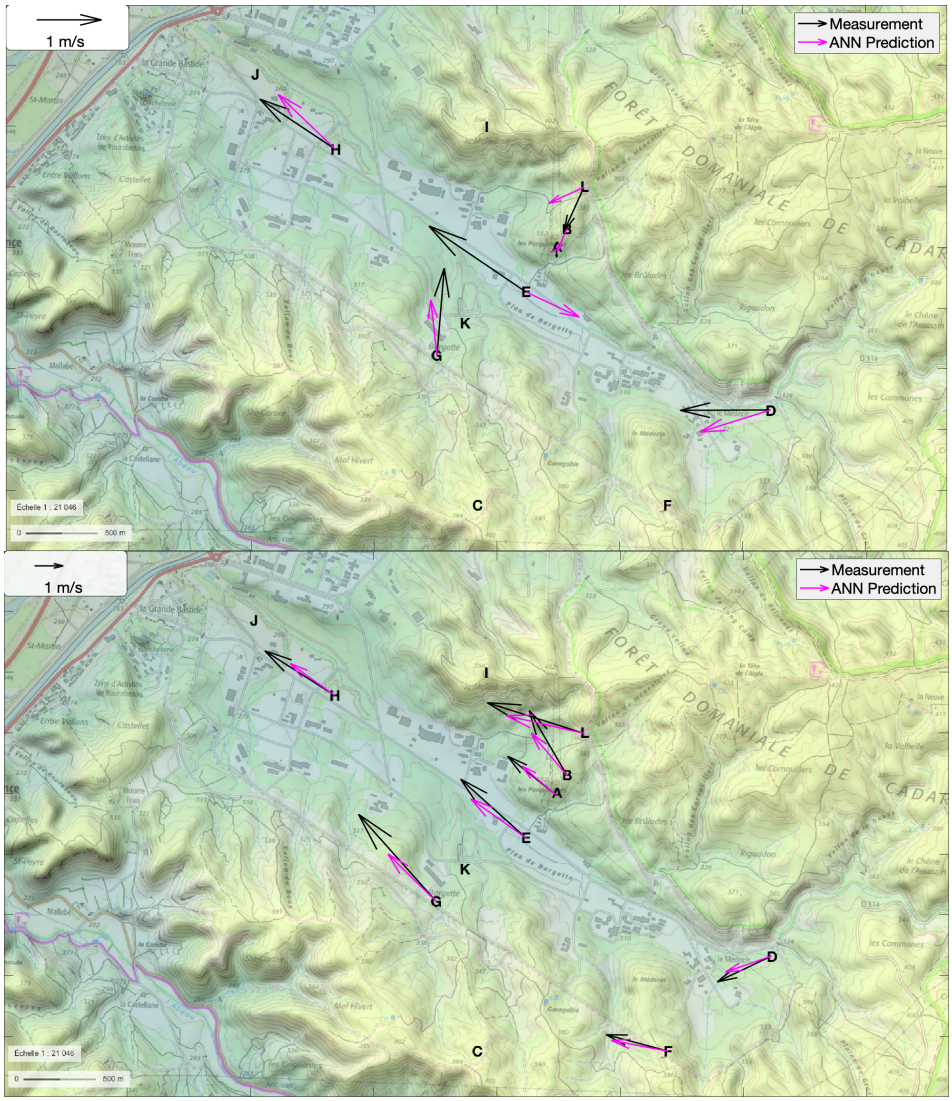

Figure 7. Measured and predicted flow visualization for a single time step in the respective period. The LEMS locations are marked by bold letters. LEMS I, J, and K do not show arrows because they were used for the training data, and LEMS C does not show arrows because it was excluded from this analysis. The (**top**) half of the figure displays an example of a typical thermally driven flow in the Cadarache Valley (17 January 2017 00:15:00 local standard time), and the (**bottom**) half of the figure displays an example of typical synoptically-forced flow in the Cadarache Valley (27 January 2017 00:15:00 local time). These snapshots can also be seen in the time series Figures 5 and 6. Map is modified, and the original is from https://www.geoportail.gouv.fr/carte, accessed on 9 October 2021. [institut national de l'information geógraphique et forestière (IGN)].

5. Discussion

Many interesting points of discussion resulted from this analysis. Figures 3 and 4 show that for both test periods, the mean NRMSE of the MLR predictions was close to the mean NRMSE of the ANN predictions. The virtual potential temperature also displayed the lowest prediction error, while the wind components had the highest prediction error. Interestingly, the prediction of the V component of the wind velocity vector had a higher inter-quartile range than the U component. It appears that for all variables, except for specific humidity during the thermally driven test period, the ANN has a better mean prediction than the MLR, though not by much. Finally, it is important to note that several of the prediction sets have outliers, which are defined as any value that is outside 1.5 × IQR, where IQR is the inter-quartile range.

To quantify the similarity between the ANN and MLR models, we statistically analyzed the data similar to Valavi et al. [73] and Shafizadeh-Moghadam et al. [74]. We used Welch's two-sample *t*-test to compare the mean NRMSE for all of the LEMSs between each of the environmental variables for each of the two test periods. When conducting the statistical test for a given test period and environmental variable, the null hypothesis was that the two methods would perform the same, and the alternate hypothesis was that the two methods would perform differently. As seen in Table 2, the only *p*-values below the 5% significance level were for U and V for the 27 January 2017–1 February 2017 test period, and only U was below the 1% significance level. This means that the null hypothesis could not be rejected for all other tests. This evidence indicates that the two methods likely have the same performance, with the exception of predicting U and V during synoptically-forced flow periods, where the ANN probably (but not definitely) performs better.

Table 2. Table of *p*-values from conducting Welch's two-sample *t*-test to compare the MLR and ANN models.

Variable	15 January 2017–20 January 2017	27 January 2017–1 February 2017
Specific Humidity	0.29	0.68
Virtual Potential Temperature	0.82	0.47
U	0.71	6.6×10^{-3}
V	0.75	2.0×10^{-2}

A difference in performance between the thermally driven flow days (15 January 2017 to 20 January 2017) and the synoptically-forced flow days (27 January 2017 to 1 February 2017) for the wind-velocity components was apparent. For example, the mean NRMSE for the U component prediction during the thermally driven period was between 0.10 and 0.12, while the same statistic for the synoptically-forced period was between 0.05 and 0.06. As expected, on thermally driven flow days, the wind velocity between stations was much less correlated and behaved more independently. In contrast, on synoptically-forced flow days, the wind velocity between stations was much more correlated, and the readings were very similar. Hence, the models predict the wind velocity much better on synoptically-forced flow days.

It is also important to discuss the physical reasons why wind-velocity components are more difficult to predict than virtual potential temperature or specific humidity. First, both virtual potential temperature and specific humidity are strongly correlated with the diurnal cycle. The wind velocity components in the Cadarache Valley also have a diurnal cycle, but the cycle is not powerful enough to overcome all the weak fluctuations and perturbations that might be present. However, wind velocity is also difficult to predict for reasons other than the underlying environmental physics. The LEMS measures wind velocity with a cup and vane anemometer, which does not measure wind speeds or direction accurately, if at all, at wind speeds less than 0.5 ms^{-1}. Therefore, many of the low-magnitude readings are likely incorrect. During these low wind speed periods, the models gave non-zero values, while the data indicated zero speed, increasing the calculated error. In addition, the anemometer "transfer function" had a discontinuity. The wind speed measurement was zero and then quickly jumped to 0.5 ms^{-1} or above. These discontinuities can be difficult to model accurately. To overcome these difficulties, it would be better to use a sonic anemometer that measures low wind speeds more accurately.

When using these methods, the input LEMS must be chosen. We chose LEMS I, J, and K because they were representative of the three types of flow in the Cadarache Valley. However, we could have chosen any three LEMS as the inputs. The question that arises is: does the choice of input LEMS affect the general prediction power of these methods? To answer this, we conducted a test where all the possible combinations of three LEMS were used to predict the values at the other LEMS. This ended up being $\frac{12!}{3!(12-3)!} = 220$ combinations total. The details of this analysis can be seen in Appendix C. In short, the

choice of LEMS does not matter for the types of predictions discussed here, i.e., any three LEMS were as predictive as the next three.

We note that we originally did not intend to test MLR models on this dataset because we assumed that the relationship between the measurements of the different LEMSs was highly non-linear. However, when conducting the ANN hidden node analysis (see Appendix A), we noticed that a single-node ANN performed nearly as well as a multi-node ANN. Since a single-node ANN is essentially a multiple linear regression whose output is passed through a sigmoidal function, we decided to try an MLR as the nowcasting method. Surprisingly, it worked well. While the governing equations of the flow and transport processes being measured are non-linear, those non-linear effects seem to be secondary to the overall forcing mechanisms (e.g., radiation processes, large-scale wind) that govern the flow that the MLR is able to capture.

Since MLR models work on this dataset, there are two main reasons to use an MLR instead of an ANN, if possible. The first is computational energy/runtime. Even though the energy costs of training and running ANNs have dropped significantly in the last several years, they are still more computationally intensive to operate than MLR models. The exact timing varies, but on our laptop-class computers, the ANNs train on the order of 10–100 s, while the MLR models train in less than a second. The training time is not necessarily important for gap filling, but can be. If gap filling is to be done in situ on low-powered hardware, then training time and computational complexity become more important. The second is interpretability. ANNs tend to be black boxes that are difficult to interpret, whereas MLR models and their associated coefficients are transparent and easier to interpret. Since the performance of the two models is so similar, and because there is not much room for improvement, we think the energy/time requirements of the MLR make it a better model to use.

However, the biggest concern associated with using MLR models is the assumptions that need to be met. According to Poole and O'Farrell [75], there are six critical assumptions made when using an MLR successfully. One of these assumptions is that the independent variables are linearly independent of each other. When they are not, collinearity exists, and the precision of the regression coefficient decreases [76]. During our analysis, we noticed that there were high variance inflation factors (VIFs) for many of the coefficients, implying multicollinearity. This makes intuitive sense. For example, when the sun sets, all stations will measure a radiation drop, and the air and surface temperature measurements will correlate. It is important to note that linear regression models can predict perfectly well with high VIFs, but the regression coefficients have high variance, and statistically interpreting the models must be done with care [77,78]. To successfully interpret the models, one must reduce the number of explanatory variables until multicollinearity is minimized. This can also be done for the ANN, but due to the ANN's inherent "black-box" nature, one should be careful interpreting it regardless. Manually, multicollinearity can be minimized by calculating correlation coefficients between all explanatory variables and removing highly correlated ones. Removing correlated variables can also be done by running algorithms, such as lasso or ridge regression [27], which automatically removes unneeded variables. Using lasso regression will also quickly reveal which stations are most correlated with others. While some preliminary work has been done with regard to this, it was not included in this paper as we felt it was out of the scope. However, we have included a table of input variable correlation coefficients along with the preliminary discussion in Appendix D for reader reference.

Despite ANNs being more complicated to train than MLRs, a standard feedforward ANN is easier to train than other models, such as deep-learning models [79]. Many deep-learning models require desktop-class computers with powerful GPUs to train, and frankly, are probably overkill for this problem given the performance of both the ANNs and MLR models. For this paper, we wanted to test simple models that can be run on limited computing resources, possibly even on the sensor stations themselves. This is also one reason we did not test state-estimation methods. State-estimation methods

are also computationally expensive, and many of them require an underlying physical model [52]. At the spatial scales present in this problem, they would require using Large-Eddy Simulations (LES) [80].

6. Summary

The main purpose of this paper is to demonstrate that artificial neural networks and multiple linear regression models can be used to nowcast environmental measurements in complex-terrain boundary-layer meteorological applications. Specifically, we show that these methods work in the Cadarache Valley located in southeastern France. The valley, which is approximately 6 km long by 1 km wide, was instrumented with a dozen low-cost weather stations (called LEMS) for a four-month-long field experiment. The weather stations measured several different variables, but the variables predicted by the models were wind-velocity components, virtual potential temperature, specific humidity, and air temperature. The valley exhibited various types of flows, including nearly pure thermally-driven flows and a wide range of synoptically-forced flows.

In general, both the ANN and MLR models performed similarly well on the test data. Two test periods were used. The test period from 15 January 2017 to 20 January 2017 represents mostly thermally driven slope- and valley-flow, while the test period from 27 January 2017 to 1 February 2017 represents mostly synoptically-forced flow. Both models performed better on synoptically-forced flow periods over thermally-driven flow periods because variability amongst the stations was smaller during synoptically-forced periods. Both models predicted specific humidity and virtual potential temperature better than the wind components, likely because the wind components are much more sensitive to small perturbations that are not overcome by the diurnal cycle. These promising results indicate that ANN models, and at times MLR models can be used for data filling after a field experiment has been completed. This provides substantial spatial information in regions of complex terrain where thermodynamic and dynamic variables are highly variable in space.

Despite these tests, there are still many tests that can be run, and future work to be done. The purpose of this paper was to present promising results from some preliminary tests of these methods, not to be an exhaustive reference on the methods' capabilities. In addition, some of these tests have virtually limitless configurations that can be run, meaning we had to limit the scope of the tests. For example, we ran the hidden node tests (presented in Appendix A) using a single input environmental variable (for 11 LEMS). We could have run it using two environmental variables for 11 LEMSs, or three environmental variables. etc. Mainly, we chose tests that we felt exposed the behavior of each model well without trying to infinitely optimize parameters and hyperparameters. Especially for the ANN, we felt that many of the possible tests we could have done were out of the scope of this particular paper since we showed that (a) MLR models work, and (b) the number of hidden nodes does not affect output performance very much.

Along these lines, we could exhaustively test a completely different model such as random forest regression. Random forest regression is a powerful method that has been used for atmospheric science applications in the past [81]. We did some preliminary testing with a random forest regression model, and it worked almost as well as the MLR and ANN models (See Appendix E for preliminary results). With hyperparameter tuning, we believe random forest regression can work similarly well. We decided not to pursue the random forest regression model in-depth for this paper because we believed that the MLR and ANN models showed sufficient performance.

Hence, there are many topics that we would consider as possibilities for future work. For example, for both ANN and MLR models, the environmental variables or stations could be added one by one, in different orders to see how results are affected, similar to Dupuy et al. [62]. The amount of training/testing data could be changed to determine the minimum amount of data required to create a successful nowcasting tool. Data from different locations with different spatial/temporal scales or weather events (e.g., precipitation events) could be used. One could test scientific hypotheses about the flow using

the MLR models. Testing scientific hypotheses this way would also reveal the importance of the variables used. Finally, one could test time-series-specific regression and neural network-based models, such as ARIMA or LSTM.

With the completion of these and other tasks, we conclude that both artificial neural networks and multiple linear regression show promise in becoming successful nowcasting tools in micrometeorology.

Author Contributions: Conceptualization, E.P., P.D. and T.H.; methodology, N.G. and E.P.; software, N.G.; validation, N.G. and E.P.; formal analysis, N.G.; investigation, N.G. and F.D.; resources, E.P., P.D. and T.H.; data curation, F.D. and N.G.; writing—original draft preparation, N.G.; writing—review and editing, all; visualization, N.G.; supervision, E.P., P.D. and T.H.; project administration, E.P., P.D. and T.H.; funding acquisition, E.P., P.D. and T.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Franco-American Fulbright Commission, the University of Utah Global Change and Sustainability Center, MRISQ project from the French Alternative Energies and Atomic Energy Commission (CEA), and l'Observatoire Midi-Pyrénées.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data and code can be found at https://doi.org/10.5281/zenodo.5921140, accessed on 26 October 2021.

Acknowledgments: We would like to thank the 4M team of the Centre National de Recherche Météorologique for kindly allowing us to use their Socrima radiation shields during KASCADE 2017, Eric Pique (from Laboratoire d'Aérologie) for the preparation and installation of the LEMS for KASCADE 2017, and Pierre Rubin (from CEA) for general experimental help. Prepared by LLNL under Contract DE-AC52-07NA27344 with additional help from the postdoctoral researcher Career Development Time. Map data copyrighted OpenStreetMap contributors and available from https://www.openstreetmap.org, accessed on 26 October 2021.

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.

Abbreviations

The following abbreviations are used in this manuscript:

ANN	Artificial Neural Network
LEMS	Local Energy-Budget Measurement Stations
MLR	Multiple Linear Regression
NRMSE	Normalized Root-Mean-Squared Error

Appendix A. Artificial Neural Network Sensitivity Tests

All neural networks have a number of hyperparameters (parameters not part of the training process) that can be tuned to change model performance [79]. In this section, we investigate sensitivity of the model results to some of these hyperparameters. Since the MATLAB Neural Network Toolbox was used for implementation, many of the default parameters of the toolbox are used, and not expanded upon here. Some example hyperparameters that are not discussed are the hidden node activation function, the training algorithm, and the training batch size. Exploration of the variation of these hyperparameters is presented in various sources in the literature (e.g., [82]). We also do not explore the number of hidden layers used in the ANNs, as there is often no practical reason to have more than one hidden layer for standard feedforward neural networks [83]. Note that LEMS C is included in these results, even though it was excluded from the main text.

One of the more important hyperparameters that can be tuned is the number of hidden nodes in the hidden layer. To study this effect, we varied the number of nodes in the ANN

to see how the performance changed. Four tests were conducted where the hidden nodes were varied. In each of the four tests, a single environmental variable from LEMS A, B, C, D, F, G, H, I, J, K, and L was used to predict the same environmental variable at LEMS E. The four environmental variables used were specific humidity, virtual potential temperature, wind velocity U component, and wind velocity V component. Note that each test only uses one environmental variable as the input, whereas the tests described in Section 4 used the wind components, virtual potential temperature, surface temperature, and barometric pressure as inputs. This was done because we thought that a simpler test would better illustrate the effect of changing the hidden nodes. The testing data for each ANN were from 15 January 2017 to 20 January 2017. This testing period was used because it was studied extensively for this experiment elsewhere [60]. The training data were from 16 December 2016 to 15 March 2017, excluding the testing data period. The number of nodes was incremented by one from 1 node to 30 nodes. For each number of nodes, an ensemble average of five neural networks was used to calculate the performance. The output of this ensemble average was used to determine the performance of any given number of nodes. An ensemble average was used here to prevent any outlier ANNs from skewing the results. The NRMSE was used as the error metric to compare variables.

As shown in Figure A1, the ANN performance is relatively independent of the number of hidden nodes for all environmental variables predicted. Interestingly, increasing or decreasing the number of hidden nodes does not significantly change the performance of the ANN. Even more interestingly, for all variables except specific humidity, a one-hidden-node ANN performs as well as, or better than "more powerful" ANNs. Even for specific humidity, the one hidden node network has a low NRMSE, but not as low as the higher-node networks. In fact, this result is what motivated us to use linear regression to predict these environmental variables as a single node neural network is essentially the linear regression equation passed through an activation function. This also explains why we used 14 hidden nodes for the tests conducted in Section 4; we could have used most values between 2 and 30 and it would not have made a very large difference in the results. The fact that a multiple linear regression can perform a task essentially as well as an artificial neural network means that an ANN is unnecessarily powerful for this specific task, and that hyperparameter tuning will only yield marginal gains.

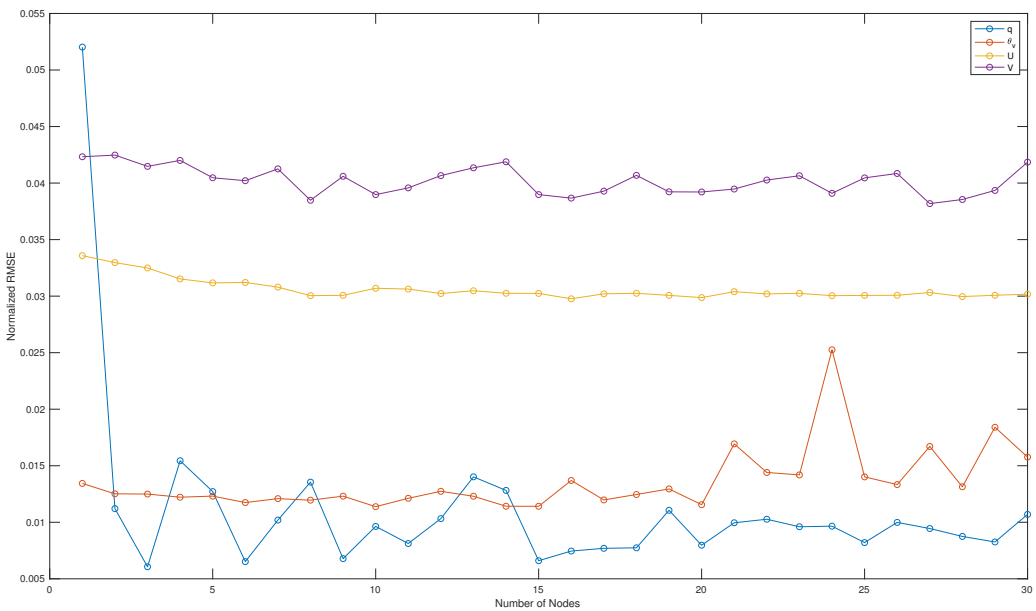

Figure A1. Illustration of the impact of incrementally adding hidden nodes to the base ANN described in Appendix A on model performance as quantified by normalized RMSE.

Another hyperparameter that is relevant to our application of ANNs is the number of output nodes. ANNs are not limited to one output; they can have several outputs for

any number of inputs. To produce the results of Section 4, eight different ANNs were trained with one output each. We repeated the tests from Section 4 identically, except we trained a single ANN with eight outputs. We found no discernible or systematic difference between the results of the two approaches. Hence, we have not presented any plots from the multiple-output test. However, we did notice that the time to train one multiple-output neural network was longer than training eight separate single-output networks.

On a more practical note, a hyperparameter that is important when using MATLAB's Neural Network Toolbox is the train/test/validate data split percentage. When training the ANN, MATLAB internally splits the data into a training subset, a testing subset, and a validation subset. By default, MATLAB uses 70% of the original data as training data, 15% as validation data, and 15% as test data. The test data are only there so the user can view the performance of the neural network. The ANN performance on validation data, however, is often used by MATLAB as a training stopping condition. Therefore, in this instance, having a small test split will not affect us very much, as we have our own separate test data that the ANN has never "seen" before. However, having more training data will increase neural network performance (and avoid overtraining), and having more validation data will improve generalization. This is why all ANNs presented in this paper have a train/validation/test split of 75%/20%/5%.

Appendix B. Paired in Space and Time ANN and MLR Evaluations

While Figures 3 and 4 show the summaries of 64 different tests, here, we present additional base results of some of these tests. The ANN and MLR scatter plots for the prediction of virtual potential temperature for the 15 January 2017 to 20 January 2017 period are shown in Figures A2 and A3.

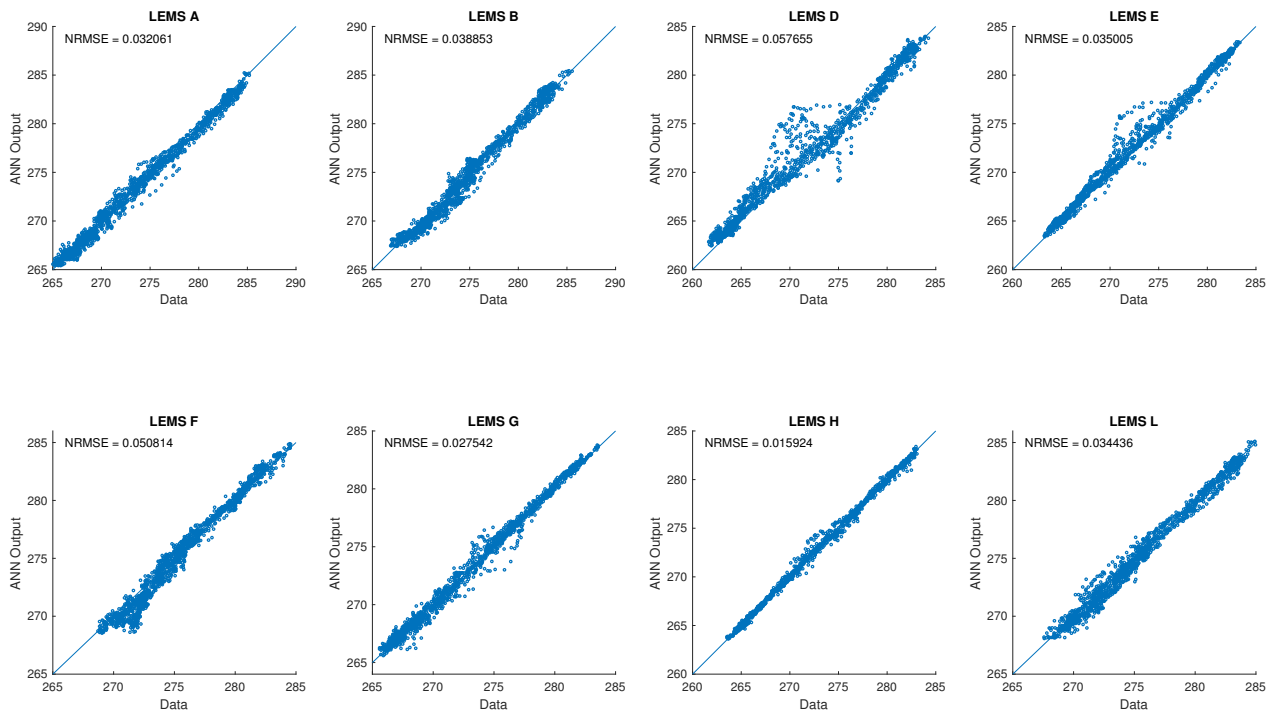

Figure A2. Scatter plots for the ANN prediction of virtual potential temperature for the 15 January 2017–20 January 2017 experimental period. LEMS I, J, and K are the inputs, and the units are in Kelvin.

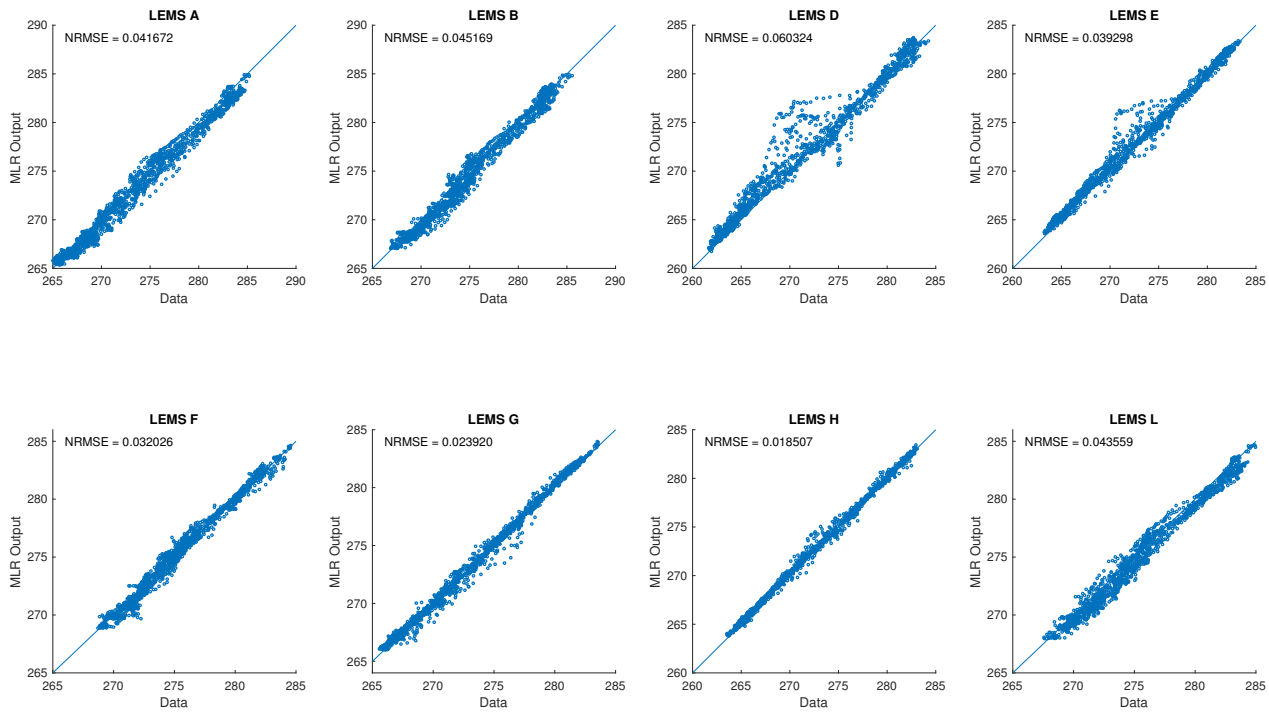

Figure A3. Scatter plots for the MLR prediction of virtual potential temperature for the 15 January 2017–20 January 2017 experimental period. LEMS I, J, and K are the inputs, and the units are in Kelvin.

Appendix C. Multiple Linear Regression Location Sensitivity

The last test we performed was an investigation of the MLR model's sensitivity to inputs. The results presented in Section 4 always used data from the same three LEMS as inputs: LEMS I, J, and K. These LEMS were chosen to be the inputs because their locations were spread across the measurement area (both horizontally and vertically) and captured many phenomena associated with thermal circulation in complex terrain (e.g., cold pools, slope/valley flows). A priori, one might assume that the good performance exhibited by the MLRs is due to the locations of the input LEMS, and not because of the inherent power of the prediction algorithm. To test this, a combinatorial analysis was performed, where every possible combination of three LEMSs was used to predict values at the other nine LEMSs. The inputs were identical to those from the tests in Section 4. Specifically, the inputs were: wind velocity components, surface temperature, barometric pressure, and virtual potential temperature from three LEMSs. The output was the virtual potential temperature of the other nine LEMSs. The prediction algorithm used was MLR. While LEMS C was excluded from the tests in Section 4, it was included here to truly explore the spatial relationship between the LEMSs.

The testing data were taken from 15 January 2017 to 20 January 2017. The training data were from 12 January 2017 to 15 March 2017, excluding the testing data period (LEMS C was available from 12 January on). There are $\frac{12!}{3!(12-3)!} = 220$ different combinations of input LEMSs. Throughout this document, when we refer to the "combination number", we mean a specific combination out of the 220 combinations. For example, combination "1" would have LEMS A, B, and C as input LEMS, and the rest as output LEMS. The metric we use for evaluating the performance was R^2, also known as the coefficient of determination. R^2 is defined as $R^2 \equiv 1 - (\sum_n^N (y_n - \hat{y}_n)^2) \cdot (\sum_n^N (y_n - \overline{y})^2)^{-1}$, and measures how well the model performs compared to the dataset mean. Since nine different MLR models were trained for each combination, there were nine R^2 values associated with each combination; each R^2 value was computed with the difference between the MLR model prediction and the experimental data.

Figure A4 shows the results of the combinatorial analysis for prediction of the virtual potential temperature. The figure shows the mean and range of R^2 values for each

combination. Most combinations have a mean R^2 value above 0.95, and most R^2 ranges were above 0.90. This shows that the choice of input LEMSs was not very important when making predictions. The combination with an unusually low R^2 range was combination 177, which had LEMS E, H, and J as input LEMSs. These LEMSs are highly correlated since they are aligned with the valley and therefore have a hard time predicting the other LEMSs. Regardless, the worst case prediction for this combination had an R^2 value of about 0.77, which is still good. The fact that the most combinations had high R^2 values also justifies the choice of LEMS I, J, K for the tests conducted in Section 4.

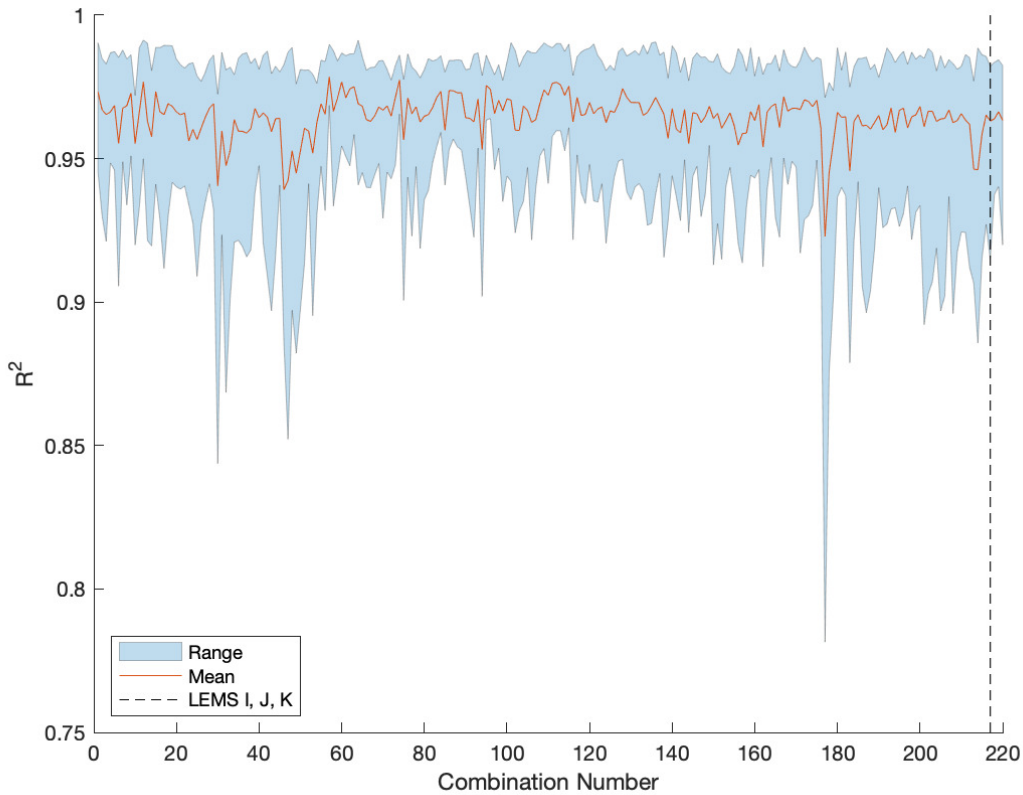

Figure A4. Example of combinatorial analysis for the virtual potential temperature predictions from MLR for 15 January 2017 through 20 January 2017. For every combination of three input and nine output LEMSs, the minimum, maximum, and mean R^2 value was calculated. The range of the R^2 values are shown in light blue and the mean R^2 value is shown in red. As is evident from the plot, all combinations of input LEMSs perform similarly. We use these statistics because they are a good way to summarize multiple time series.

Appendix D. Input Variable Correlations

Figure A5 shows the Pearson's R correlation coefficients between all input variables when predicting specific humidity. As expected, like variables between stations were highly correlated. However, correlations between unlike variables were very poorly correlated, even showing no correlations. If statistical inference were to be done on the results presented here, work would need to be done to remove highly correlated variables while maintaining predictive power. This would likely lower the variance inflation factors that were referenced in the main text.

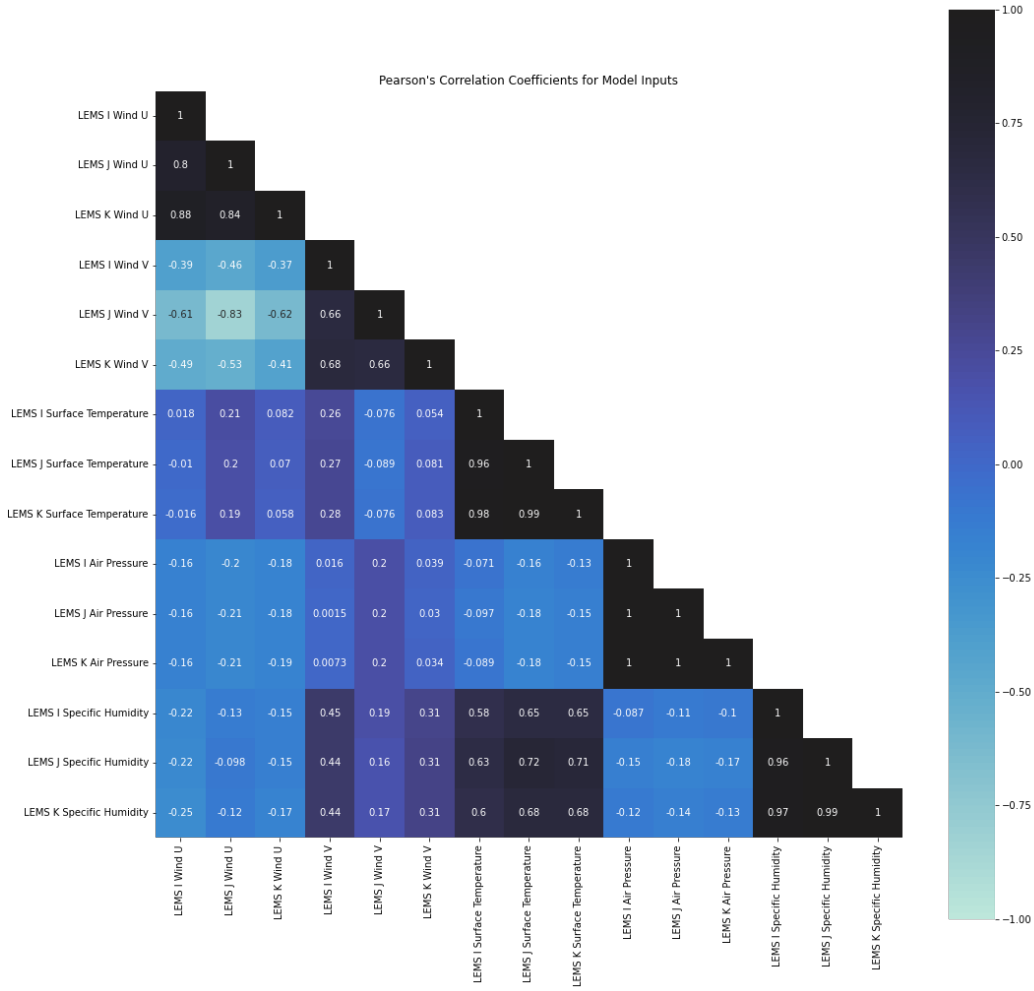

Figure A5. Pearson's R correlation coefficients between the input variables when predicting specific humidity. Further discussion can be found in Appendix D.

Appendix E. Random Forest Preliminary Results

In addition to MLR and ANN tests, we performed preliminary testing with a random forest regression (RFR) model. As stated in Section 6, we chose not to pursue this further because we believed that the other two models performed sufficiently. However, the results may be of interest to the reader so we have included them here.

We used Matlab's Statistics and Machine Learning Toolbox's implementation of boosted random trees with default settings. The toolbox fit 100 boosted regression trees to the data. We used identical parameters and training sizes as the MLR and ANN models. Figures A6 and A7 below show the results in the same format as Figures 3 and 4.

It is evident from Figures A6 and A7 that the RFR model performs about as well as the other two models at best, and significantly worse than the other two models at worst. However, we believe that hyperparameter tuning will increase the performance of the RFR model. We do not believe that it is able to perform significantly better than the other two models though, as the other two models already perform quite well.

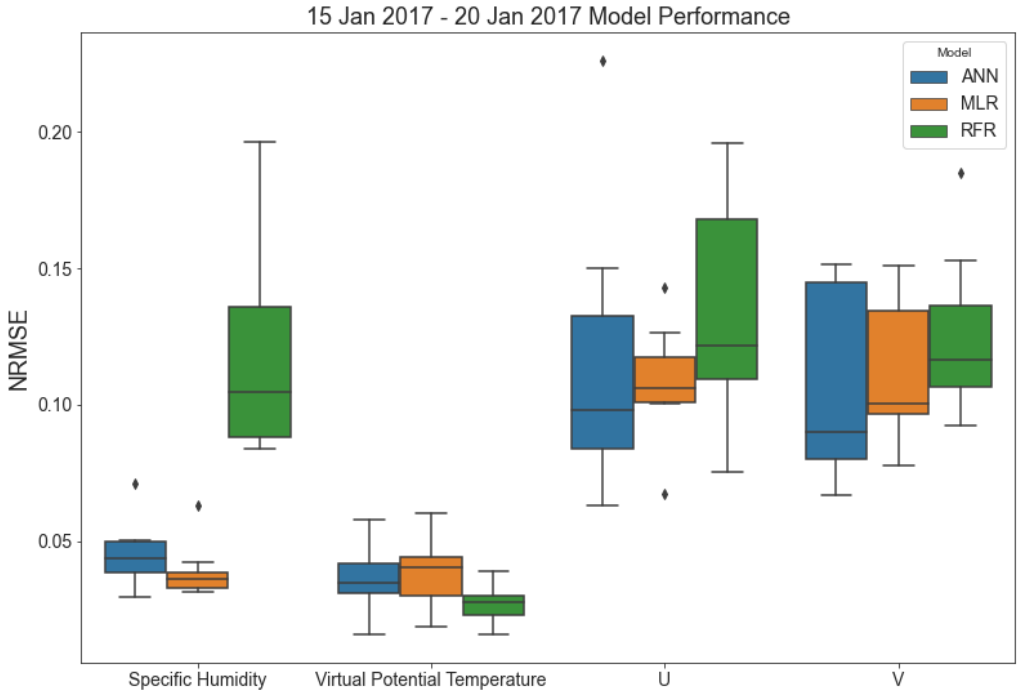

Figure A6. Same as Figure 3 but including the random forest model.

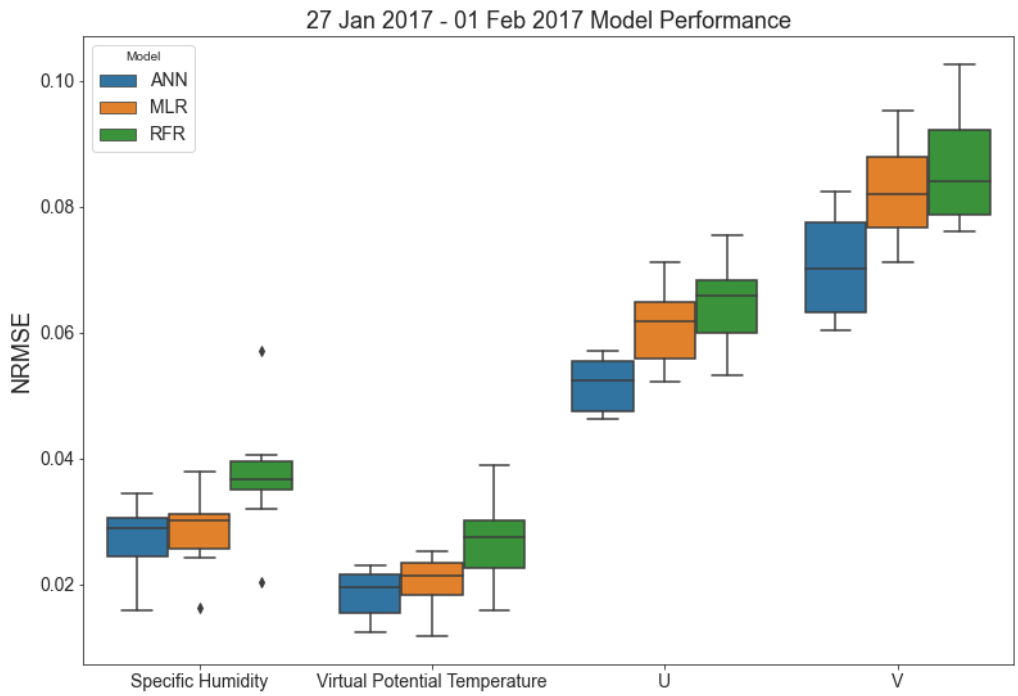

Figure A7. Same as Figure 4 but including the random forest model.

References

1. Emeis, S. *Measurement Methods in Atmospheric Sciences: In Situ and Remote*; Gebrüder Borntraeger Science Publishers: Stuttgart, Germany, 2010.
2. Miller, N.E.; Stoll, R.; Mahafee, W.; Neill, T.; Pardyjak, E.R. Field-scale particle transport in a trellised agricultural canopy during periods of row-aligned winds. In Proceedings of the 22nd Symposium Boundary Layers and Turbulence, Salt Lake City, UT, USA, 20–24 June 2016.
3. Duine, G.J.; Hedde, T.; Roubin, P.; Durand, P. A Simple Method Based on Routine Observations to Nowcast Down-Valley Flows in Shallow, Narrow Valleys. *J. Appl. Meteorol. Climatol.* **2016**, *55*, 1497–1511. [CrossRef]

4. Duine, G.J.; Hedde, T.; Roubin, P.; Durand, P.; Lothon, M.; Lohou, F.; Augustin, P.; Fourmentin, M. Characterization of valley flows within two confluent valleys under stable conditions: Observations from the KASCADE field experiment. *Q. J. R. Meteorol. Soc.* **2017**, *143*, 1886–1902. [CrossRef]
5. Philippopoulos, K.; Deligiorgi, D. Application of artificial neural networks for the spatial estimation of wind speed in a coastal region with complex topography. *Renew. Energy* **2012**, *38*, 75–82. [CrossRef]
6. Fernando, H.; Pardyjak, E.; Di Sabatino, S.; Chow, F.; De Wekker, S.; Hoch, S.; Hacker, J.; Pace, J.; Pratt, T.; Pu, Z.; et al. The MATERHORN: Unraveling the intricacies of mountain weather. *Bull. Am. Meteorol. Soc.* **2015**, *96*, 1945–1967. [CrossRef]
7. Lothon, M.; Lohou, F.; Pino, D.; Couvreux, F.; Pardyjak, E.R.; Reuder, J.; de Arellano, J.; Durand, P.; Hartogensis, O.; Legain, D.; et al. The BLLAST field experiment: Boundary-layer late afternoon and sunset turbulence. *Atmos. Chem. Phys.* **2014**, *14*, 10931–10960. [CrossRef]
8. Banta, R.; Olivier, L.; Gudiksen, P.; Lange, R. Implications of small-scale flow features to modeling dispersion over complex terrain. *J. Appl. Meteorol. Climatol.* **1996**, *35*, 330–342. [CrossRef]
9. Giovannini, L.; Ferrero, E.; Karl, T.; Rotach, M.W.; Staquet, C.; Trini Castelli, S.; Zardi, D. Atmospheric Pollutant Dispersion over Complex Terrain: Challenges and Needs for Improving Air Quality Measurements and Modeling. *Atmosphere* **2020**, *11*, 646. [CrossRef]
10. Zhou, B.; Du, J. Fog prediction from a multimodel mesoscale ensemble prediction system. *Weather. Forecast.* **2010**, *25*, 303–322. [CrossRef]
11. Grubišić, V.; Doyle, J.D.; Kuettner, J.; Mobbs, S.; Smith, R.B.; Whiteman, C.D.; Dirks, R.; Czyzyk, S.; Cohn, S.A.; Vosper, S.; et al. The Terrain-Induced Rotor Experiment: A field campaign overview including observational highlights. *Bull. Am. Meteorol. Soc.* **2008**, *89*, 1513–1534. [CrossRef]
12. Chow, F.K.; De Wekker, S.F.; Snyder, B.J. *Mountain Weather Research and Forecasting: Recent Progress and Current Challenges*; Springer Science & Business Media: Dordrecht, The Netherlands, 2013; p. 760. [CrossRef]
13. Rotach, M.W.; Stiperski, I.; Fuhrer, O.; Goger, B.; Gohm, A.; Obleitner, F.; Rau, G.; Sfyri, E.; Vergeiner, J. Investigating Exchange Processes over Complex Topography: The Innsbruck Box (i-Box). *Bull. Am. Meteorol. Soc.* **2017**, *98*, 787–805. [CrossRef]
14. Lehner, M.; Rotach, M.W. Current challenges in understanding and predicting transport and exchange in the atmosphere over mountainous terrain. *Atmosphere* **2018**, *9*, 276. [CrossRef]
15. Sfyri, E.; Rotach, M.W.; Stiperski, I.; Bosveld, F.C.; Lehner, M.; Obleitner, F. Scalar-Flux Similarity in the Layer Near the Surface Over Mountainous Terrain. *Bound.-Layer Meteorol.* **2018**, *169*, 11–46. [CrossRef] [PubMed]
16. Finnigan, J.; Ayotte, K.; Harman, I.; Katul, G.; Oldroyd, H.; Patton, E.; Poggi, D.; Ross, A.; Taylor, P. Boundary-Layer Flow Over Complex Topography. *Bound.-Layer Meteorol.* **2020**, *177*, 247–313. [CrossRef]
17. Fernando, H.J.S. Fluid dynamics of urban atmospheres in complex terrain. *Annu. Rev. Fluid Mech.* **2010**, *42*, 365–389. [CrossRef]
18. Kampa, M.; Castanas, E. Human health effects of air pollution. *Environ. Pollut.* **2008**, *151*, 362–367. [CrossRef]
19. Lareau, N.P.; Crosman, E.; Whiteman, C.D.; Horel, J.D.; Hoch, S.W.; Brown, W.O.J.; Horst, T.W.; Lareau, N.P.; Crosman, E.; Whiteman, C.D.; et al. The Persistent Cold-Air Pool Study. *Bull. Am. Meteorol. Soc.* **2013**, *94*, 51–63. [CrossRef]
20. Mahrt, L. Stably stratified flow in a shallow valley. *Bound.-Layer Meteorol.* **2017**, *162*, 1–20. [CrossRef]
21. Hang, C.; Nadeau, D.F.; Gultepe, I.; Hoch, S.W.; Román-Cascón, C.; Pryor, K.; Fernando, H.J.S.; Creegan, E.D.; Leo, L.S.; Silver, Z.; et al. A case study of the mechanisms modulating the evolution of valley fog. *Pure Appl. Geophys.* **2016**, *173*, 3011–3030. [CrossRef]
22. Baskett, R.L.; Nasstrom, J.S.; Lange, R. Emergency response model evaluation using Diablo Canyon nuclear power plant tracer experiments. In *Air Pollution Modeling and Its Application VIII*; Springer: Boston, MA, USA, 1991; pp. 603–604.
23. Stohl, A.; Seibert, P.; Wotawa, G.; Arnold, D.; Burkhart, J.F.; Eckhardt, S.; Tapia, C.; Vargas, A.; Yasunari, T.J. Xenon-133 and caesium-137 releases into the atmosphere from the Fukushima Dai-ichi nuclear power plant: Determination of the source term, atmospheric dispersion, and deposition. *Atmos. Chem. Phys.* **2012**, *12*, 2313–2343. [CrossRef]
24. Adams, R.; Houston, L.; Weiher, R. *The Value of Snow and Snow Information Services*; Report Prepared for NOAA's National Operational Hydrological Remote Sensing Center, Chanhassen, MN, under contract DG1330-03-SE-1097; NOAA: Chanhassen, MN, USA, 2004.
25. Acevedo, O.C.; Fitzjarrald, D.R. The early evening surface-layer transition: Temporal and spatial variability. *J. Atmos. Sci.* **2001**, *58*, 2650–2667. [CrossRef]
26. LeMone, M.a.; Ikeda, K.; Grossman, R.L.; Rotach, M.W. Horizontal variability of 2-m temperature at night during CASES-97. *J. Atmos. Sci.* **2003**, *60*, 2431–2449. [CrossRef]
27. Shalev-Shwartz, S.; Ben-David, S. *Understanding Machine Learning: From Theory to Algorithms*; Cambridge University Press: Cambridge, MA, USA, 2014.
28. Beale, M.H.; Hagan, M.T.; Demuth, H.B. *MATLAB Neural Network Toolbox User's Guide*; Technical Report R2017b; Mathworks: Natick, MA, USA, 2017.
29. Stathakis, D. How many hidden layers and nodes? *Int. J. Remote Sens.* **2009**, *30*, 2133–2147. [CrossRef]
30. Mirchandani, G.; Cao, W. On hidden nodes for neural nets. *IEEE Trans. Circuits Syst.* **1989**, *36*, 661–664. [CrossRef]
31. Mass, C. Nowcasting: The promise of new technologies of communication, modeling, and observation. *Bull. Am. Meteorol. Soc.* **2012**, *93*, 797–809. [CrossRef]

32. Xu, K.; Wikle, C.K.; Fox, N.I. A kernel-based spatio-temporal dynamical model for nowcasting weather radar reflectivities. *J. Am. Stat. Assoc.* **2005**, *100*, 1133–1144. [CrossRef]
33. Novak, P. The Czech Hydrometeorological Institute's severe storm nowcasting system. *Atmos. Res.* **2007**, *83*, 450–457. [CrossRef]
34. Wilson, J.W.; Crook, N.A.; Mueller, C.K.; Sun, J.; Dixon, M. Nowcasting thunderstorms: A status report. *Bull. Am. Meteorol. Soc.* **1998**, *79*, 2079–2099. [CrossRef]
35. Rasmussen, R.; Dixon, M.; Hage, F.; Cole, J.; Wade, C.; Tuttle, J.; McGettigan, S.; Carty, T.; Stevenson, L.; Fellner, W.; et al. Weather Support to Deicing Decision Making (WSDDM): A winter weather nowcasting system. *Bull. Am. Meteorol. Soc.* **2001**, *82*, 579–595. [CrossRef]
36. Gultepe, I.; Kuhn, T.; Pavolonis, M.; Calvert, C.; Gurka, J.; Heymsfield, A.J.; Liu, P.S.K.; Zhou, B.; Ware, R.; Ferrier, B.; et al. Ice fog in Arctic during FRAM–Ice Fog Project: Aviation and nowcasting applications. *Bull. Am. Meteorol. Soc.* **2014**, *95*, 211–226. [CrossRef]
37. Yates, D.N.; Warner, T.T.; Leavesley, G.H. Prediction of a flash flood in complex terrain. Part II: A comparison of flood discharge simulations using rainfall input from radar, a dynamic model, and an automated algorithmic system. *J. Appl. Meteorol.* **2000**, *39*, 815–825. [CrossRef]
38. Kumar, A.; Islam, T.; Sekimoto, Y.; Mattmann, C.; Wilson, B. Convcast: An embedded convolutional LSTM based architecture for precipitation nowcasting using satellite data. *PLoS ONE* **2020**, *15*, e0230114. [CrossRef] [PubMed]
39. Demetriades, N.W.S.; Holle, R.L. Long range lightning nowcasting applications for tropical cyclones. In Proceedings of the Conference Meteorology Application of Lightning Data, Atlanta, GA, USA, 9–13 January 2006; pp. 353–365.
40. Gunawardena, N.; Pardyjak, E.; Stoll, R.; Khadka, A. Development and evaluation of an open-source, low-cost distributed sensor network for environmental monitoring applications. *Meas. Sci. Technol.* **2018**, *29*, 024008. [CrossRef]
41. Öztopal, A. Artificial neural network approach to spatial estimation of wind velocity data. *Energy Convers. Manag.* **2006**, *47*, 395–406. [CrossRef]
42. Benvenuto, F.; Marani, A. Neural networks for environmental problems: Data quality control and air pollution nowcasting. *Glob. NEST Int. J.* **2000**, *2*, 281–292.
43. Videnova, I.; Nedialkov, D.; Dimitrova, M.; Popova, S. Neural networks for air pollution nowcasting. *Appl. Artif. Intell.* **2006**, *20*, 493–506. [CrossRef]
44. Ruppert, J.; Mauder, M.; Thomas, C.; Lüers, J. Innovative gap-filling strategy for annual sums of CO2 net ecosystem exchange. *Agric. For. Meteorol.* **2006**, *138*, 5–18. [CrossRef]
45. Falge, E.; Baldocchi, D.; Olson, R.; Anthoni, P.; Aubinet, M.; Bernhofer, C.; Burba, G.; Ceulemans, R.; Clement, R.; Dolman, H.; et al. Gap filling strategies for long term energy flux data sets. *Agric. For. Meteorol.* **2001**, *107*, 71–77. [CrossRef]
46. Ehsani, M.R.; Arevalo, J.; Risanto, C.B.; Javadian, M.; Devine, C.J.; Arabzadeh, A.; Venegas-Quiñones, H.L.; Dell'Oro, A.P.; Behrangi, A. 2019–2020 Australia Fire and Its Relationship to Hydroclimatological and Vegetation Variabilities. *Water* **2020**, *12*, 3067. [CrossRef]
47. Moffat, A.M.; Papale, D.; Reichstein, M.; Hollinger, D.Y.; Richardson, A.D.; Barr, A.G.; Beckstein, C.; Braswell, B.H.; Churkina, G.; Desai, A.R.; et al. Comprehensive comparison of gap-filling techniques for eddy covariance net carbon fluxes. *Agric. For. Meteorol.* **2007**, *147*, 209–232. [CrossRef]
48. Tardivo, G.; Berti, A. A dynamic method for gap filling in daily temperature datasets. *J. Appl. Meteorol. Climatol.* **2012**, *51*, 1079–1086. [CrossRef]
49. Kemp, W.; Burnell, D.; Everson, D.; Thomson, A. Estimating missing daily maximum and minimum temperatures. *J. Clim. Appl. Meteorol.* **1983**, *22*, 1587–1593. [CrossRef]
50. Coutinho, E.R.; Silva, R.M.d.; Madeira, J.G.F.; Coutinho, P.R.d.O.d.S.; Boloy, R.A.M.; Delgado, A.R.S. Application of artificial neural networks (ANNs) in the gap filling of meteorological time series. *Rev. Bras. Meteorol.* **2018**, *33*, 317–328. [CrossRef]
51. Lahoz, W; Khattatov, B; Menard, R. *Data Assimilation*; Springer: Berlin/Heidelberg, Germany, 2010.
52. Bishop, G.; Welch, G. An Introduction to the Kalman Filter. In Proceedings of the SIGGRAPH 2001, Los Angeles, CA, USA, 12–17 August 2001; University of North Carolina at Chapel Hill: Chapel Hill, NC, USA, 2001; Volume 8, p. 41.
53. Asa, E. Nonlinear spatial characterization and interpolation of wind data. *Wind Eng.* **2012**, *36*, 251–272. [CrossRef]
54. Friedland, C.J.; Joyner, T.A.; Massarra, C.; Rohli, R.V.; Treviño, A.M.; Ghosh, S.; Huyck, C.; Weatherhead, M. Isotropic and anisotropic kriging approaches for interpolating surface-level wind speeds across large, geographically diverse regions. *Geomat. Nat. Hazards Risk* **2017**, *8*, 207–224. [CrossRef]
55. Rasmussen, C.E.; Williams, C.K.I. *Gaussian Processes for Machine Learning*; MIT Press: Cambridge, MA, USA, 2006; Volume 1.
56. Osborne, M.A.; Roberts, S.J.; Rogers, A.; Ramchurn, S.D.; Jennings, N.R. Towards real-time information processing of sensor network data using computationally efficient multi-output Gaussian processes. In Proceedings of the 2008 International Conference on Information Processing in Sensor Networks (IPSN 2008), St. Louis, MO, USA, 22–24 April 2008; pp. 109–120. [CrossRef]
57. Hart, Q.J.; Brugnach, M.; Temesgen, B.; Rueda, C.; Ustin, S.L.; Frame, K. Daily reference evapotranspiration for California using satellite imagery and weather station measurement interpolation. *Civ. Eng. Environ. Syst.* **2009**, *26*, 19–33. [CrossRef]
58. Apaydin, H.; Kemal Sonmez, F.; Yildirim, Y.E. Spatial interpolation techniques for climate data in the GAP region in Turkey. *Clim. Res.* **2004**, *28*, 31–40. [CrossRef]

59. Luo, W.; Taylor, M.C.; Parker, S.R. A comparison of spatial interpolation methods to estimate continuous wind speed surfaces using irregularly distributed data from England and Wales. *Int. J. Climatol.* **2008**, *28*, 947–959. [CrossRef]
60. Dupuy, F.; Duine, G.J.; Durand, P.; Hedde, T.; Pardyjak, E.; Roubin, P. Valley Winds at the Local Scale: Correcting Routine Weather Forecast Using Artificial Neural Networks. *Atmosphere* **2021**, *12*, 128. [CrossRef]
61. OpenStreetMap Contributors. Planet Dump. 2017. Available online: https://www.openstreetmap.org (accessed on 26 October 2021).
62. Dupuy, F.; Duine, G.J.; Durand, P.; Hedde, T.; Pardyjak, E.R.; Roubin, P. Valley-winds at the local scale: Local-scale valley wind retrieval using an artificial neural network applied to routine weather observations. *J. Appl. Meteorol. Climatol.* **2019**. [CrossRef]
63. van der Meulen, J.P.; Brandsma, T. Thermometer screen intercomparison in De Bilt (The Netherlands), Part I: Understanding the weather-dependent temperature differences. *Int. J. Climatol.* **2008**, *28*, 371–387. [CrossRef]
64. Kristensen, L. Cup anemometer behavior in turbulent environments. *J. Atmos. Ocean. Technol.* **1998**, *15*, 5–17. [CrossRef]
65. Wyngaard, J.C. Cup, propeller, vane, and sonic anemometers in turbulence research. *Annu. Rev. Fluid Mech.* **1981**, *13*, 399–423. [CrossRef]
66. Skamarock, W.C.; Klemp, J.B.; Dudhia, J.; Gill, D.O.; Barker, D.M.; Wang, W.; Powers, J.G. *A Description of the Advanced Research WRF Version 2*; Technical Report; National Center For Atmospheric Research Boulder Co., Mesoscale and Microscale Meteorology Division: Boulder, CO, USA, 2005.
67. MathWorks. *Levenberg-Marquardt Backpropagation—MATLAB Trainlm*; MathWorks: Natick, MA, USA, 2018.
68. MathWorks. *Choose a Multilayer Neural Network Training Function*; MathWorks: Natick, MA, USA, 2021.
69. Krogh, A.; Vedelsby, J. Neural network ensembles, cross validation, and active learning. In Proceedings of the Advances in Neural Information Processing Systems, Denver, CO, USA, 27–30 November 1995; pp. 231–238.
70. Hansen, L.K.; Salamon, P. Neural network ensembles. *IEEE Trans. Pattern Anal. Mach. Intell.* **1990**, *12*, 993–1001. [CrossRef]
71. MathWorks. *MATLAB Statistics and Machine Learning Toolbox User's Guide*; Technical Report R2017b; Mathworks: Natick, MA, USA, 2017.
72. Dupuy, F. Amélioration de la Connaissance et de la Prévision des Vents de Vallée en Conditions Stables: Expérimentation et Modélisation Statistique avec Réseau de Neurones Artificiels. Ph.D. Thesis, University of Toulouse III—Paul Sabatier, Toulouse, France, 2018.
73. Valavi, R.; Guillera-Arroita, G.; Lahoz-Monfort, J.J.; Elith, J. Predictive performance of presence-only species distribution models: A benchmark study with reproducible code. *Ecol. Monogr* **2021**, *92*, e01486. [CrossRef]
74. Shafizadeh-Moghadam, H.; Weng, Q.; Liu, H.; Valavi, R. Modeling the spatial variation of urban land surface temperature in relation to environmental and anthropogenic factors: A case study of Tehran, Iran. *GISci. Remote Sens.* **2020**, *57*, 483–496. [CrossRef]
75. Poole, M.A.; O'Farrell, P.N. The assumptions of the linear regression model. *Trans. Inst. Br. Geogr.* **1971**, *52*, 145–158. [CrossRef]
76. James, G.; Witten, D.; Hastie, T.; Tibshirani, R. *An Introduction to Statistical Learning*; Springer: New York, NY, USA, 2013; Volume 112.
77. McElreath, R. *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*; Chapman and Hall/CRC: Boca Raton, FL, USA, 2018.
78. Neter, J.; Kutner, M.H.; Nachtsheim, C.J.; Wasserman, W. *Applied Linear Statistical Models*; Irwin: Chicago, IL, USA, 1996.
79. Aggarwal, C.C. *Neural Networks and Deep Learning*; Springer: Berlin/Heidelberg, Germany, 2018; Volume 10, pp. 105–107.
80. Sagaut, P. *Large Eddy Simulation for Incompressible Flows: An Introduction*; Springer Science & Business Media: Berlin, Germany, 2006.
81. Ehsani, M.R.; Behrangi, A.; Adhikari, A.; Song, Y.; Huffman, G.J.; Adler, R.F.; Bolvin, D.T.; Nelkin, E.J. Assessment of the Advanced Very High Resolution Radiometer (AVHRR) for Snowfall Retrieval in High Latitudes Using CloudSat and Machine Learning. *J. Hydrometeorol.* **2021**, *22*, 1591–1608. [CrossRef]
82. Orr, G.B.; Müller, K.R. *Neural Networks: Tricks of the Trade*; Springer: Berlin, Germany, 2003.
83. Heaton, J. *Introduction to Neural Networks with Java*, 2nd ed.; Heaton Research, Inc.: St. Louis, MO, USA, 2008.