UNIVERSITY OF ICELAND

A Neural Network Approach to Predicting Gust Factors in Complex Landscape

Brynjar Geir Sigurðsson

June 2025

M.Sc. thesis
in Mechanical Engineering

FACULTY OF INDUSTRIAL ENGINEERING, MECHANICAL ENGINEERING AND COMPUTER SCIENCE

A Neural Network Approach to Predicting Gust Factors in Complex Landscape

Brynjar Geir Sigurðsson

60 ECTS thesis submitted in partial fulfillment of a
***Magister Scientiarum* degree in Mechanical Engineering**

Supervisor
Kristján Jónasson

M.Sc. Committee
Kristján Jónasson
Ólafur Pétur Pálsson
Guðrún Nína Petersen

Examiner
Halldór Björnsson

Faculty of Industrial Engineering, Mechanical Engineering and Computer Science
School of Engineering and Natural Sciences
University of Iceland
Reykjavik, June 2025

A Neural Network Approach to Predicting Gust Factors in Complex Landscape

60 ECTS thesis submitted in partial fulfillment of a M.Sc. degree in Mechanical Engineering

Faculty of Industrial Engineering, Mechanical Engineering and Computer Science
School of Engineering and Natural Sciences
University of Iceland
Dunhagi 5
107, Reykjavik Iceland

Telephone: 525 4000

Bibliographic information:
Brynjar Geir Sigurðsson (2025) *A Neural Network Approach to Predicting Gust Factors in Complex Landscape*, M.Sc. thesis, Faculty of Industrial Engineering, Mechanical Engineering and Computer Science, University of Iceland.

Reykjavik, Iceland, June 2025

Abstract

In many industries, whether it be windmill farming or transportation, being able to precisely predict the load caused by weather is crucial to prevent harm coming to people and machinery as well as increasing productivity. The efficiency of wind turbines in high wind speed conditions is a vital challenge in optimizing renewable energy systems. Gust predictions can be instrumental in loss prevention. If looking only at wind, the most destructive moments are gusts, by definition. To predict these values there are two ways, traditional numerical weather prediction systems and machine learning (or other pattern recognition) methods. Computer-aided numerical weather prediction systems originated in the 1950s, while the use of machine learning in numerical weather predictions is much more recent, having gained traction in the last decade. The goal of this thesis is to combine these two methods to predict the wind gust factor, defined as the ratio between wind gust and average wind speed. This is done by having the input data to the machine learning model be reanalysis variables that were generated by traditional numerical weather prediction systems. In addition to these variables (and derived variables), a digital elevation model (DEM) was used. A deep neural network was constructed using some set of variables and the results are compared to a baseline as well as each other so as to show the impact each feature has. Neural networks can help make wind gust predictions in complex ladscape.

Útdráttur

Hvort sem um er að ræða vindmyllubúgarða eða stöðuleiki flutningabíla er spágetan um álag vegna vinds mikilvæg til að koma í veg fyrir tjón á iðnviðum, auk þess er hægt að auka framleiðni. Hviðuspár geta verið stór þáttur í tjónaminnkun. Ef einungis er horft á vind, þá eru hviður hæsti álagspunktur samkvæmt skilgreiningu. Sögulega hefur verið notast við hefðbundin spálíkön sem reiða sig á töluleg eðlisfræðileg líkön til að spá fyrir um niðurstöður. Á síðustu árum hefur gervigreind þróast mikið og líkön verið þjálfuð til þess að spá fyrir um veður. Svo miklar framfarir hafa orðið að sum líkön geta keppt við hefðbundin veðurlíkön. Markmið þessa verkefnis er reyna að nýta bæði hefðbundin veðurlíkön og gervigreind til þess að giska á hviðustuðla, þar sem hviðustuðullinn er skilgreindur sem hlutfall hæstu hviðu og meðalvinds. Þetta er gert með því að nota endurgreiningargögn úr hefðbundnu veðurlíkani sem grunngögn í gervigreindarlíkan. Einnig eru notað hæðarlíkan til að lýsa umhverfi. Djúpt tauganet var búið til með því að velja breytur úr endurgreiningargögnunum ásamt hæðarpunktum og afleiddum breytum. Þetta líkan er svo þjálfað og niðurstöður bornar saman við grunnlíkan sem og önnur líkön með öðrum breytum til að skoða áhrif breyta. Djúp tauganet geta búið til hjálplegar spár fyrir vindhviður í flóknu landslagi.

Contents

List of Figures

List of Tables

Listings

Abbreviations

API Application Programming Interface

ASL Above Sea Level

AWS Automatic Weather Stations

CARRA Copernicus Artic Regional ReAnalysis dataset

CNN Convolutional Neural Networks

DEM Digital Elevaiton Model

ELI5 Explain Like I am 5

ECMWF European Centre for Medium-Range Weather Forecast

GCM General Circulation Model

GeoTIFF Georeferenced TIFF

GPU Graphical Processing Unit

HRES High Resolution forecas

IMO Icelandic Meteoroligcal Office

IRCA Icelandic Road and Coastal Administration

JNWPU Joint Numerical Weather Prediction Unit

LWM Large AI Weather forecast Model

MAE Mean Asbolute Error

MAPE Mean Absolute Percentage Error

NN Neural Network

NWP Numerical Weather Prediction

SENS School of Engineering and Natural Sciences

TIFF Tag Image File Format

UoI University of Iceland

Acknowledgments

Special thanks go to my advisor, Kristján Jónasson. He would help me with the actual work of the thesis and sit with go over the code with me. Thanks also go to Ólafur Pétur Pálsson, who would read over the thesis with notes and give helpful suggestions. Guðrún Nína Petersen, also provided valuable insight whenever questioned about meteoroligcal matters as well as giving notes on the thesis.

Finally, I would like to thank my parents without whom I would probably never had any want to finish.

1 Introduction

Wind gusts are brief increases in wind speed (lasting seconds) compared to the mean wind speed over intervals of one or several minutes. The gust factor is defined as the peak gust divided by the mean wind speed over a specified time period. The peak wind gust is often defined as the highest 3-second rolling-average wind speed measured over a 10-minute period, while the mean wind speed is the average of all measurements in the same interval. This thesis uses that definition. However, definitions vary: for example, the US uses a 1-minute interval, leading to approximately 14% higher values [17].

The Navier–Stokes equation (1.1) shows that changes in wind, both in time and space, depend on the pressure gradient, the oscillating force of the Earth (the Coriolis force), and frictional forces [2].

$$\frac{\delta \mathbf{V}}{\delta t} + \mathbf{V} \cdot \nabla \mathbf{V} = \underbrace{-\frac{1}{\rho}\nabla P}_{\text{pressure}} - \overbrace{f\mathbf{k} \times \mathbf{V}}^{\text{Coriolis}} - g - \underbrace{\frac{\delta(u'\omega')}{\delta z} - \frac{\delta(v'\omega')}{\delta z}}_{\text{resistance}} \tag{1.1}$$

Traditionally, numerical weather prediction (NWP) systems are used to forecast and analyze weather patterns [3]. These models describe the evolution of discretized atmospheric states using partial differential equations grounded in physics. Forecasts are typically produced every hour, or at coarser intervals for climate simulations. With increasing computational power and efficiency, the trend is to output data more frequently [20]. However, such outputs summarize conditions over each period and may not capture short-term fluctuations well, including variations in wind speed and gusts [23].

This thesis examines methods to predict the gust factor using various predictors and multiple data sources, including NWP outputs and observational data. Accurate gust predictions are crucial, as it is often the peak wind gust that will cause damaging incidents, such as structural failures, traffic accidents and downed power lines. Moreover, the prevalence of extreme wind events is expected to increase in the future [16].

1.1 Background

1.1.1 Numerical weather prediction

The history of numerical weather prediction dates back to the 1920s, when Lewis Fry Richardson pioneered the field and attempted to produce forecasts. His results were

flawed due to numerical noise. The ENIAC, built in 1945, was a general-purpose computer used—among other tasks—for weather prediction. These forecasts took 24 hours to compute and predicted 24 hours into the future. While a proof of concept, they were not operationally useful [14].

With the advent of electronic computers in the 1950s, the first operational forecasts emerged. In September 1954, Carl-Gustaf Rossby and his Stockholm-based team produced the first real-time barotropic forecasts. The following year, the Joint Numerical Weather Prediction Unit (JNWPU), based in Princeton, New Jersey, released its first 36-hour forecasts at 400, 700, and 900 mb. Although these forecasts were inferior to subjective human analyses, they demonstrated feasibility and spurred further development in the field [10]. Since then, NWP has made tremendous strides in parallel with increases in computational power and efficiency.

1.1.2 AI in weather forecasts

In the last decade, there has been another transformation in weather prediction driven by artificial intelligence (AI). Interest in AI has come in waves: progress is made, then interest diminishes, but over the past 15 years growth has been steady. Notable drivers of this wave include advances in computational power (notably parallel processing on graphics processing units, GPUs), the development of neural networks (NNs) for processing massive datasets, and the availability of large online datasets. One type of NN is the convolutional neural network (CNN), originally designed for image processing but applicable to any gridded data with spatial structure [23]. Since 2018, significant work has applied AI to weather prediction.

There are two common approaches to AI-based forecasting: combining NWP outputs with NNs, or using NNs alone. In the former, NWP forecasts can be used as inputs to NN training or integrated in other ways; in the latter, NNs are trained directly on meteorological observations, bypassing NWP entirely.

In 2018, Düben and Bauer showed that an NN could outperform a simple persistence forecast and compete with very coarse-resolution atmospheric models of similar complexity for short lead times [7]. Also in 2018, Scher developed a deep CNN to emulate a general circulation model (GCM)—a numerical model representing physical processes—by training on GCM output. This approach allowed stable emulation of model dynamics for much longer horizons [22]. These papers were proofs of concept rather than production-ready replacements for NWP, but they demonstrated that deep-learning–based models could, with further development, compete with standard models in the field.

1.1.3 Large weather models

In the last two years, there have been even more developments with the emergence of Large AI Weather forecast Models (LWMs). In 2024, Ling et al. [13] proposed a standardized definition of LWMs in meteorology, outlining three criteria, referred to as the "Three Large Rules":

1. **Large parameter count:** Typically ranging from tens of millions to billions of parameters.

2. **Multiple predictands:** Forecasting at various levels (e.g., pressure or height levels) to provide detailed atmospheric vertical structure and surface conditions.

3. **Scalability and downstream applicability:** Demonstrated, for example, by predicting cyclones even when not explicitly trained on cyclone data (e.g., GraphCast) to showcase model versatility [13].

Before 2022, LWMs had been shown to compete with traditional NWP in specific cases and to generate forecasts much faster after training, No model had been shown to be able to completely replace traditional NWP systems. In early 2022, Pathak et al. [19] introduced FourCastNet, which employs an Adaptive Fourier Neural Operator model leveraging transformer architecture instead of convolution. FourCastNet matches the performance of standard forecasting techniques at short lead times for large-scale variables and outperforms them for smaller-scale features. It generates a one-week forecast in under two seconds—orders of magnitude faster than conventional physical methods [19]. In 2022, several ML-based models demonstrated faster-than-NWP predictions after one-time training, with performance rivaling or exceeding NWP in some cases.

In 2023, Remi Lam and the GraphCast team at Google released GraphCast, which outperformed the European Centre for Medium-Range Weather Forecasts' (ECMWF) industry-standard High-Resolution Forecast (HRES). GraphCast uses a graph-based representation instead of a regular grid, operating on a global latitude–longitude grid at 0.25° resolution, which introduces nonuniform point spacing near the poles; the graph structure helps mitigate this bias [12].

1.1.4 Conclusion

Substantial progress has been made since 2018 and especially since 2022 [13]. The evolution from proof-of-concept ML methods to models competitive with—or surpassing—standard NWP has been remarkably rapid. It is important to note that training data for these large models are drawn from traditional NWP outputs,

underscoring how ML and physics-based approaches can complement each other. The coming years promise further exciting developments in machine learning based weather prediction.

1.2 Methodology and related work

In this study, data from three sources are used:

1. Three-hourly reanalysis data for Iceland (2004–2023)

2. Dense (20 by 20 m) gridded elevation data for Iceland

3. Hourly observations from Icelandic meteorological stations

The aim of the study is to improve prediction of gust factors that can be applied at any given place in Iceland. Currently, gust factor predictions can be obtained directly from NWP output, but one may also create a simple regression model (involving, e.g., wind speed and location) to predict them. To accomplish this, an NN is used as a backend to the NWP. The network is trained with gust factors from observations as ground truth, and model quality is measured by comparing predictions with either the NWP or the regression values. Note that only data at the prediction time is used; the time-series aspect of the data is not utilized.

In 2004, Hálfdán Ágústsson and Haraldur Ólafsson [1] investigated gust factor variability in complex landscapes. They used data from automatic weather stations measuring wind at 10 m above ground during 1999–2001. They examined how three parameters affected the gust factor: d_m, D, and H—the wind direction relative to a mountain, the distance to the mountain, and the height of the mountain above the station. Their main results showed that the gust factor is inversely correlated with distance and directly correlated with mountain height. Ágústsson and Ólafsson considered the effect of a dominant point upwind but did not consider the broader landscape.

1.2.1 Neural networks

To capture patterns in the data, a neural network was constructed. An NN architecture was chosen because NNs handle complex data well and can manage large numbers of parameters. This flexibility is useful when training on different types of data and allows testing various network designs. An NN uses many matrix calculations to weight input parameters and predict an output. A deep neural network (DNN) has an input layer, an output layer, and one or more hidden layers. Hidden layers transform data from the input layer until it reaches the output layer. The in-

put layer width equals the number of input parameters, and the output layer width equals the number of predictands. Models were created with different parameter sets to gauge each parameter's influence. Each model produces a single numerical output. Every layer also has an activation function; common choices include Rectified Linear Unit (ReLU), Exponential Linear Unit (ELU), and hyperbolic tangent (tanh), defined in Equations (1.2), (1.3), and (1.5).

$$r(x) := \max(0, x) \tag{1.2}$$

$$f(x) := x, \ x > 0 \tag{1.3}$$
$$f(x) := \alpha(e^x - 1), \ x \leq 0, \ \alpha > 0 \tag{1.4}$$

$$\tanh(x) := \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{1.5}$$

These activation functions control neural activation and can help stabilize the network.

1.2.2 Model evaluation

To measure the performance of these models, both in training and testing, mean absolute percentage error (MAPE) as defined in Equation (1.6) was used.

$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^{n} \frac{|y_{\text{predict}} - y_{\text{true}}|}{y_{\text{predict}}} \tag{1.6}$$

This measure was chosen because the target is the gust factor (the wind gust over the average wind). If the target had been the wind gust rather than the gust factor, mean absolute error might be more appropriate.

1.2.3 Model explainability

Neural networks are often considered mysterious black boxes [4]. To understand model predictions, explainability methods are used. One such method is Shapley values [15]. Shapley values are calculated as the average marginal contribution of a feature value across all possible coalitions. The contribution for a single feature j to a prediction $\hat{f}(X)$ is given by Equation (1.7). In Equation (1.7), x_j is the feature value, β_j its weight, and $\beta_j E[X_j]$ the mean effect estimate for feature j.

$$\phi_j(\hat{f}) = \beta_j x_j - \beta_j E[X_j] \tag{1.7}$$

For any combination of parameters, Shapley values explain the individual prediction by attributing a contribution to each feature. Other methods, like ELI5 (Explain Like I'm 5), randomly shuffle a feature and measure the effect on model performance [9].

2 Data gathering and processing

Data were sourced from several streams. The Icelandic Meteorological Office (IMO) provided measurements from weather stations across Iceland, NWP data were downloaded from the Copernicus Arctic Regional Reanalysis dataset (CARRA), and finally a land-elevation model was also provided by the IMO.

2.1 Automatic Weather Station Data

Automatic weather stations (AWS) have been set up all around the world, including Iceland. AWSs might include a variety of sensors, such as barometers, thermometers, and anemometer along with solar panels (or other power source) and some kind of data logger. IMO provided measurements from AWSs across Iceland. One of the stations used in this study can be seen in Figure (2.1). Not every station had consistently available data. Data from 327 AWSs were included looking at a period from 2004 and ended in 2023. Of these 327 stations provided by IMO, 212 were IMO stations, with the anemometer at 10 m above ground, while the remaining 115 stations were from the Icelandic Road and Coastal Administration (IRCA), with the anemometer at 6–7 m above ground [18]. The locations of these weather stations are shown in Figure 2.2.

Data from these AWSs were stored in hourly files, which aggregate the original 10-minute files; measurement errors—unrealistic spikes known as "nails"—have been removed in most cases. Each record contains the following information: date and time; station number (convertible to coordinates using another dataset of Icelandic meteorological stations); average wind speed (f); wind gust (f_g); standard deviation of the wind gust; wind direction (d); and standard deviation of the wind direction.

These measurements began at the end of the 20th century with the installation of the first AWSs, and more stations have been added in subsequent decades.

Figure 2.1: AWS at Stórhöfði in Vestmannaeyjar, south of Iceland. The anemometer can be seen as three pins in the upper-central area of the figure just slightly to the left.

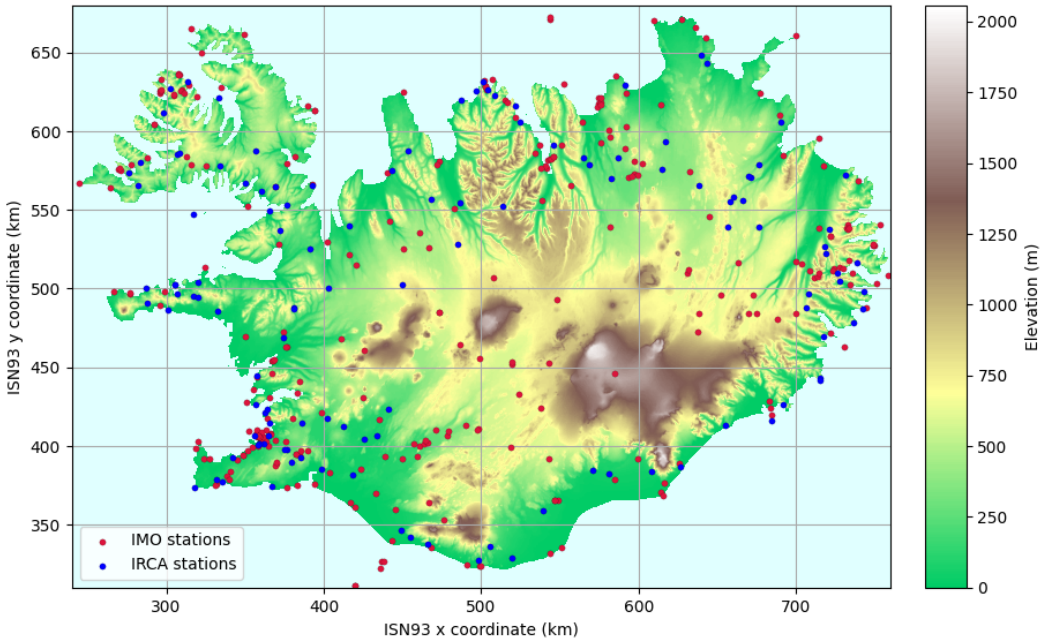


Figure 2.2: Locations of all 412 stations that were looked at in this study. Most of these were from IMO but over a hundred were from IRCA. IMO anemometers are placed at 10 meters above ground, while IRCA ones are placed at 6–7 meters above ground.

2.2 CARRA Data

CARRA is a high-resolution atmospheric reanalysis produced by the Copernicus Climate Change Service and run by ECMWF. It covers two regions, a west region covering Greenland and Iceland and an east region covering the European Arctic. It has a 2.5 km horizontal resolution and dates from 1991 to the present, with monthly updates. CARRA provides three-hourly analysis fields and short-term forecasts (hourly for lead times under 6 h and three-hourly beyond) of surface and near-surface variables—wind, temperature, pressure, precipitation, etc. It is based on the HARMONIE-AROME limited-area NWP model, forced at its boundaries by ERA5 (ECMWF Re-Analysis v. 5) and enhanced by local observations to better represent complex terrain, land–sea contrasts, and sea-ice processes. It is updated monthly, with a latency of 2-3 months [5].

The CARRA dataset covers all the IMO observations that fulfill criteria of consistent availability – the oldest observation is from 2004. The CARRA-West region covers a vastly larger area than the area of interest. This leads to having to store a large amount of data. To download CARRA data one has two options, a web interface or using an API client provided by CARRA. Using the API client is the only realistic option here, as there are thousands of requests made for different times. If using the API, it is possible to query a smaller area (such as a rectangular area around Iceland) given a set of coordinates, but this is not possible with the web interface.

The requests to the API were made at each available CARRA hour ([00, 03, 06, 09, 12, 15, 18, 21]) on a grid covering Iceland, for each available observation time. The downloaded data were interpolated to get values at the weather stations. CARRA contains several types of layers: single levels, model levels, height levels, and pressure levels. The data for this thesis was downloaded from height levels. They were requested at heights of 15, 250 and 500 meters above ground. For each point 4 parameters were requested, wind speed, wind direction, pressure, and temperature.

2.3 Elevation data

A GeoTIFF file containing a digital elevation model (DEM) for Iceland on a 20 m by 20 m grid was provided by the Icelandic Meteorological Office (IMO). The entire country is covered by this file, and its size is approximately 685 MB.

The Python package `rasterio` is used, enabling rapid elevation lookup via its spatial indexing and affine-transform capabilities. Elevation at specified geographic coordinates can be retrieved directly, and grid indices may be used for efficient access. Elevation for any exact location can be interpolated by fetching neighboring grid points.

2.4 Combining data sources

Three main data sources were used, each requiring querying, filtering, and merging to prepare the combined dataset. When handling hundreds of thousands of rows, code efficiency is essential: row-by-row iteration can increase execution time dramatically compared to vectorized operations.

The sources were provided in different formats: IMO measurement data in text files, elevation data in GeoTIFF, and CARRA reanalysis data in GRIB. To train the models, these datasets were combined into a single file using the IMO measurements as a reference. CARRA data are supplied on a rectangular grid with approximately 2.5 km spacing, while IMO observations are tied to specific station locations. Elevation data are on a 20 m by 20 m grid covering Iceland. Linear interpolation was applied to merge the sources.

The merging procedure, shown in Figure 2.3, was as follows: for each AWS observation, a query was constructed for the CARRA API by specifying year, month, day, hour, and spatial extent. Because the API returns all specified days when queried by hour, and all specified months when queried by day, monthly queries were issued for only the required days, retrieving all eight three-hourly time points (00, 03, 06, 09, 12, 15, 18, and 21 UTC). After downloading the requested variables at the desired pressure levels, point values were interpolated and appended to a pandas dataframe.

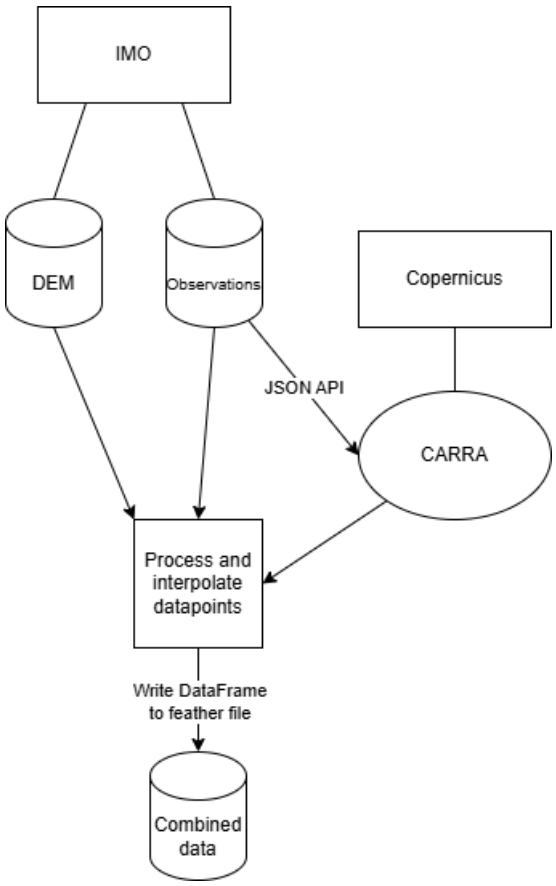


Figure 2.3: Flow chart illustrating the data-combination workflow

The monthly GRIB files were then discarded before proceeding to the next month. This strategy reduced storage needs from several terabytes to under one gigabyte.

Elevation values from the GeoTIFF were interpolated in the same manner: the four surrounding grid points were used in a linear interpolation to estimate the elevation at each station location. These interpolated values were included in the dataframe, since topography influences both average wind speed and gustiness [20].

2.5 Comparison of observed and reanalysis wind speed

Differences between reanalysis and measured wind speeds can be substantial. Absolute error increases with wind speed, while percentage error decreases. An overview of these errors by wind-speed range is shown in Table 2.1.

Next, the distribution of MAE by station is examined, considering station location and number of observations. Figure 2.4 shows this distribution, and Table 2.2 lists

Table 2.1: Comparison of measured and reanalysis wind speeds using mean absolute error (MAE) and mean absolute percentage error (MAPE). Values for observed wind speeds below 1 m/s are excluded to avoid inflated MAPE values. Measured speeds are at 10 m above ground (IMO) or 6–7 m (IRCA); reanalysis speeds are at 15 m.

f	n	MAE	MAPE
$[1; 5[$	5,260,814	2.0	83.9%
$[5; 10[$	4,150,923	2.2	31.3%
$[10; 15[$	1,480,487	2.5	21.0%
$[15; 20[$	388,905	3.0	17.8%
$[20; 25[$	84,099	4.0	18.4%
$[25; \infty[$	20,288	6.6	23.0%
$[1; \infty[$	11,385,516	2.2	53.7%

the five stations with the lowest MAE and the five with the highest MAE.

Table 2.2: Mean absolute error (MAE) between reanalysis and observed wind speeds at the five stations with lowest and highest errors. The observation counts have been rounded.

Station	n	MAE
Reykjavík Háahlíð	6,800	1.17
Keflavíkurflugvöllur	52,000	1.18
Reykjavík Víðidalur	14,000	1.23
Rif á Melrakkasléttu	13,000	1.29
Reykjavíkurflugvöllur	56,000	1.29
Almannaskarð	4,000	4.30
Kerlingarfjöll	15,000	4.36
Fáskrúðsfjarðargöng	7,900	4.40
Seljalandsdalur	1,600	4.51
Botn í Súgandafirði	32,000	4.95

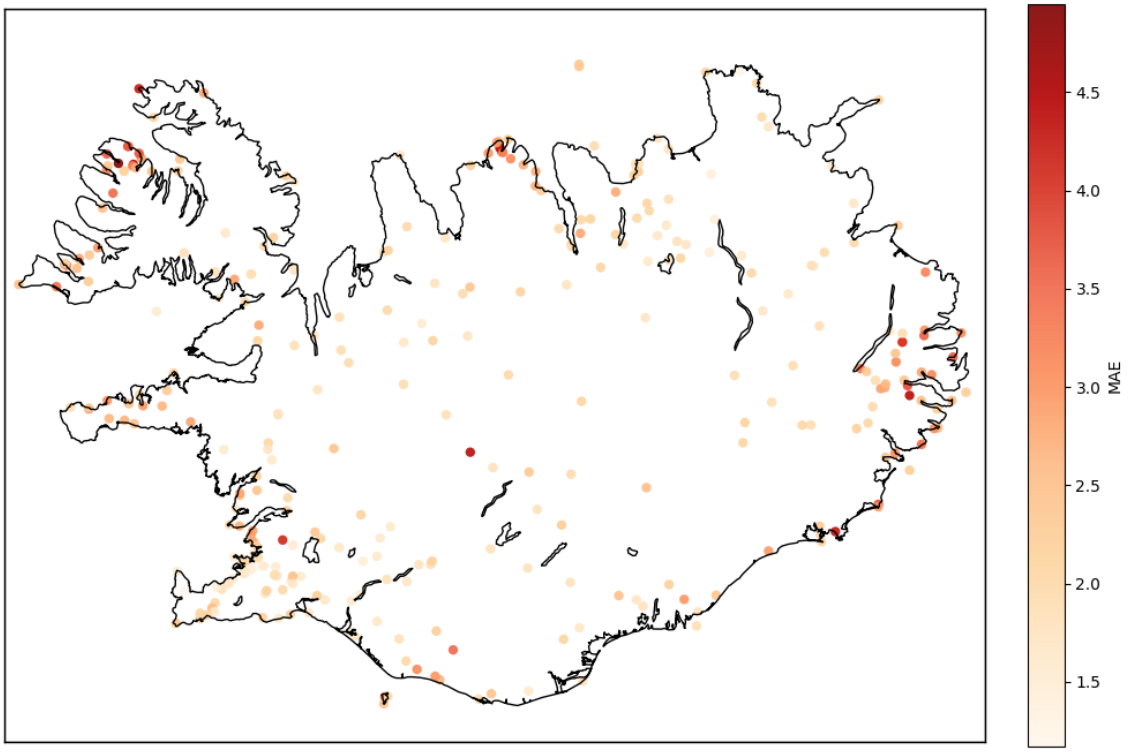


Figure 2.4: Distribution of mean absolute error (MAE) by station for all observed wind speeds.

2.6 Layout of combined data

Once data from all three sources have been retrieved and processed—including interpolation—they must be combined and formatted for modeling (training, validation, and testing). The starting point is a DataFrame containing AWS observations: average wind speed, wind gust, wind direction, station number, and coordinates. CARRA data are provided at selected height levels, each as a separate row in the CARRA DataFrame. Thus a single observation will span multiple rows, one per level. These rows are merged by time and station location, enabling AWS and CARRA data to be joined on station and time fields.

The DEM elevation data are handled by defining an upwind sector: a range of angles relative to the wind direction d and radial distances from each station. Points within this sector are generated as shown in Code Listing 2.1.

Listing 2.1: Generation of elevation points in the upwind sector

```
angles = [(angle + (90 - d)) * pi/180 for angle in angleRange]
length_rng = [(exp(i * log(n + 1)/ k) - 1) * 1000
              for i in  range(1, k + 1)]
points = np.array([[(X + l * cos(angle), Y + l * sin(angle))
                    for angle in angles] for l in length_rng])
```

The resulting DataFrame includes AWS measurements (our target), CARRA weather variables, and elevation points. An example of the combined data structure is shown in Table 2.3.

Table 2.3: Example of the combined data structure of features used for modeling. Data include derived variables Richardson number (Ri) and squared Brunt–Väisälä frequency (N^2), station altitude (meters above sea level), transformed wind direction (twd), wind speed (ws_{15}), wind direction (wd_{15}), temperature (t_{15}), pressure (p_{15}) (CARRA values at 15 m height), and elevation points (from DEM) in a sector pointing upwind.

Ri	N^2	station altitude	twd	ws_{15}	wd_{15}	t_{15}	p_{15}	elevation_0	elevation_1	...
-1.18	26700	100	1.5	10	5	0	100	2	4	...
...										

The transformed wind direction is a proxy showing whether the wind is coming from land or sea. It is computed as the angle between the wind direction and a vector from the center of Iceland to the station.

The Richardson number and the Brunt–Väisälä frequency describe atmospheric stability, see equations (2.1) and (2.2) [24, 8, 1]. These values are calculated using reanalysis data at two different height levels. Thus Ri refers to the Richardson number calculated between height levels 15m and 500m. Exactly the same notation is used with the Brunt–Väisälä frequency, except the square is used.

$$Ri = \frac{g \cdot dT \cdot dz}{T_{ave} \cdot dU^2} \, [] \tag{2.1}$$

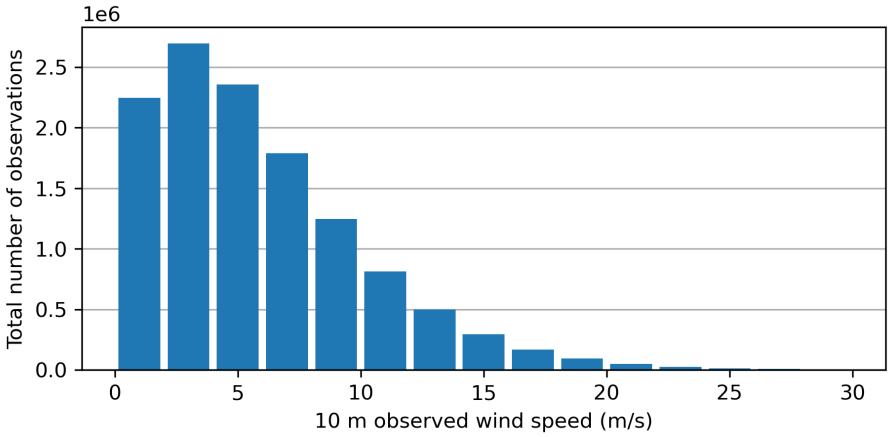
$$N = \sqrt{\frac{g \cdot dT}{T_{ave} \cdot dz}} \, [\text{Hz}] \tag{2.2}$$

Here, g is gravitational acceleration; dT the temperature difference between levels; dz the height difference; T_{ave} the average temperature; and dU the wind-speed difference. Lower Ri indicates greater turbulence, with typical values between 0.1 and 10, and values below 1 signifying significant turbulence [24]. Negative N^2 denotes instability, as an air parcel will accelerate away from its original position [8].

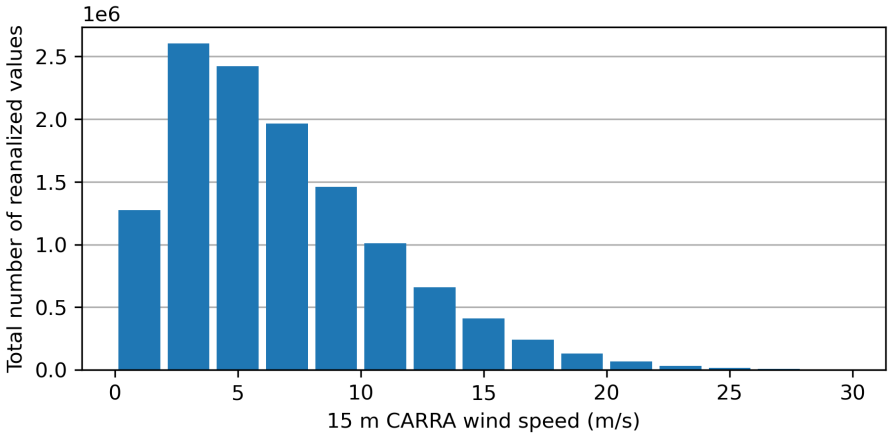
Since Ri and N^2 are derived from reanalysis variables, they add limited new information compared to the raw data, but they can reduce input dimensionality and improve model explainability via Shapley-value analysis [15]. Shapley values assess feature importance by evaluating all possible feature subsets—an operation of computational complexity 2^n. In practice, approximations are used but can still be costly for large models. Because dU appears squared in the denominator of Ri, if the wind-speed difference between levels is very small, Ri can become arbitrarily large, potentially distorting predictions.

2.7 Distributions of observed and reanalysis data

CARRA reanalysis data may exhibit bias or systematic differences when compared to measurements. Figure 2.5 shows the distributions of observed and reanalysis wind speeds. Although the shapes are similar, reanalysis values tend to be higher.



(a) Histogram of observed wind speeds from IMO and IRCA at all stations, sampled at 3-hour intervals (00, 03, 06, ... ,21 UTC).



(b) Histogram of interpolated CARRA wind speeds at station locations on a 2.5 km grid and 3-hour intervals.

Figure 2.5: Comparison of observed and reanalysis wind-speed distributions. CARRA values are spatially interpolated by linear weighting of surrounding grid points.

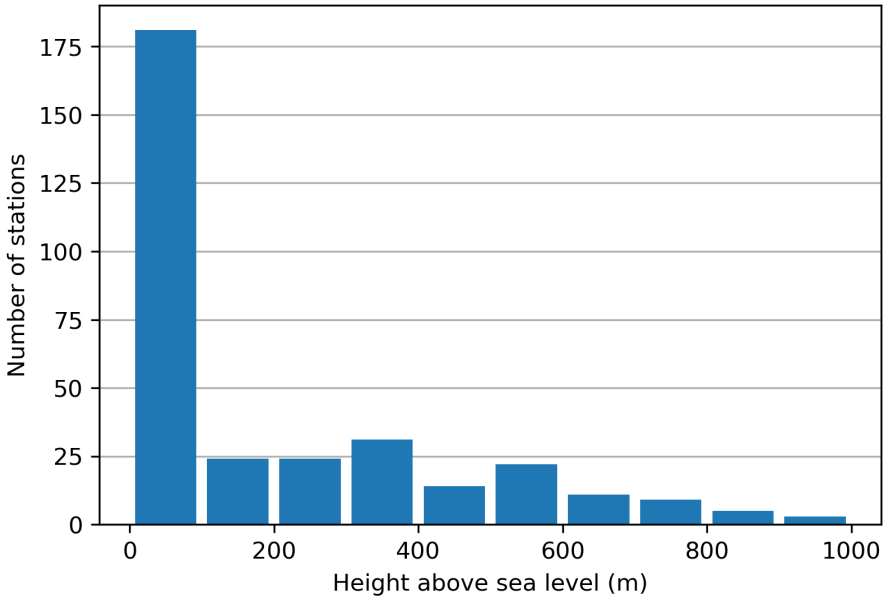


Figure 2.6: Distribution of weather station altitudes above sea level. One outlier near 2000 m was excluded due to limited and inconsistent data.

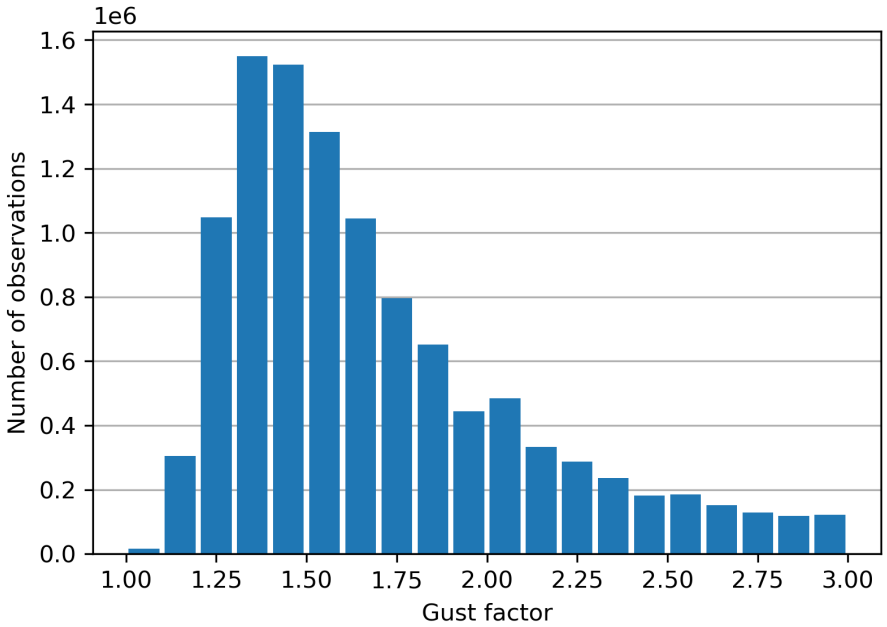


Figure 2.7: Histogram of gust factors. By definition, gust factor ≥ 1; most values lie between 1.2 and 2. The gust factor declines as wind speed increases.

3 Model Architecture

The network consists of n fully connected layers, with n around 10. Each hidden layer has the same number of units and is followed by batch normalization. A dropout layer with rate 50% is applied after the final hidden layer. A dense output layer with one unit produces the predicted gust factor. A grid search was used to find the hyperparameters that minimize the loss. Hyperparameters are settings defined before training begins [6]. They include the number of units per layer, the number of layers, the number of training epochs, batch size, optimizer choice, and regularization penalty. The ranges tested are listed in Table 3.1.

Table 3.1: Hyperparameter search was performed using the Hyperband algorithm, which begins by sampling configurations at random and then focuses on the most promising ones. The table shows the ranges tested and the best-performing combination. Unlike random search, Hyperband does not impose a fixed limit on the number of trials.

Parameter	Range of values	Selected
Layers	min_value = 4, max_value = 15, step = 1	**10**
Units	min_value = 32, max_value = 512, step = 32	**64**
Penalties	min_value = 1e-5, max_value = 1, sampling = log	**1e-4**
Epochs	min_value = 10, max_value = 1000, step = 10	**250**
Optimizers	Adam, RMSprop, Adamax	**Adamax**
Activation	ReLU, ELU, tanh	**ReLU**

As mentioned, Hyperband doesn't set a hard upper limit to the number of epochs it will train in total. When using the Hyperband class several factors can be set. One of interest here is the Hyperband iterations argument. This determines how often the Hyperband algorithm is run and defaults to 1. For each iteration the epochs are distributed between tries (that is each set of hyperparameters) with the total amount of epochs approximately $n_{epochs} = max_{epochs} * log^2(max_{epochs}) \approx 10^4$, where $max_{epochs} = 1000$ gives the maximum number of epochs that one set of hyperparameters can be trained for. Searching a space generally takes a lot of time but this drastically improves on grid search. If each epoch takes around 10 seconds to run then the total search would take around 28 hours on a shared resource. This is resource-intensive and cannot be repeated often. Another question that remains is whether the ranges given are optimal.

As noted, Hyperband does not impose a fixed limit on the total number of training epochs. When using the Hyperband tuner, the Hyperband iterations parameter controls how many times Hyperband is run and defaults to 1. On each run, the total number of epochs is distributed across the hyperparameter trials. The total

number of epochs is approximately

$$n_{\text{epochs}} = \text{max\_epochs} \times \big(\log(\text{max\_epochs})\big)^2 \approx 10^4$$

for max_epochs = 1000. Although this method is more efficient than grid search, if each epoch takes about 10 seconds, a complete Hyperband search can still require about 28 hours on shared resources, making repeated searches impractical. It remains unclear whether the current hyperparameter ranges are optimal.

4 Results

4.1 Baseline models

The simplest model for the gust factor is to guess the average gust factor for all available observations, i.e. use no variables from the training data. A slightly better model is obtained by using the wind speed as a training feature, either through regression or NN training. The MAPEs for the constructed base models are shown in Table 4.1. The ws_{15} model was obtained with NN training but regression produces an almost identical result.

Table 4.1: MAPE for several lower limits on the reanalysis 15 m wind speed (ws_{15}). Note that the ws_{15} model was only evaluated for the 10 m/s lower bound.

Wind speed limit $[m/s]$	MAPE mean model	MAPE ws_{15} model
≥ 0	39.2%	
≥ 5	28.1%	
≥ 10	23.9%	17.2%
≥ 15	23.2%	
≥ 20	24.7%	
≥ 25	27.7%	

As the average wind speed increases, then the variability in gust as a percentage decreases [1]. Therefore lower MAPE would be expected if the training data is restricted to higher wind speeds. This fact explains why several models were created for different lower bounds.

4.2 Modeling for ws_{15} above 10 m/s

As noted in Chapter 1, it is more important to predict accurate gust factor when it is windy than when the weather is calm. Therefore, at first, the data is restricted to $ws_{15} \geq 10$ m/s. The first constructed model incorporates the features temperature, pressure and wind direction at 15 m height. This gives a moderately reduced MAPE of 16.7% compared with 17.2% for the baseline ws_{15} model. When the next pair of features, station altitude and transformed wind direction, is included, the MAPE is reduced considerably, down to 13.8%. The atmospheric stability indicators, Ri and N^2, add little to the model performance, contrary to what might have been expected. When the atmosphere is unstable, wind gusts from higher up might be expected. However, another considerable performance increase is observed when digital elevation model (DEM) information is included, down to a MAPE of 12.0%.

This represents over 30% reduction in the MAPE from the ws_{15} baseline model. The resulting model will be referred to as the primary model.

Table 4.2: Models defined by a selection of parameter sets, with increasing complexity. The table shows decreasing MAPE as more variables are added.

Model variables	MAPE
[ ] Baseline	23.9%
$[ws_{15}]$ Baseline	17.2%
$[ws_{15}, t_{15}, p_{15}, wd_{15}]$	16.7%
$[ws_{15}, t_{15}, p_{15}, wd_{15}, \text{altitude}, twd]$	13.8%
$[ws_{15}, t_{15}, p_{15}, wd_{15}, \text{altitude}, twd, N^2, Ri]$	13.7%
$[ws_{15}, t_{15}, p_{15}, wd_{15}, \text{altitude}, twd, N^2, Ri]$ + DEM*	12.0%
$[ws_{15,250,500}, t_{15,250,500}, p_{15,250,500}, wd_{15,250,500}, \text{altitude}, twd, N^2, Ri]$	12.0%

*Primary model

4.3 Modeling for various ws_{15} lower bounds

To assess how model performance depends on ws_{15} lower bound several models were trained using the variable sets in rows 5 and 6 in Table 4.2, i.e. ws_{15}, t_{15}, p_{15}, wd_{15}, altitude, twd, N^2, Ri, as well as the same set together with DEM (the *primary model*). The results can be seen in Table 4.3. For this investigation comparison was only made with the constant baseline model. In all cases adding the DEM information is seen to increase the model performance. The reason for the difference between the 10 m/s lower bound results seen in Tables 4.2 and 4.3 is they were obtained with different training setups.

Table 4.3: MAPE for each ws_{15} lower bound without and with landscape elevation data (DEM) in a 30° sector into the direction of the reanalysis wind. Adding DEM data reduces the error noticably. Note that the MAPE is higher for both the lower lower bounds and for the higher ones.

Wind Speed limit $[m/s]$	MAPE Baseline	MAPE Without DEM	MAPE With DEM
≥ 0	39.2%	19.2%	18.9%
≥ 5	28.1%	15.3%	14.9%
≥ 10	23.9%	13.3%	12.5%
≥ 15	23.2%	14.4%	11.9%
≥ 20	24.7%	15.7%	13.3%
≥ 25	27.7%	19.4%	17.3%

4.4 Results for specific locations

In this section, the distribution of MAPE by station will be considered. For each station the primary model MAPE was computed using just the $ws_{15} \geq 10$ data for that station was calculated and the results are shown in Figure (4.1). The stations with low performance are all near mountains and often in fjords. The worst cases are in Vestfirðir and Austfirðir, and several bad stations are e.g. in Tröllaskagi and Snæfellsnes, and near Öræfajökull, Esja, Eyjafjallajökull. Inland stations seem to have lower error. The worst performing station is Seljalandsdalur in Vestfirðir while the best performing station is Garðskagaviti at the tip of the Reykjanes peninsula. Evidently rugged landscape makes it difficult to estimate the gust factor accurately.

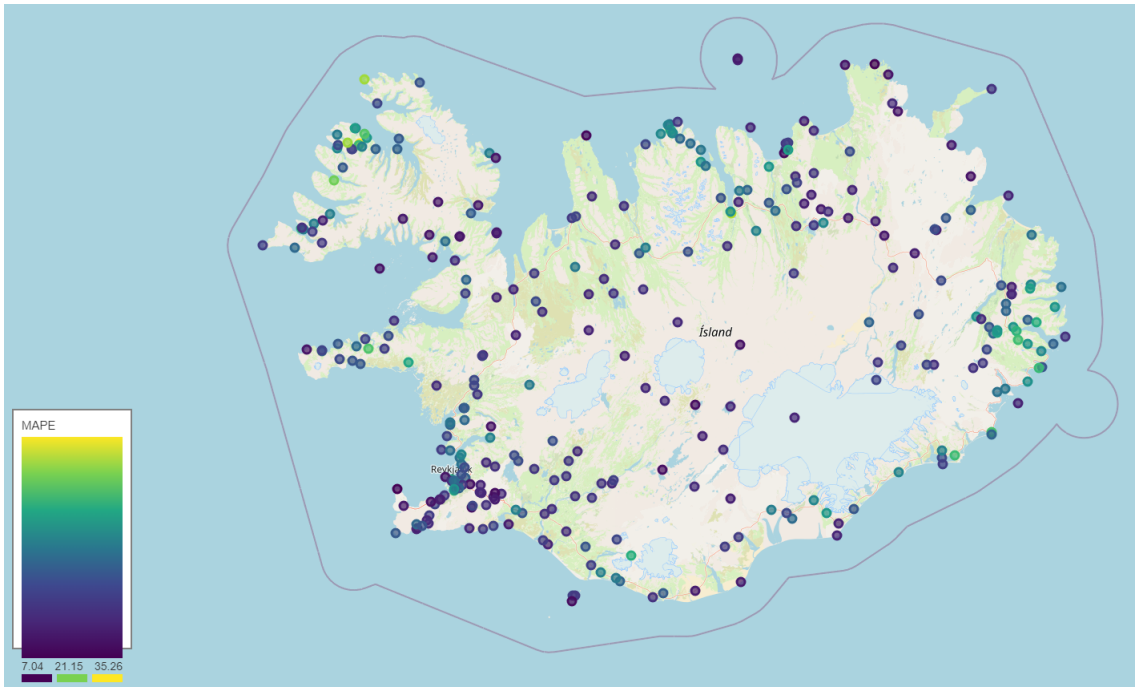


Figure 4.1: Mean absolute percentage error (MAPE) at each station, shown as colored circles (dark blue = low error; yellow = high error). The lowest station error is about 7% at Garðskagaviti and the highest about 35% at Seljalandsdalur. The model uses a 10 m/s lower bound based on ws_{15}.

After examining the MAPE at each station individually, it is logical to try accounting for the station (or its location) in the model's design. There are two ways to do this: To include the station location in the training data, and to train on subsets of the data, e.g. by station or groups of stations. Here the latter approach is considered. Results of an investigation for a few stations can be seen in Table 4.4. For many of the stations, the results are worse than expected, sometimes much worse, calling for further investigation.

Table (4.5) shows no improvement over the values in the last column in Table (4.3).

Table 4.4: MAPE for a few selected stations, for a model trained with the whole data set (general training), and for a model trained only with the data of the station itself (site training), with a lower bound of 10 m/s. This investigation was done in an earlier phase of the study, when f instead of ws_{15} was being used as the lower bound cut off, and that leads to some data leakage and a somewhat lower MAPE.

Station name	n	MAPE	
		General training	Site training
Akrafjall	43,000	18.6%	93.7%
Almannaskarð	4,000	12.2%	86.7%
Ásgarðsfjall	15,000	9.1%	9.4%
Jökulheimar	17,000	7.7%	7.7%
Sandbúðir	19,000	6.8%	6.4%
Stórholt	35,000	7.1%	29.2%
Þúfuver	20,000	6.4%	6.8%

As previously mentioned, the gust factor decreases with increasing wind speed and thus, training on intervals and lessening this effect might be expected to give better results. This does not seem to be the case. Some interesting sites to look closer at for drivers might include places such as Ingólfsfjall, Kjalarnes and others. These can be seen in Table (4.6).

Table 4.5: The MAPE results for different wind speed limit intervals. Here instead of training for all data above a certain threshold of wind speed, training is done only on data between two wind speeds. The percentage variance in gust factor as a function of wind speed increases with decreasing wind speed. Measured wind speed is used for the cutoff and thus have data leakage. These results should thus be somewhat comparable to the last column in Table (4.3)

Interval	MAPE	
$[m/s]$	Without Elevation	With Elevation
$[5, 10[$	17.1%	16.4%
$[10, 15[$	14.5%	13.0%
$[15, 20[$	15.0%	12.0%
$[20, 25[$	15.6 %	13.1%
$[25, 30[$	18.4%	19.0%

Table 4.6: The MAPE results of different stations for several stations of interest, both when training for the specific site and when the stations are a part of the general data. For every station the wind speed limit is set at 10 m/s. In training for a single station at a time, some site specific information can be gauged. This does not mean that the a better result can be reached for that site. Factors such as the number of datapoints at given location can significantly impact the result. This table uses the measured wind speed to determine the cut off for data points. This leads to some data leakage and an increased performance compared to using the reanalysis CARRA speed for cut off.

Station name	MAPE	
	Baseline	Model
Fáskrúðsfjörður	28.2%	21.8%
Ingólfsfjall	30.0%	19.6%
Kjalarnes	20.7%	13.5%
Sandskeið	13.0%	10.2%
Seyðisfjörður	32.1%	23.0%
Þjórsárdalur	12.2%	11.4%
Þrengsli	13.6%	11.3%

4.5 Comparison for models with variable parameters

In Figure (4.2) the contribution of each feature, excluding the elevation points, can be seen for the model in general (a significant subset of data is used). Looking at Figure (4.2), there is an outlier. Exactly calculating the Shapley values is time intensive, in the subset shown there is an outlier that skews the figure and makes it so that viewing the importance distribution excluding the outlier is difficult. For this reason, another shapley summary was created that looked at different, and larger distribution. This can be seen in Figure (4.3). The Figures (4.2) and (4.3) show the summary for a model trained only on the features shown and not on landscape elevation of the surrounding area. This is done as the number of points there is too high to display in one figure (70 total points). Looking specifically at Figure (4.3), the station elevation is most influential and the Richardson number's influence is very low. Most of the feature values bunch up in the middle, while the station elevation is elongated compared to other features. Station elevation seems to have two bunches on either side of 0. Overall, the values seem to be skewed to the right of 0. This would be expected as the predicted values are expected to be in the range of 1.2-2, or at least always above 1 by definition. Finally, for the Shapley values, looking that all the data there are again outliers that skew the data so that spotting general distribution is difficult. This can be seen in Figure (A.4).

Further Shapley figures can be seen in Appendix 5. In each of these plots, even without the feature labels, the station elevation is easily noticed as the value is constant for a station. This is not noteworthy. What is noteworthy is the range of impact from this single value. For simpler models, this would not happen. It is important to note that SHAP assumes feature independence [21]. This might explain why the impact of the Richardson number is so low. Both the squared Brunt–Väisälä frequency and the Richardson number are derived features from reanalysis data. They carry with them some extra information over the other features in the dataset. This is because both are variables over elevation ranges. That is, as seen in Equations (2.1, 2.2), both are dependent on values at lower and upper elevations and try to describe the stability of that range. Shapley tries to assign contribution values for each feature for each observation. SHAP assumes that the features are independent, but this is not the case. It is clearly not the case for the derived variables, but how the contribution should be distributed between the features is not clear. Seemingly the SHAP python package is giving all the impact to the Brunt–Väisälä squared frequency and none to Richardson number. If the Brunt–Väisälä would be excluded from the data, the impact of the Richardson number would likely increase. Another point to note is that the features are ordered by their impact. This means that the station elevation is the most impactful for each plot, but the ordering of other variables changes. Looking at Figure (A.4), which shows the Shapley summary plot for all data, the wind speed is the second most important feature. This is reversed in Figure (A.5). This is not unexpected as Akrafjall station was specifically selected

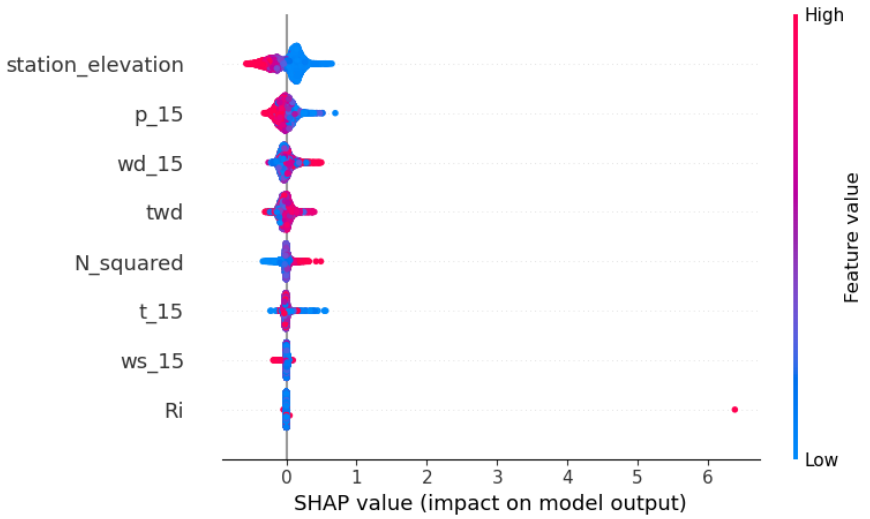


Figure 4.2: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. We can see that generally multiple factors influence the prediction, with the station elevation being highly influential. There is seemingly one outlier for the Richardson number, which usually has very little influence. Elevation data is excluded when working with Shapley values, as the contribution of each elevation point is very low and there are very many of them. To see their influence on the model output see Table 4.3.

as the MAE for reanalysis wind speed was very high as can be seen in Table (A.5), where you will also find the stations whose summary plot is shown in Figures (A.6, A.7) and these also fall into the category of very high MAE for wind speed. What is interesting is that the reanalysis wind speed is also of low impact at stations like Háahlíð and Keflavíkurflugvöllur, as shown in Figures (A.8, A.9). These stations had the lowest of MAE for difference between measured wind speed and reanalysis wind speed. As the summary plot over all stations (Figure (A.4)) shows that reanalysis wind speed is impactful, might lead to the conclusion that the reanalysis wind speed is not a good predictor at these locations or something else is skewing the data. A simpler way to look at feature importance is to create models that are trained on and use to predict different sets of parameters. The results of such a comparison can be seen in Table 4.2.

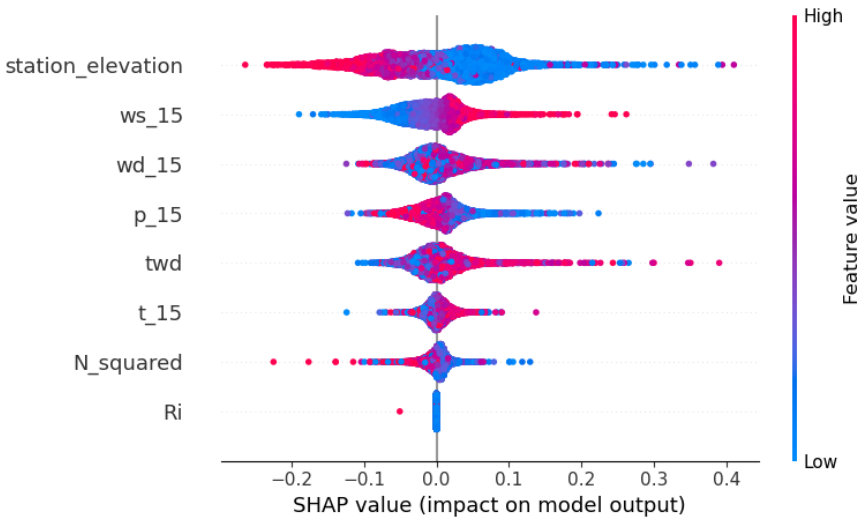


Figure 4.3: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. Generally multiple factors influence the prediction, with the station elevation being highly influential. Elevation data is excluded when working with Shapley values, as the contribution of each elevation. In contrast to Figure (4.2), the distribution doesn't have as extreme outliers. This means that more details can be seen in the figure. The X-axis shows the influence of feature values on the model. The color gradient shows the value of each feature. As an example, there is a very red value for station elevation (top line, all the way to the left). This means that in this instance, the station elevation contributed around -0.25 to the final output and that the station had an elevation significantly above average.

5 Discussion and conclusions

As shown in Table 4.3, the neural-network model achieves a significant reduction in mean absolute percentage error (MAPE) compared to a constant baseline. Table 4.2 further demonstrates an improvement over a simple regression model. A statistical test yields $p < 0.01$, confirming that these improvements are unlikely to be due to chance. The influence of individual features on model performance is illustrated by Shapley plots (Figure A.9 and Appendix 5). Although one might expect stability indicators—such as the Richardson number and the Brunt–Väisälä frequency—to be among the strongest predictors of gust variability, they contributed less than other variables. For reanalysis wind speeds above 10 m/s, the network reduces error by about 30% relative to the regression baseline. The largest gains appear in the 10–25 m/s range (Table 4.5). At lower speeds, percentage errors inflate even when absolute gust–wind differences remain small; at higher speeds, sample sizes become sparse.

Predictive skill varies significantly from station to station. This variability can be partly attributed to discrepancies between reanalysis and observations (Table 2.2), differences in local terrain, and the number of measurements per site. Training separate models for individual stations did not improve performance (Tables 4.4 and 4.6); in fact, some well-sampled stations performed worse when modeled in isolation. Future work should investigate whether these anomalies arise from systematic measurement errors, inherent atmospheric variability at those sites, or other factors.

This study did not include station coordinates (XY) as direct inputs. Adding location information could improve site-specific accuracy without losing the benefits of training on the full dataset. Similarly, instead of applying a fixed linear interpolation to CARRA and elevation grids, the network could receive raw neighboring grid values and their relative positions, allowing it to learn an optimal interpolation scheme.

Finally, treating the data as a time series may offer further gains. Here, only instantaneous inputs at forecast time were used, partly because CARRA fields are available every three hours and may not capture rapid stability changes. Access to higher-frequency data or the inclusion of temporal history could enable models to exploit evolving atmospheric conditions and further enhance gust-prediction skill.

The power generated by wind turbines increases with the cube of the wind speed [11]. The highest wind gusts in Iceland are around 70 m/s. Knowing the gust factor with half as much error as before can allow better anticipation and thus spare turbines for high wind gusts.

Bibliography

[1] Hálfdán Ágústsson and Haraldur Ólafsson. "Mean gust factors in complex terrain". In: *Meteorologische Zeitschrift* 13 (Apr. 2004), pp. 149–155. DOI: 10.1127/0941-2948/2004/0013-0149.

[2] Haraldur Author Ólafsson and Jian-Wen Author Bao. *Uncertainties in Numerical Weather Predictions*. 1st edition. Accessed on 26th of March 2024, from lecture notes of Ólafsson, Haraldur in course EÐL401M at UoI, which referenced the book. Elsevier, 2020. ISBN: 9780128154915. URL: https://uhi.on.worldcat.org/oclc/1225354709.

[3] Kaifeng Bi et al. "Accurate medium-range global weather forecasting with 3D neural networks". In: *Nature* 619 (July 2023), pp. 533–538. DOI: 10.1038/s41586-023-06185-3. URL: https://doi.org/10.1038/s41586-023-06185-3.

[4] Charles Q. Choi. *200-Year-Old Math Opens Up AI's Mysterious Black Box*. 2023. URL: https://spectrum.ieee.org/black-box-ai (visited on 04/09/2024).

[5] Copernicus. *Copernicus Arctic Regional Reanalysis data now updated monthly*. URL: https://climate.copernicus.eu/copernicus-arctic-regional-reanalysis-data-now-updated-monthly (visited on 04/08/2024).

[6] DeepAI. *What is a hyperparameter?* URL: https://deepai.org/machine-learning-glossary-and-terms/hyperparameter (visited on 04/03/2024).

[7] P. D. Dueben and P. Bauer. "Challenges and design choices for global weather and climate models based on machine learning". In: *Geoscientific Model Development* 11.10 (2018), pp. 3999–4009. DOI: 10.5194/gmd-11-3999-2018. URL: https://gmd.copernicus.org/articles/11/3999/2018/.

[8] Eumtrain. *The Brunt-Väisälä Frequency*. 2017. URL: https://resources.eumetrain.org/data/4/452/navmenu.php?tab=4&page=4.0.0 (visited on 04/03/2024).

[9] Mikhail Korobov and Konstantin Lopuhin. *ELI5 documentation*. 2017. URL: https://eli5.readthedocs.io/en/latest/overview.html (visited on 04/09/2024).

[10] H. Kristine et al. "50th anniversary of operational numerical weather prediction". In: *Bulletin of the American Meteorological* 88 (5 May 2007), pp. 639–650. DOI: 10.1175/BAMS-88-5-639. URL: https://doi.org/10.1175/BAMS-88-5-639.

[11] Franklin A. Mendoza L. 2024. URL: https://www.linkedin.com/advice/0/how-can-you-calculate-power-output-wind-turbine-viidf (visited on 05/06/2024).

[12] Remi Lam et al. *GraphCast: Learning skillful medium-range global weather forecasting*. 2023. arXiv: 2212.12794 [cs.LG].

[13] Fenghua Ling et al. *Is Artificial Intelligence Providing the Second Revolution for Weather Forecasting?* 2024. arXiv: 2401.16669 [cs.LG].

[14] Peter Lynch. "The ENIAC Forecasts: A Re-creation". In: *Bulletin of the American Meteorological Society* 89.1 (2008), pp. 45–56. DOI: 10.1175/BAMS-89-1-45. URL: https://journals.ametsoc.org/view/journals/bams/89/1/bams-89-1-45.xml.

[15] Cristoph Molnar. *Interpretable Machine Learning. A Guide for Making Black Box Models Explainable*. 2nd edition. Independently published, 2022. ISBN: 979-8411463330. URL: https://christophm.github.io/interpretable-ml-book/.

[16] NASA. *Extreme Weather and Climate Change*. NASA Website. The article site was last updated in March of 2024, not original publish date. Mar. 2024. URL: https://science.nasa.gov/climate-change/extreme-weather/.

[17] Rachelle Oblack. *Causes of Wind Gusts and Squalls*. Thought Co. website. An article by Rachelle Oblack on the Thought Co. website. She is a textbook writer for Holt McDougal. Apr. 2018. URL: https://www.thoughtco.com/why-wind-gusts-3444339.

[18] Einar Pálsson. *Personal Communication*. Email correspondance. Sent to author on 9/4/2024 from einar.palsson@vegagerdin.is. 2024.

[19] Jaideep Pathak et al. *FourCastNet: A Global Data-driven High-resolution Weather Model using Adaptive Fourier Neural Operators*. 2022. arXiv: 2202.11214 [physics.ao-ph].

[20] Guðrún Nína Petersen. *Meeting at Veðurstofa*. In-person meeting. Date of communication: November 14th, 2023. Nov. 2023.

[21] Ahmed M. Salih et al. "A Perspective on Explainable Artificial Intelligence Methods: SHAP and LIME". In: *Advanced Intelligent Systems* (June 2024). ISSN: 2640-4567. DOI: 10.1002/aisy.202400304. URL: http://dx.doi.org/10.1002/aisy.202400304.

[22] Sebastian Scher. "Toward Data-Driven Weather and Climate Forecasting: Approximating a Simple General Circulation Model With Deep Learning". In: *Geophysical Research Letters* 45 (Nov. 2018). DOI: 10.1029/2018GL080704.

[23] Martin Schultz et al. "Can deep learning beat numerical weather prediction?" In: *Philosophical Transactions of The Royal Society A Mathematical Physical and Engineering Sciences* 379 (Feb. 2021). DOI: 10.1098/rsta.2020.0097. URL: https://doi.org/10.1098/rsta.2020.0097.

[24] Skybrary. *Richardson Number*. 2022. URL: https://skybrary.aero/articles/richardson-number (visited on 04/03/2024).

Appendix A: Feature importance on Shapley plots

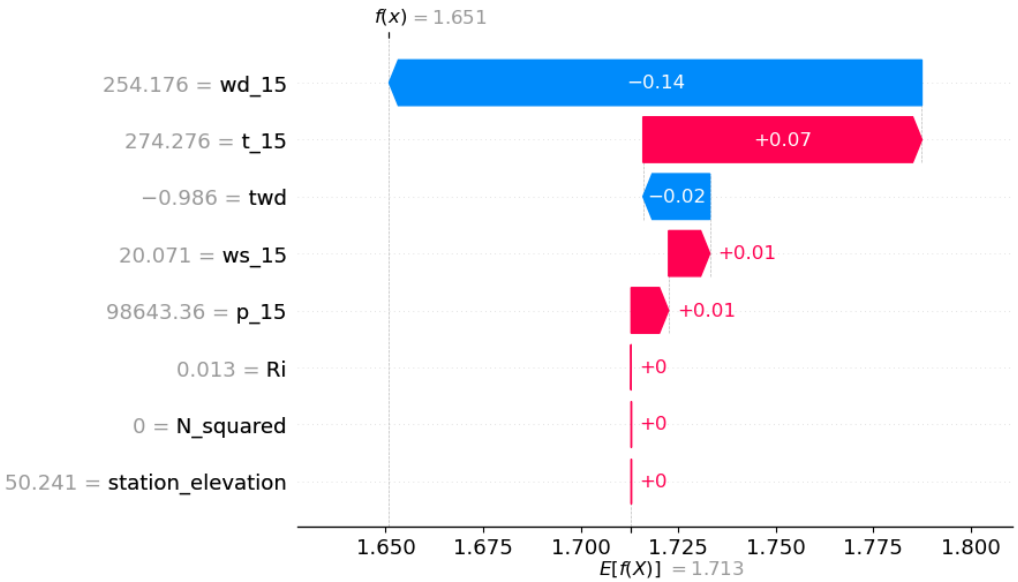


Figure A.1: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. In this specific instance the wind direction (wd_{15}) has the highest negative influence and the temperature has the highest positive influence. Elevation data is excluded when working with Shapley values, as the contribution of each elevation point is very low and there are very many of them. To see their influence on the model output see Table 4.3.

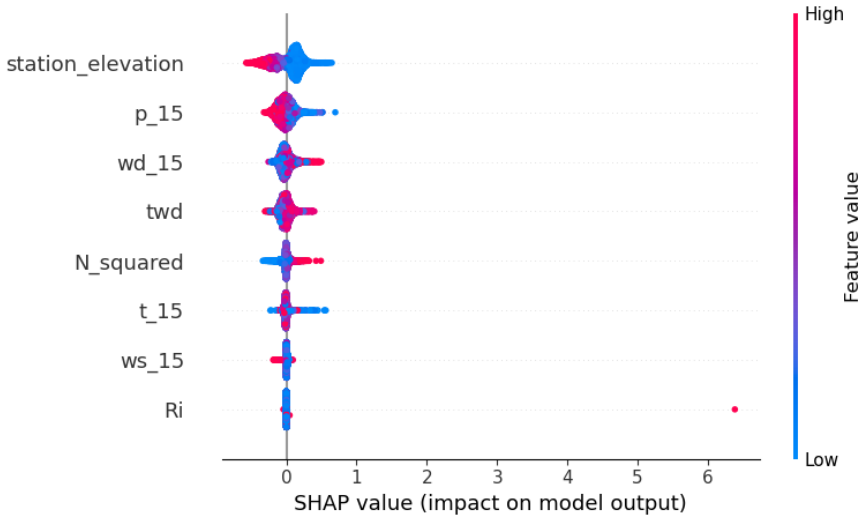


Figure A.2: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. We can see that generally multiple factors influence the prediction, with the station elevation being highly influential. There is seemingly one outlier for the Richardson number, which usually has very little influence. Elevation data is excluded when working with Shapley values, as the contribution of each elevation point is very low and there are very many of them. To see their influence on the model output see Table 4.3.

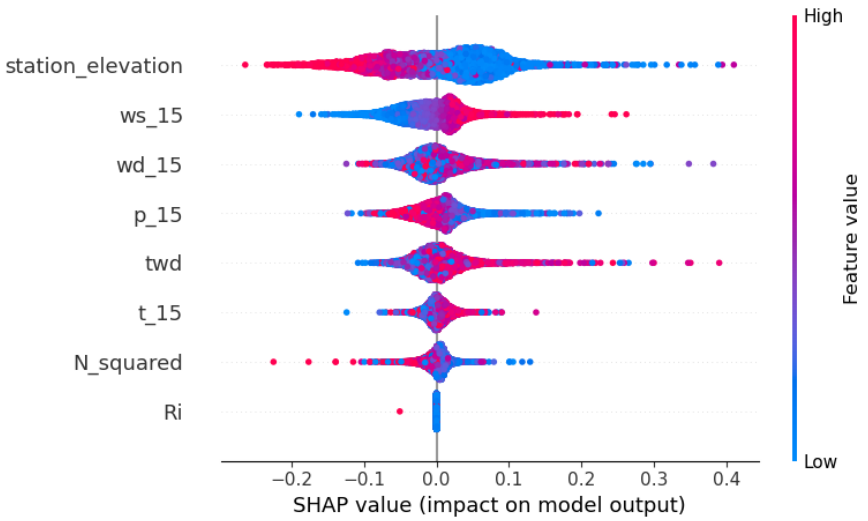


Figure A.3: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. Generally multiple factors influence the prediction, with the station elevation being highly influential. Elevation data is excluded when working with Shapley values, as the contribution of each elevation. In contrast to Figure (4.2), the distribution doesn't have as extreme outliers. This means that more details can be seen in the figure. The X-axis shows the influence of feature values on the model. The color gradient shows the value of each feature. As an example, there is a very red value for station elevation (top line, all the way to the left). This means that in this instance, the station elevation contributed around -0.25 to the final output and that the station had an elevation significantly above average.

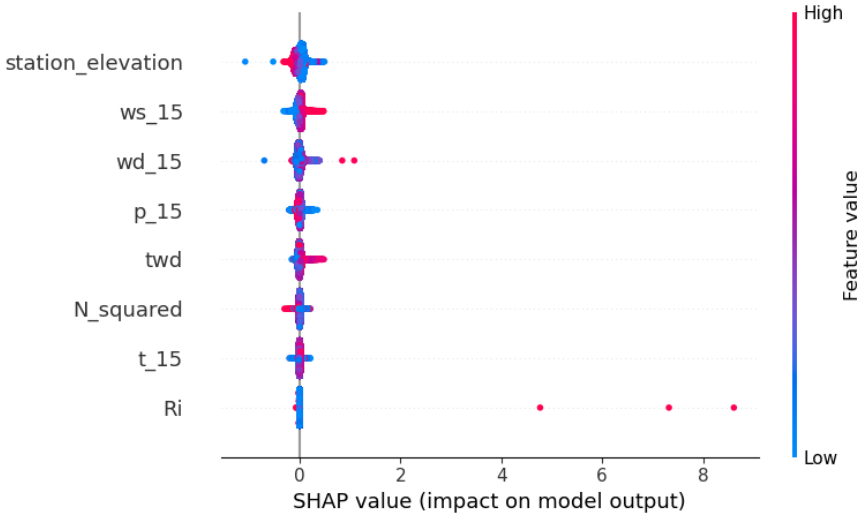


Figure A.4: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. The distribution seems to be the same as before, discounting the outliers.

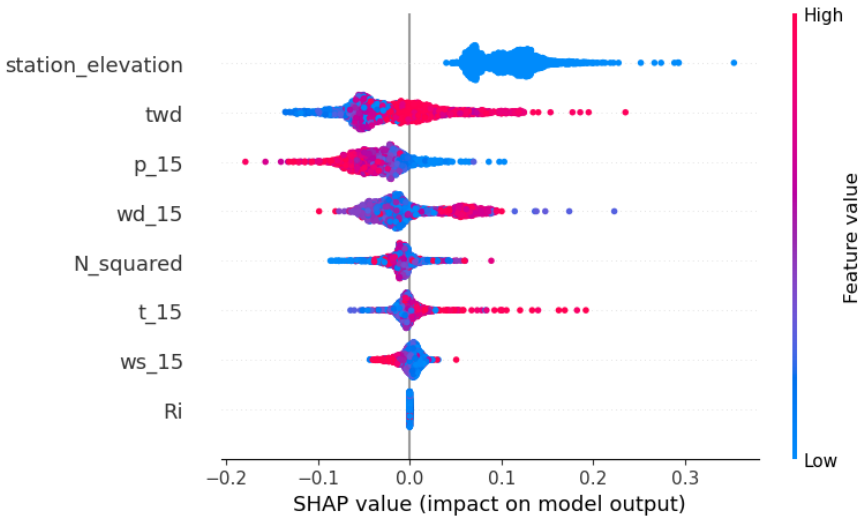


Figure A.5: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. This plot only looks at datapoints from Akrafjall. This seems to show the same distribution as previous summary plots. Station elevation is influential and Richardson number has no impact.

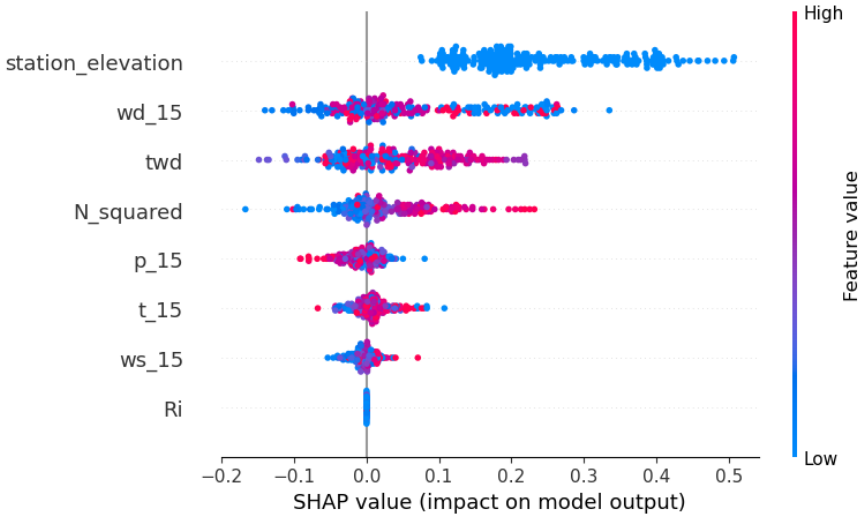


Figure A.6: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. This plot only looks at datapoints from Almannaskarð. This seems to show the same distribution as previous summary plots. Station elevation is influential and Richardson number has no impact.

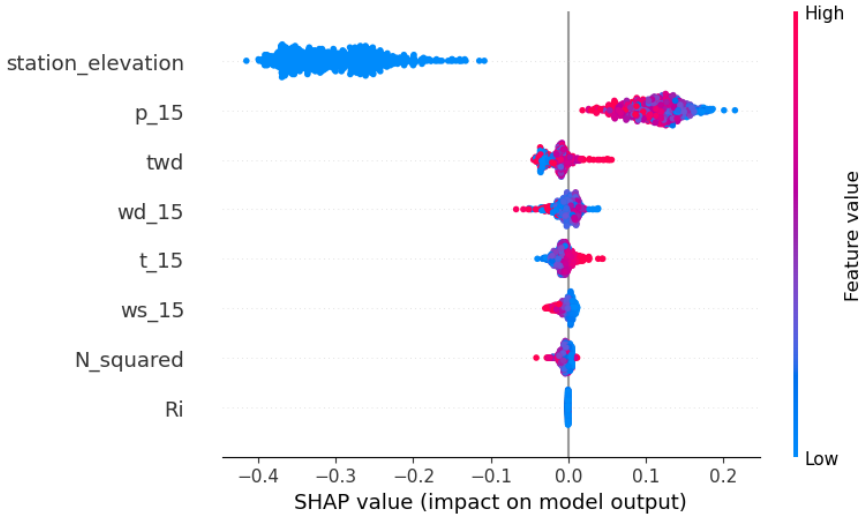


Figure A.7: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. This plot only looks at datapoints from Ásgarðsfjall. This seems to show the same distribution as previous summary plots. Station elevation is influential and Richardson number has no impact.

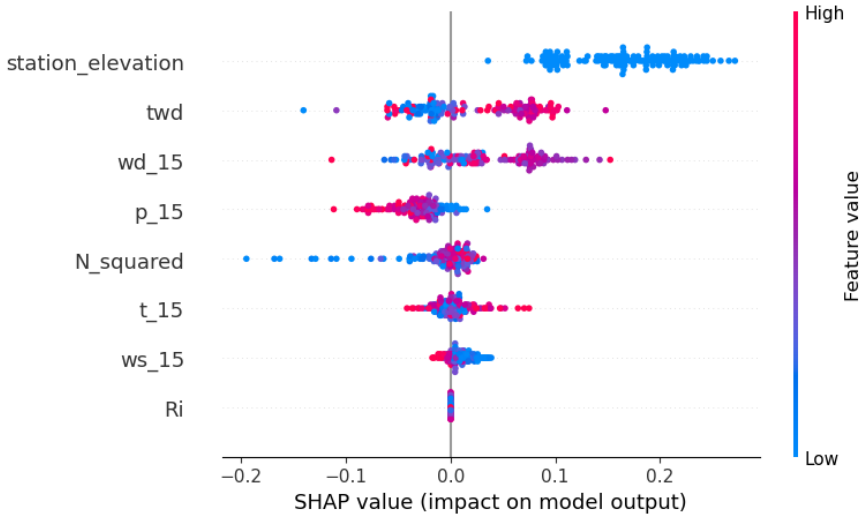


Figure A.8: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. This plot only looks at datapoints from Háahlíð. This seems to show the same distribution as previous summary plots. Station elevation is influential and Richardson number has no impact.

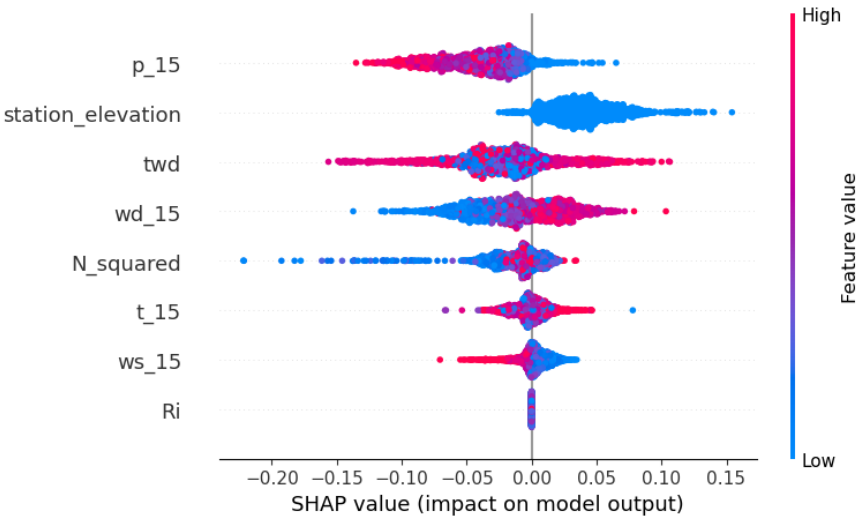


Figure A.9: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. This plot only looks at datapoints from Keflavíkurflugvöllur. This seems to show the same distribution as previous summary plots. Station elevation is influential and Richardson number has no impact.