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
XXNN3XX

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

To all the students who made the wise decision to use LaTeX.

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 systems. Computer aided numerical weather prediction systems originated in the 1950's, while the use of machine learning in numerical weather predictions is much more recent, having truly gained traction in the last decade. The goal of this thesis is try to combine these two methods to try and predict wind gust factor, where the gust factor is 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) data 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 flutningar 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 út frá 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 giska á 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
AWSL	Average Wind Speed Limit
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 increase in wind speed (lasting seconds) as compared to mean wind speed. The gust factor is defined as the peak gust divided by the mean wind speed over some defined time period. The peak wind gust is often defined as the highest 3 second rolling average measured wind speed over a period of 10 minutes, while the mean wind is the average of all measurements in the 10 minute intervalf. This thesis uses this definition. This varies, with the US using a 1 minute interval, leading to 14% higher results [17]. The Navier Stokes Equation (1.1) shows that the change of the wind, in time and space, is dependent upon the pressure gradient, the oscillating force of the earth (the Coriolis force), and frictional force.[2]

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

Traditionally, numerical weather prediction (NWP) systems are used to forecast and analyze weather patterns[3]. These models describe the transition between discretized packages of atmospheric states using partial differential equations based on physical reality. These results are usually published every hour, or at courser time intervals for climate simulations. With increasing computer power and efficiency the trend is to output data more often[20].They describe the state over the period and so do not necessarily grasp fluctuations well. These fluctuations would include fluctuations in the wind speed, wind gusts[23].

This thesis looks at how best to predict gust factor based on various factors, using several different data sources, including NWP and observations. Being able to accurately predict the wind gust is important as it is often the peak wind gusts that will cause failures in structures. A problem that will become increasingly prevalent in the near future[16].

1.1 Background

The history of numerical weather predictions goes all the way back to the 1920's when Lewis Fry Richardson pioneered the field and tried to produce forecasts. The results were flawed due to noise in the calculations. ENIAC was built in 1945, it was a general purpose computer that was used, among other things, to make predictions. These predictions took 24 hours to make and were predicting 24 hours into the future. It was a proof of concept but not usable[14]. In the 1950's, with the advent of computers the first operational forecasts emerged. In September of 1954, Rossby and his Stockholm based team produced the first real-time barotropic forecasts. The next year the Joint Numerical Weather Prediction Unit (JNWPU), based in Princeton New Jersey, released their first forecasts. These forecasts were for 36 hours at 400, 700 and 900 mb. The results were inferior to subjective human-based forecasts but showed that such forecasts were feasible and promoted further development in the area [10]. The field of NWP has taken great strides since then following the development of computer power and efficiency.

In the last decade there has been another transformation in the field of weather prediction driven by artificial intelligence. Interest in AI has come in waves. Some progress is made, then interest dwindles. Interest in AI has been increasing steadily since 2010. Notable work that has driven this wave of interest include increase in computational abilities due to parallel processing in graphical processing units (GPU), convolutional neural networks (CNN), which allowed much faster processing of massive (image) datasets and the availability of large datasets online. It is to be noted that images are grid data with some number of channels. Using CNNs could work on any gridded data where there are some spatial features[23]. Since 2018, there has been significant work done in the weather prediction field using AI. In 2018, Dueben and Bauer showed that you can build a NN that can outperform a simple persistence forecast and is competitive with very coarse-resolution atmosphere models of similar complexity for short lead times[7]. Also in 2018, Scher created a deep convolutional neural network (CNN) to emulate a general circulation model (GCM, a numerical model representing the physical processes), training on the GCM which allows it to emulate the dynamics of the model and maintain stability for much longer than Dueben[22]. These two papers were more proof of concept rather than production ready models to replace NWP. They showed that models based on deep learning might, with further development, compete with standard models in the field.

In the last two years there have been even more developments with the emergence of Large AI Weather forecast Models (LWM). In 2024, Ling et al.[13] tried to standardize the definition of LWM in meteorology and came up with 3 rules that need to be met to count as LWM.

Rule 1: **Large Parameter Count:** The number of parameters can vary wildly but a general range might be from tens of millions to billions of parameters

Rule 2: **Large Number of Predictands:** predicting on different levels (such as pressure levels or height levels) and offering detailed information on the atmospheric vertical structure and surface conditions

Rule 3: **Scalability and downstream applicability:** This might crystallize in predicting cyclones. Often, the teams responsible for creating these models try to show their applicability to predict cyclones when not trained specifically on cyclone data (e.g. GraphCast)[13]. This is done to show the versatility of the models.

Before 2022, LWM had been shown to be able to compete with traditional NWP for some specifc cases as well as making predictions quicker, after training. No model had shown that it could in any way completely replace the traditional systems. In early 2022, Pathak et al.[19] presented FourCastNet. FourCastNet uses an Adaptive Fourier Neural Operator model that leverages transfomer architecture rather than the popular convolutional model architechture. FourCastNet matches the performance of standard forecasting techniques at short lead times for large-scale variables and outperforms for smaller variables. It generates a week-long forecast in less than 2 seconds, orders of magnitude faster than standard physical methods[19]. In 2022, machine learning methods were presented that made predictions much faster than traditional NWP, after a one time training (or at least training that wouldn't have to be redone often). These were in some cases performing better than NWP. In 2023, Remi Lam and the GraphCast team at Google introduced GraphCast. This model was able to outperform the industry standard High RESolution Forecast (HRES) produced by the European Centre for Medium-Range Weather Forecasts (ECMWF). This model as the name suggests leverages graphing connections rather than traditional grid like data structure. The base data is given in latitude and longitude degrees at a resolution of 0.25 degrees. This means points are closer to each other at the poles. Using the graphing structure is supposed to help with bias incurred as a result of this[12].

There has been a lot of progress made over the last 6 years (since 2018) and especially in the last 2 years (since 2022)[13]. The progression from machine learning methods being an interesting idea in the field of numerical weather predictions, to outperforming the standard NWP has been remarkably quick. Two years ago, machine learning methods were able to predict quickly and in some niche cases outperform traditional models. They were not generally competitive with standard weather models. Now they are competitive. It is worth noting that the training of these large models is based on data from traditional large weather models. It will be very interesting to watch what the next few years will have in store for the development

of machine learning in weather predictions.

1.2 Methodology and related work

In this study data looks at data from three sources and an attempt is made to predict the gust factor in a given place in Iceland. Reanalysis data, along with elevation data is used to predict the gust factor. This study looks at data at the point of interest but does not look at the data as a time series. This thesis aims to improve on the baseline model of always predicting the mean gust factor and show that some structure can be learned from reanalysis data about gusts. To do this a neural network was created. Any significant improvement on a base model, that always guesses the mean gust factor, would indicate that the final model has something to contribute.

In 2004, H. Ágústsson and H. Ólafsson[1] looked at the variability of gust factor in complex landscapes. They looked at data from automatic weather stations that measure wind at 10 meters above ground. The data that was studied in 2004 comes from the same source as used in this thesis, but limits itself to a smaller section. They only looked at the years 1999-2001. They looked at three factors and how these three parameters effected the gust factor. These were d_m, D, H, that is direction of wind blowing off a mountain, distance to the mountain and the height of the mountain above the weather station. Their main results were that the gust factor is inversely correlated to the distance from a mountain and correlated to the height of the mountain. Ágústsson and Ólafsson (2004) looked at the effect of a dominant point upwind. It did not look at the effects of the landscape more broadly. In this study, landscape upwind is looked at.

1.2.1 Neural networks

To be able to capture the patterns in the data a neural network was constructed. A NN architechture was chosen as they are known to be able to capture patterns well in complex data and handle high parameter counts. This comes in handy when training on different types of data. It is also easy to construct different types of neural networks and see how they fit well with parts of the dataset. A NN uses a lot of matrix calculations to weight input parameters and predict an output. A NN has some number of layers, a deep neural network (DNN) has an input layer, output layer and some number of hidden layers. Hidden layers are middle layers that take data from input layer and eventually reach the output layer. The input layer has a width equal to the number of parameters and the ouput has width equal to the number

of predictants. Models were created for several number of parameters as a way to gauge the influence of each parameter. Each model has one numerical output variable. Each layer also has an activation function. There are several popular activation functions such as Rectified Linear Unit (ReLU), Exponential Linear Unit (ELU) and Hyperbolic tangetn (tanh). The definitions of ReLU, ELU and tanh can be seen in Equations (1.2), (1.3) and (1.5).

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

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

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

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

1.2.2 Model evaluation

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

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

This measure was chosen because the target is the gust factor (the wind gust over the average wind). If the target would have been the wind gust rather than the gust factor then something like mean absolute error might be more appropriate.

1.2.3 Model explainability

Neural networks are often considered as mysterious black boxes[4]. In an attempt to understand the model predictions, methods designed for explainability 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 prediciton $\hat{f}(X)$ for Shapley values is given by Equation (1.7). In Equation (1.7), x_j is the feature value, with β_j as the weight of that feature and $\beta_j E[X_j]$ is 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 what is the contribution of a given parameter. This means that Shapley values can explain individual predictions. Other machine learning tools, like ELI5 (Explain like I am 5), randomly shuffle a feature and look at the effect on model performance[9].

2 Data gathering and processing

Data was sourced from several streams. The Icelandic Meteorological Office (IMO) provided measurements from weather stations all around Iceland. NWP data was downloaded from Copernicus Arctic Regional Reanalysis dataset (CARRA). A land elevation model was also provided by IMO.

2.1 Automatic Weather Station Data

IMO provided 10 minute measurements from 327 weather stations all around Iceland. The measurements that met the filtering criteria, started in 2004 and ended in 2023. Of these 327 stations, 212 were from IMO and placed at 10 meters above ground, while the rest (115) were from the Icelandic Road and Coastal Administration (IRCA) and placed at 6-7 meters above ground[18]. The location of these weather stations can be seen in Figure 2.1. The information that is provided by these Automatic Weather Stations (AWS) is presented in two different types of data files, hourly and 10 minute files. The hourly files are summations of the 10 minute files, with the exception that errors, such as nails, still in the 10 minute files should have been removed from the hourly documents. Nails, are sharp increases from the rest of the data and are unrealistic outliers that are considered measurement errors and are discarded. Each type of document contain the following information: the date and time, the station number (that can be converted to the coordinates using another data set), the average wind speed (f), the wind gust (f_g), the standard deviation of the wind gust, the direction of the wind (d) and the standard deviation for the wind direction. These measurement started at the end of the 20th century, when the first AWS stations was installed. More have been added in the following decades. In this thesis the data is not considered as a time series, instead the predictions are made using only data at a given time.

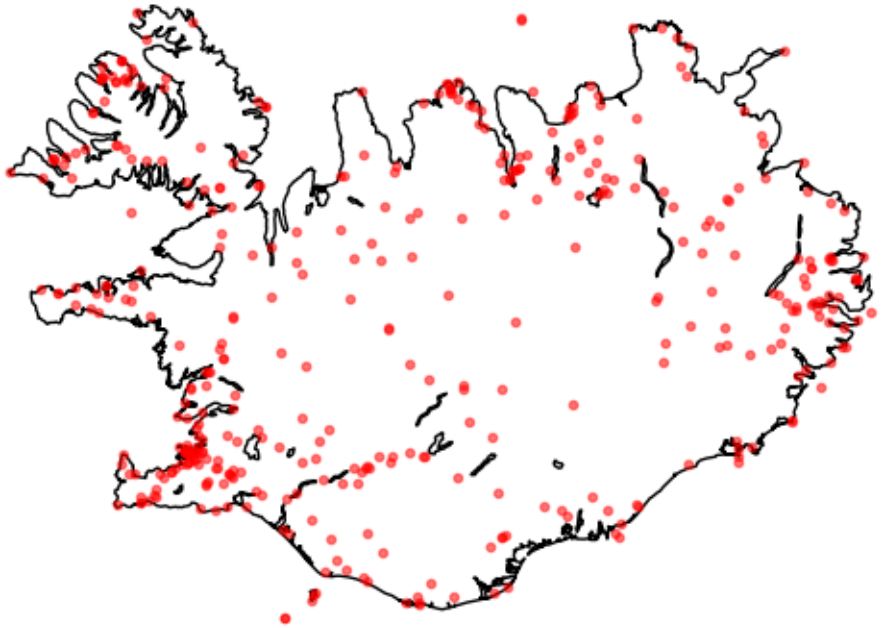

Figure 2.1: Locations of all 412 stations that were looked at in this study. Most of these were from IMO but over a hundred were from IRCA. IMO AWS are placed at 10 meters above ground, while IRCA stations are placed at around 6-7 meters above ground.

2.2 CARRA Data

The CARRA dataset goes back to September 1991 and is currently updated monthly, with a latency of 2-3 months[5]. The oldest IMO data point that fulfills given criteria of consistent available datapoints is from 2004. This is covered by CARRA. The CARRA dataset is available for two regions, West and East. These regions cover a large part of the North Atlantic, including Greenland and Scandinavia. Each of these covers a vastly larger area than the area of interest. This leads to having to store a large amount of data. To get the data one has two options. CARRA web interface or using the API client provided by CARRA. Using the API client is the only realistic option here, as there were 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.

The requests to the API are made at each available CARRA hour ([00, 03, 06, 09, 12, 15, 18, 21]) for each available observation. Only these hours can be used as the CARRA predicted output is represents the wind speed at those given hours and would thus not encapsulate well the extreme values in between these times. Using these datapoints, interpolation was used to get an estimation for the point of the given weather station. The CARRA data contains several types of layers. These are single levels, model levels, height levels, pressure levels. The data for this thesis was

downloaded from height levels. That is, data was 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. Each of these features needed to be interpolated to create data for model to be trained on, as the CARRA grid points do not coincide with weather stations.

2.3 Elevation data

IMO provided a TIFF file containing the elevation of Iceland on a 20 meter by 20 meter grid. This file encompasses Iceland and is around 685 MB. The Python package Rasterio allowed for quick lookup with it's index and the affine transform. Using this package it is possible to quickly look up elevation given coordinates using matrix calculations. This package allowed for lookup by coordinates and by positioning inside the grid encompassing Iceland. It also allows for index lookup using coordinates. Using index lookup, neighboring points can be looked up and used to bridge the elevation for exact posisiton of given point.

2.4 Combining data sources

This project used three main data sources, which need to be queried, filtered and combined to prepare the data for use in the models. When working with hundred of thousands of rows, the efficiency of the code is very important. Iterating through those rows might be necessary at times but will increase the time exponentially as compared to using vectorizing methods were possible. The three data sources were all in a different format. Measurement data from IMO was in text files, elevation data was in GeoTiff and reanalysis data from CARRA was in a GRIB format. To use the data to train, these three data sources needed to be combined into one file. This was done based on the measurement data from the IMO. A limit was set on the average wind speed and it was used to select measurement points. Along with the average wind speed having to be above a certain limit, to ensure that the same weather for the same location is not duplicated. The data from IMO was supplied for 10 minute increments, while CARRA data is in 3 hour intervals. This means that to use the CARRA data to predict the measured values from IMO, temporal interpolation would need to be done. Along with the temporal interpolation, note that the CARRA data is given in a rectangular grid where the distance between each point is around 2.5 km while the the information from the IMO is given at specific locations. The elevation information was given by a 20 by 20 rectangular grid that covers Iceland. When combining these data sources an interpolation method

needs to be decided upon. Here linear interpolation was applied, both temporally and spatially. As the measurement points chosen were not outliers, that is the strongest average wind speed in a given range, interpolating in time does not work well. This necessitates using only measurements that are exactly at three hour intervals (00, 03, 06, 09, 12, 15, 18, 21 hours) at interpolating only in space. The choice of interpolation method, although potentially impactful on the results, was not specifically addressed in this study.

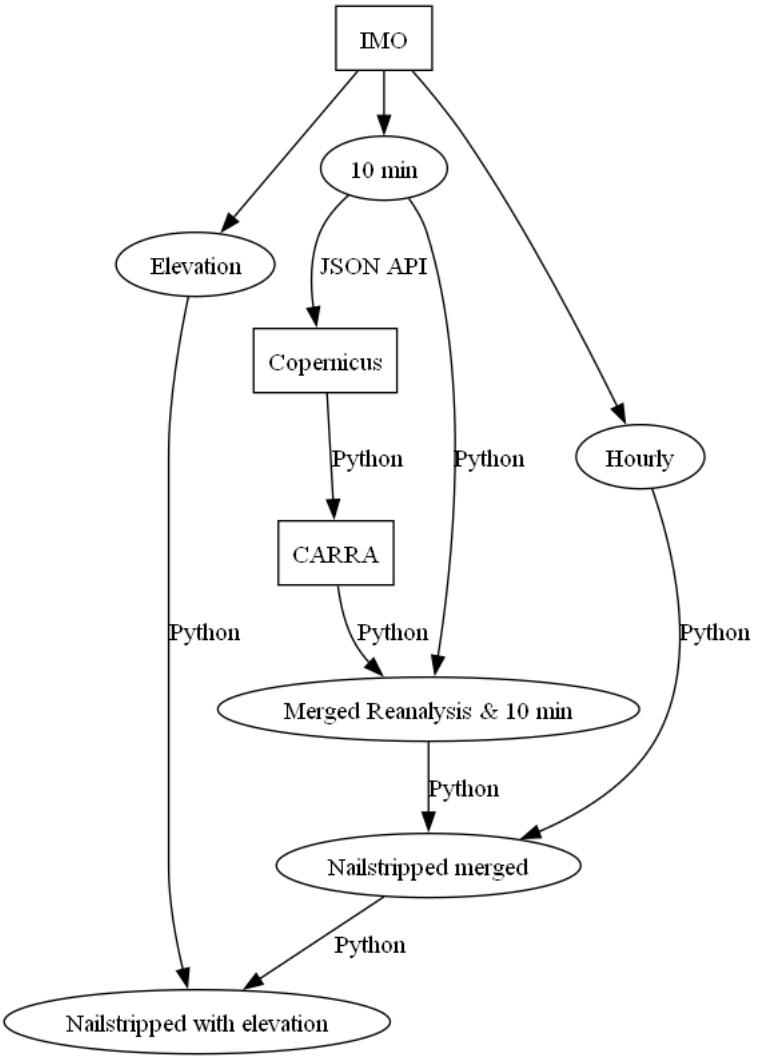

Figure 2.2: A flow chart showing how data sources were combined

The procedure of combining these sources was as follows and can be seen in Figure 2.2. The measured data from the AWS is filtered by using a limit on the average wind speed. The gust factor generally drops with increased wind speed (although not always dependent on factors such as the landscape [20]). Even so being able to predict the gust factor is more important for higher average wind speed due to higher wind gusts. After this stripped dataset over every AWS has been created it is

used to query the CARRA data by using their API. The CARRA API needs to be queried for given hours, days, months, years and a given area. That is, if queried for a given hour, it returns that hour for every day that is queried. Similarly if queried for a given day, it returns that day for every month. In light of these restraints, it was decided to query month by month. Querying only the days needed but every hour of the day (UTC 00, 03, 06, 09, 12, 15, 18 and 21). After querying and downloading the data for the height levels and variables requested, points of interest are interpolated and values stored in a pandas dataframe. After this the downloaded data is discarded and the next month is queried. This drastically decreases the amount of data that needs to be stored as compared to downloading the entire area and keeping all the data points (a reduction from several terabytes to less than a gigabyte).

Once CARRA data has been merged with AWS data, using station and time columns, then this combined file needs to checked for nails. This is done by using the hourly data (which is supposedly error free). As the data has been filtered in for the highest average wind speed in a 48 hour interval, the hourly data can be used to find nails. The hourly data is combined with the merged AWS and 10 min data. Then filtering is applied on the average wind. If the average wind speed for the 10 minute data is higher than the hourly the rows are dropped. Most stations have nails in less that 10% measurements. The stations that have higher than 10% error are ignored.

The elevation data comes in a GeoTIFF file that covers Iceland. It is a rectangular grid of resolution 20 meters. For every point of interest (every weather station), the elevation of that given point along with other points surrounding the weather station is retrieved. For each point retrieved interpolation needs to be done. This is done in a similar manner to the interpolation of the CARRA data. The four points bounding the point of interest were used to linearly interpolate the value of the point of interest. This information is included in the training data as the landscape is known to influence both the average wind and the gustiness [20].

The error in reanalysis wind speed and measured wind speed can be significant. The absolute error increases as the measured wind speed increases, while the percentage wind speed decreases. A grouping of these errors by wind speed can be seen in Table 2.1

Another thing to look at is the distribution of error by station, both in terms of their coordinates and number of measurements. Looking at Figure 2.3, this distribution can be seen.

Table 2.2 shows the 5 best and worst stations in terms of MAE.

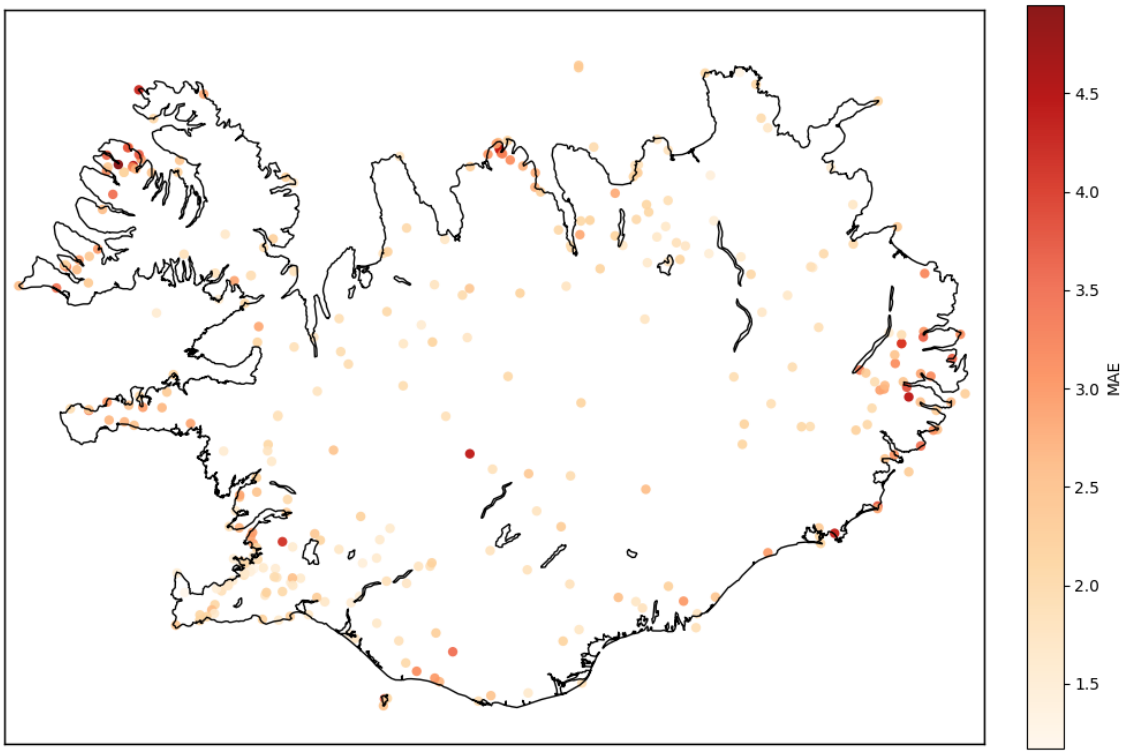

Figure 2.3: The distribution of mean absolute errors by station. Using mean absolute error instead of mean absolute percentage error allows for all points to be used. Mean absolute percentage error can only be used if 0 values are ignored.

Table 2.1: Comparison of measured and reanalysis wind speed using mean absolute error (MAE) and mean absolute percentage error (MAPE). Note that for the computation of MAPE for ranges that otherwise include 0, 0 values have been excluded so as to prevent division by zero and exploding values. The comparisons are done using measured wind speed (at 10 meters above ground for IMO and 6-7 meters above ground for IRCA) and reanalysis wind speed at 15 meters above ground.

f	n	MAE
$[0; 5[$	6.2e6	2.1
$[5; 10[$	4.2e6	2.2
$[10; 15[$	1.5e6	2.5
$[15; 20[$	3.9e5	3.0
$[20; 25[$	8.4e4	4.0
$[25; \infty[$	2.0e4	6.6
$[0; \infty[$	1.2e7	2.2

Table 2.2: Mean absolute error for reanalysis wind speed as compared to measured wind speed, for the five stations with the highest difference and the five stations with the lowest difference.

Station	Number of measurements	MAE	Location
1470	6.8e3	1.17	Reykjavík Háahlíð
1350	5.2e4	1.18	Keflavíkurflugvöllur
1482	1.4e4	1.23	Reykjavík Víðidalur
4921	1.3e4	1.29	Rif á Melrakkasléttu
1477	5.6e4	1.29	Reykjavíkurflugvöllur
35553	4.0e3	4.30	Almannaskarð - göng
6745	1.5e4	4.36	Kerlingarfjöll - Ásgarðsfjall
35978	7.9e3	4.40	Fáskrúðsfjarðargöng suður
2640	1.6e3	4.51	Seljalandsdalur
32635	3.2e4	4.95	Botn í Súgandafirði

2.5 Data Structure

Once data has been retrieved for all three sources and processed, including interpolating values, it needs to be made ready to use by the model, for both training, validation and test. The starting point is a dataframe that contains measured information from AWS. This includes the average wind, the wind gust, wind direction along with the station number and coordinates. When selecting the CARRA data certain height levels are chosen. These present as separate lines in the CARRA dataframe. Information for one observation is represented in as many lines as height levels requested in the reanalysis data. These rows need to be combined on the position (the weather station). When this is done it is possible to combine the AWS IMO data and CARRA reanalysis data on the location and time columns. The last data source is the elevation. A circle sector upwind is looked at. In any case the points, that represent these sections, were selected as shown in Code Listing 2.1. A range of angles are defined based on the wind direction d at some distance from the given point. This means that the resultant points (equal in number to the length of angleRange by k) from arcs at several distances from the given weather station.

Listing 2.1: Sector elevation points generated

```
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 result is a dataframe that has measured data from AWS, which gives us our target, reanalysis data from CARRA, which gives us weather variables to train on, and finally elevation points in the landscape to include in our training data. An example of what the data looks like can be seen in Table 2.3.

Table 2.3: An example of data structure used to train model. Data points include the derived variables Richardson number (Ri) and Brunt-Väisäla frequency (N) (defined below), the elevation of the station, direction of wind and relative direction of the wind (twd, that is the direction of the wind relative to center of Iceland), along with some combination of wind speed, pressure and temperature at the different height levels. Finally there are the elevation points around a given station, where the elevation is relative to the station.

Ri	N^2	station elevation	twd	ws_{15}	wd_{15}	t_{15}	p_{15}	$elevation_0$	...
-1.18e+00	2.67e+04	100	1.5	10	5	0	100	2	...

Looking at Table 2.3 note that the first two columns represent two variables that describe the stability of the air. These are the Richardson number (Ri)[24] and

Brunt–Väisälä frequency (N)[8], and are calculated using Equations (2.1) and (2.2)[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 the acceleration due to gravity, dT is the temperature difference between the two height levels, dz is the elevation difference, T_{ave} is the average temperature (that is the average of the two temperatures in the height levels) and dU is the wind speed difference between the two height levels. Both of these numbers provide some insight about the stability of the air. A lower value for the Richardson number indicates a higher turbulence. A typical range of values could be between 0.1 and 10, with values below 1 indicating significant turbulence[24]. When the square of the Brunt-Väisälä frequency is negative, then the air is unstable (an air parcel will move away from its original position)[8]. These are derived factors from the reanalysis data and as such there shouldn't be a significant information gain using Ri and N as opposed to having the raw data. However, including these factors instead of every reanalysis variable requested might speed up training as well as making the model more easily explainable with the use of Shapley values or other tools for explainability. Using Shapley a feature importance value is attributed to a given feature by creating all possible permutations of any possible length (up to number of features) and seeing how the predictions are skewed when the given parameter is included or excluded. This needs to be done for all parameters. The time complexity of this is very high (2^n coalitions)[15]. Most implementations use some approximations, which still can take a considerable amount of time for models with a high parameter count and many examples. The Richardson number includes the difference in wind speeds in the denominator. In certain cases, where the difference in wind speed between two levels is very, can blow up to infinity. This can cause problems and distort the predicitons.

2.6 Data distribution

The CARRA data is reanalysis and as such might have a bias or some systemic distortion when compared to the measured data. The distribution of the observed

and reanalysis wind speed can be seen in Figure 2.4. Looking at the figure, the reanalysis wind tends to be higher even though the distribution is similar.

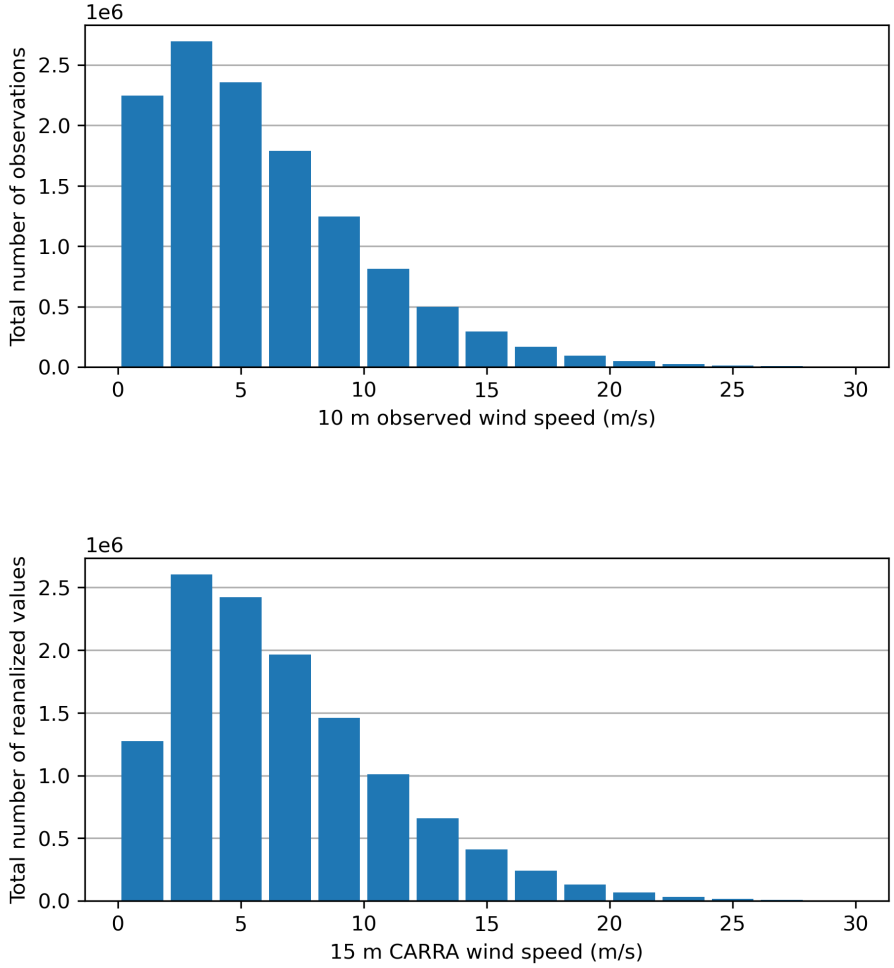

Figure 2.4: Upper figure shows a histogram of observed wind speeds provided by IMO and IRCA for all used weather stations. Lower figure shows the interpolated CARRA reanalysis values at weather stations. The reanalysis data is only available at 3 hour intervals and 2.5km grid. As such the observed values are only chosen at these specific times (00, 03, 06, ..., 21). Spatial interpolation is needed and is linearly weighted based on distance of CARRA grid points from stations.

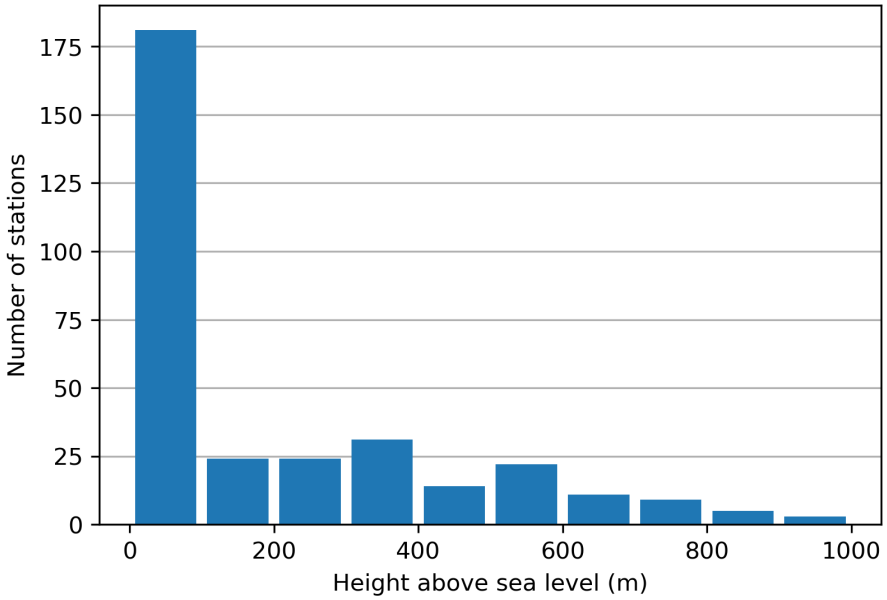

Figure 2.5: Distribution of heights of weather stations above sea level. One station has been excluded as an outlier of well above 1000 meters, having few and inconsistent datapoints.

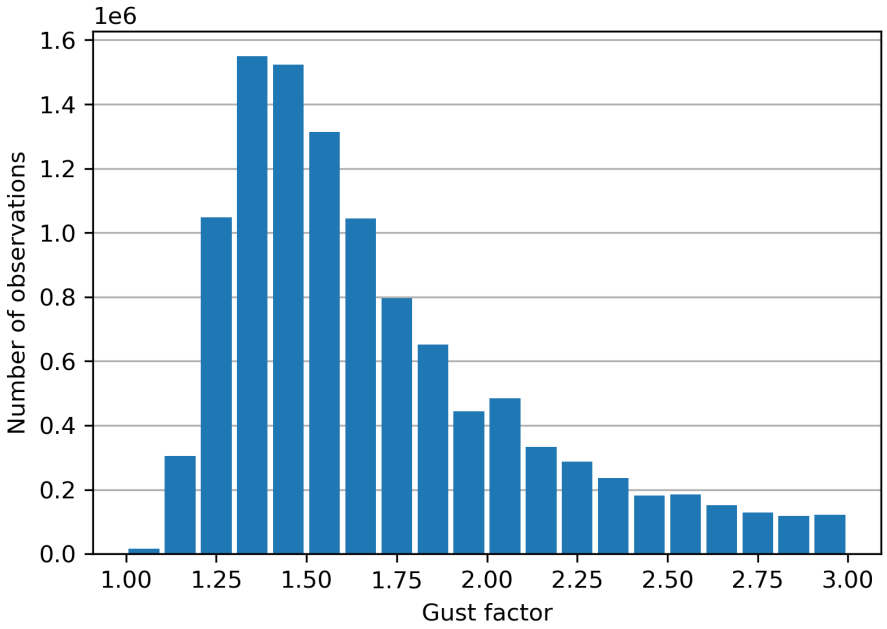

Figure 2.6: Histogram of gust factors. By definition the lower bound of gust factor is 1. The majority of observed gust factors fall in the range 1.2 to 2. Gust factor decreases with increasing wind speed.

3 Model Architecture and Training

3.1 Model structure

The structure of the neural network is such that it contains some number n of fully connected layers and batch normalization for each layer, along with regularization. All of these layers have the same number of units. The last layer has a dropout of 50%. In addition to these layers there is one more output layer. This is simply a dense layer with 1 unit. A grid search was performed to determine the hyperparameters that minimize the loss. Hyperparameters are parameters that are set before the training begins[6]. These hyperparameters include number of units in each layer, number of epochs to train for, number of layers, batch size, optimizer and penalty to enforce in the regularization. The possible combinations tried can be seen in Table 3.1.

Table 3.1: Hyperparamter search with best performing combination shown. Hyperparameter search was done using hyperband algorithm that initially searches randomly for the best parameters but then hones in on what is working and as such is neither exhaustive nor completely random. This means that a hard upper limit will not be set on the number of combinations to try like with randomsearch.

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 gridsearch. 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.

3.2 Model Training

To determine the required number of epochs a high number of epochs can be selected and the loss plotted. These plots can be seen in Figures 3.1 and 3.2. Looking at these two plots, training loss decreases for over 1900 epochs in both plots, while the validation loss stops decreasing at around 200 epochs for DEM model and 1800 epochs for model without DEM. This is unexpected. The DEM model has vastly more parameters and as such should contain more useful information to be learned, which one might expect to take longer to learn. This does not seem to be the case. The DEM model only gets trivially better after 1000 epochs while the non DEM model keeps meaningfully improving until around 1800 epochs. With respect to time constraints of shared resources models were trained for 250 epochs.

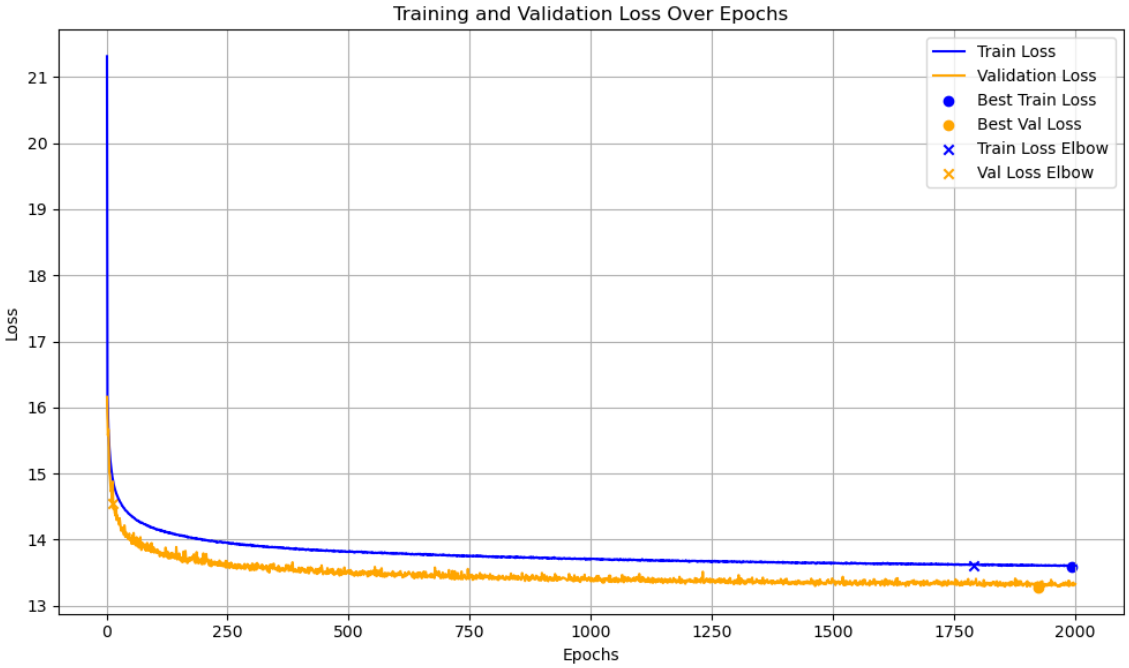

Figure 3.1: Loss plot of model trained for 2000 without DEM.

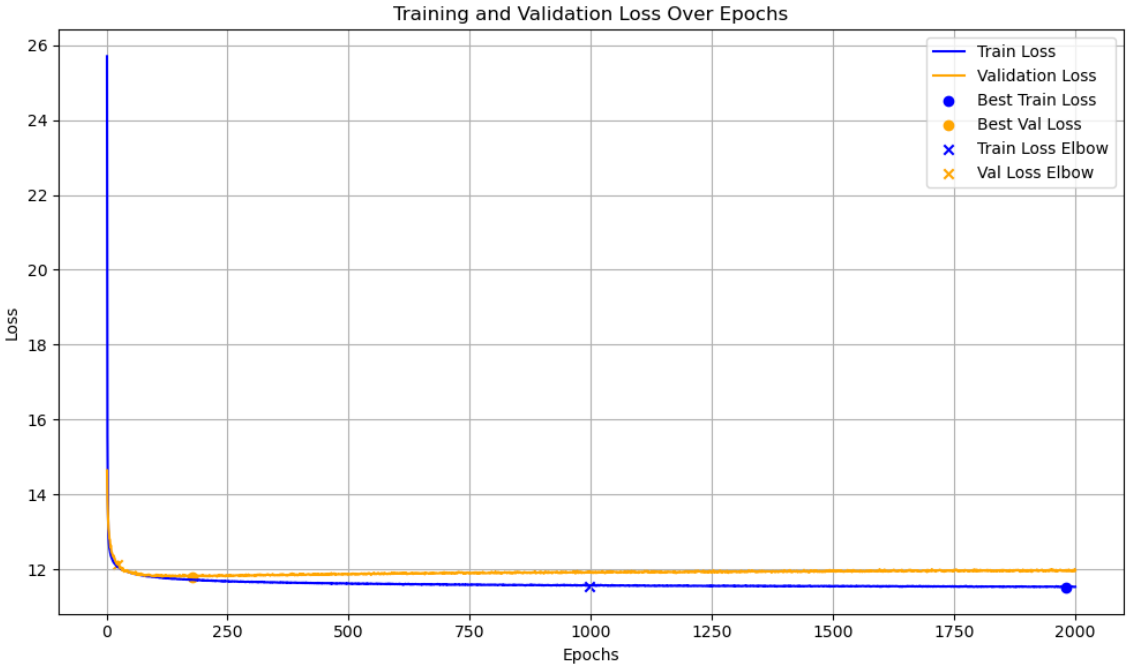

Figure 3.2: Loss plot of model trained for 2000 with DEM.

4 Results

4.1 Results

A baseline model was constructed. This model looked at the gust factor for some training data and took the average of this and predicted this average everytime. Several baseline guesses were created based on a lower limit assigned to the average wind speed limit (AWSL). As the average wind speed increases, then the variability in gust as a percentage decreases[1]. This means looking at a subset of the data where the AWSL is higher, a better result can be expected. The results show this. Several different AWSL were used for the baseline model, as can be seen in table 4.1. This sets a goal. A model that does not significantly improve on this baseline suggests either failure to capture essential patterns in the data or that the data itself may lack the necessary information for substantial improvements upon the baseline. Using the previously described neural network architecture setups for each AWSL, with and without landscape elevation information, MAPE was determined. The results can be seen in Table 4.1.

Table 4.1: MAPE for each average wind speed limit with and without landscape elevation in a 30° sector around the point of interest into the direction of the reanalysis wind. The influence of adding elevation data seems to reduce the error. The percentage error is higher for lower wind speeds and thus observing the error for different lower bound of wind speed will produce different results. This lower bound is determined using the reanalysis wind speed at 15 meters (ws_{15}).

AWSL	**MAPE**		
$[m/s]$	*Baseline*	*Without DEM*	*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%

This is some improvement upon the baseline error, with a decrease in error from 23.9% to 13.6% and 10.6% for the baseline, model without DEM and with model

with DEM for 10 m/s cutoff. The power generated by wind mill increases with wind speed cubed[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. Another way to look at the error improvement is by station. No location data was directly included in the training data. In Table 2.2 the mean absolute error of predicted average wind speed and measured average wind speed can be seen for the extreme values. A question to ask is it possible to achieve better results when only looking at a single station?

Table 4.2: The MAPE results for selected 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 AWSL 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	**# points**	**MAPE**	
		General training	Site training
Akrafjall	42,791	18.6%	93.7%
Almannaskarð	4,014	12.2%	86.7%
Ásgarðsfjall	15,121	9.1%	9.4%
Jökulheimar	17,176	7.7%	7.7%
Sandbúðir	18,718	6.8%	6.4%
Stórholt	35,126	7.1%	29.2%
Þúfuver	19,538	6.4%	6.8%

Instead of looking at the exact values of MAPE at select stations, a plot of the error distribution can be created. This can be seen in Figure (4.1). Looking at Figure (4.1), the worst performing stations can be seen at Vestfirðir and around the coastline. Stations further inland seem to have lower error. The worst performing station, at Seljalandsdalur, is at Vestfirðir while the best performing station Garðskagaviti is at the South-Western tip of Iceland in Reykjanes.

Table (4.3) shows no improvement over the values in the last column in Table (4.1). 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 (4.4).

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

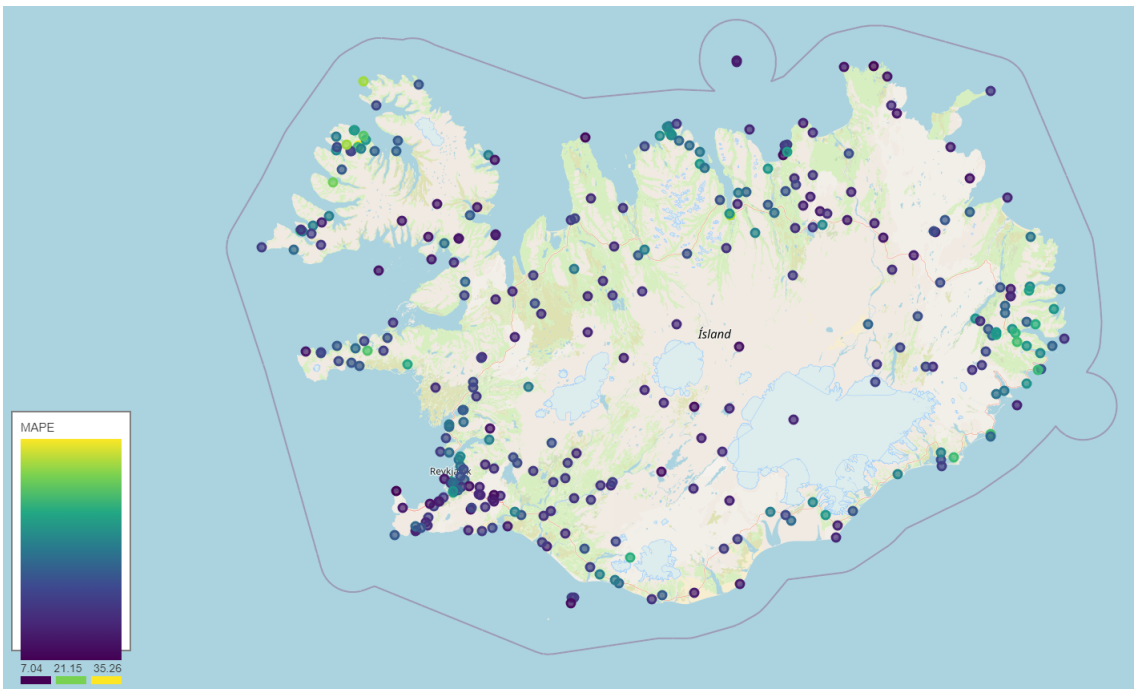

Figure 4.1: The MAPE error of each station in data shown as color gradient circles. That is each station is represented by one circle, with the error value represented as a color gradient from dark blue to yellow. The lowest error at a single station was around 7% at Garðskagaviti and the highest around 35% at Seljalandsdalur. It is important to note that the model is trained using a cutoff of 10 m/s and this cutoff point is determined using the reanalysis wind at 15 meters above ground.

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 vas 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

Table 4.3: The MAPE results for different AWSL intervals. Here instead of training for all data above a certain threshold put of the wind, 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. This results should thus be somewhat comparable to the last column in Table (4.1)

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%

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 independece[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 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

Table 4.4: 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 AWSL 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%

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 4.5.

Table 4.5: Showing the results of different models. Models defined by different sets of parameters, with increasing complexity. This table shows the feature importance as the loss decreases as more variables are added. Adding all height levels decreases the loss but very little.

Model parameters	**MAPE**
Baseline constant	23.9%
ws_{15}	17.2%
$[ws_{15}, t_{15}, p_{15}, wd_{15}]$	16.7%
$[ws_{15}, t_{15}, p_{15}, wd_{15}, ASL, twd]$	13.8%
$[ws_{15}, t_{15}, p_{15}, wd_{15}, ASL, twd, N, Ri]$	13.7%
$[ws_{15}, t_{15}, p_{15}, wd_{15}, ASL, twd, N, Ri]$ + DEM	12.0%
$[ws_{15,250,500}, t_{15,250,500}, p_{15,250,500}, wd_{15,250,500}, ASL, twd, N, Ri]$	12.0%

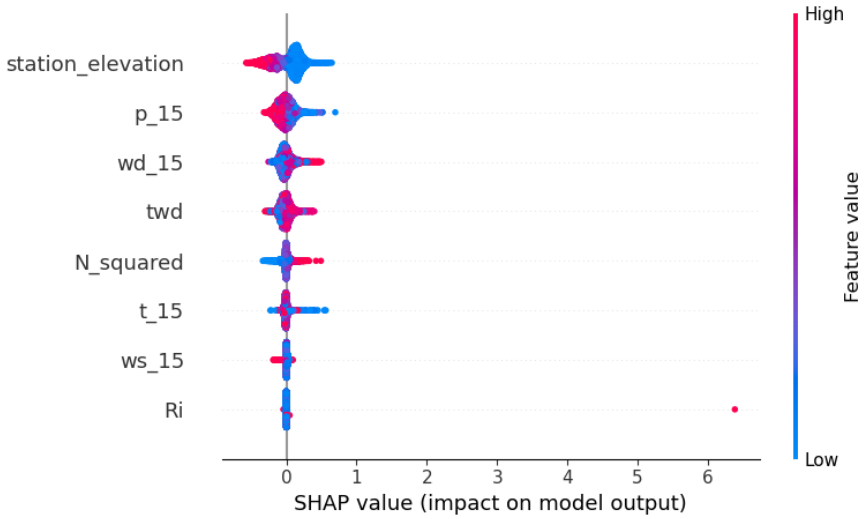

Figure 4.2: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. We can see that generally multiple factors influence the prediction, with the station elevation being highly influential. There is seemingly one outlier for the Richardson 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 4.1.

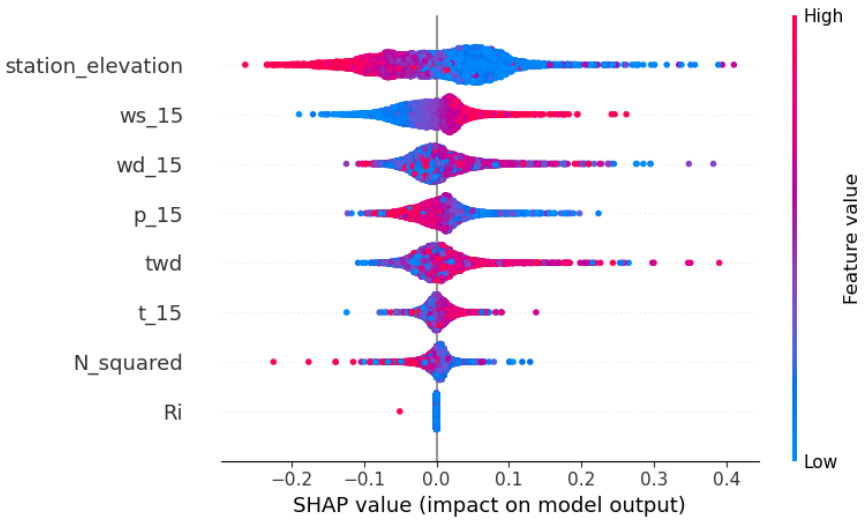

Figure 4.3: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. Generally multiple factors influence the prediction, with the station elevation being highly influential. Elevation data is excluded when working with Shapley values, 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

The goal of this tesis was to research whether it was possible to use reanalysis data to increase the predictability of wind gusts and see what influence including the landscape elevation would have. A quick statistical analysis will give a p-value of 0, that is these results are statistically significant. The improvement shown in Table (4.1) is not chance. There is some pattern to be learned in the interval that leads to a higher predictability of wind gusts over baseline. There is also a large variability in the predictability based on weather stations. A large part of that can be attributed to the differing amount of data points for each station. It would be interesting to know how much of the predictability difference can be explained away with more data for these stations that have few data points or if there is some inherent difference. If there is inherent less predictability at those stations, what could be contributing factors in that?

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

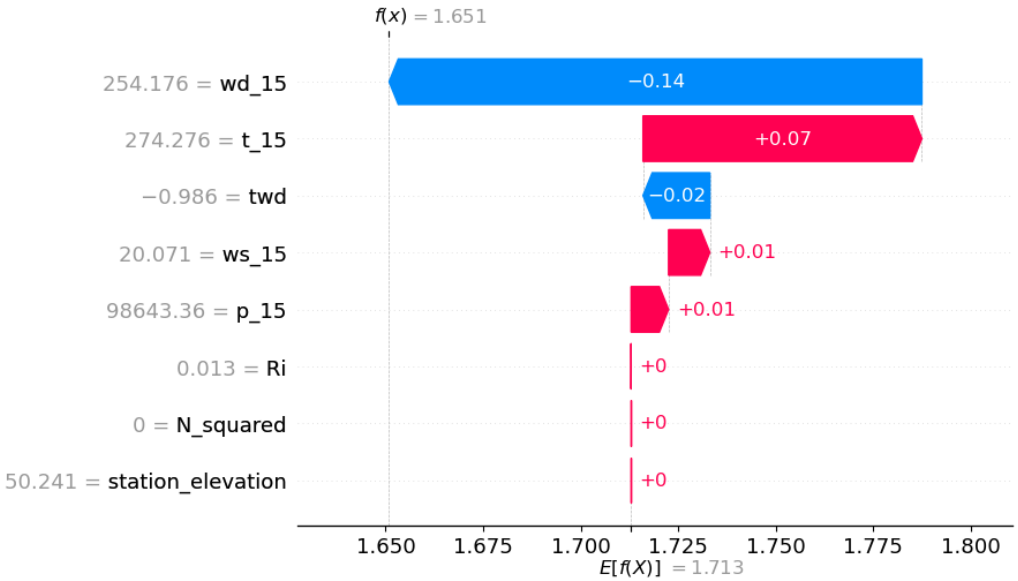

Figure A.1: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. In this specific instance the wind direction (wd_{15}) has the highest negative influence and the temperature has the highest positive influence. Elevation data is 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 4.1.

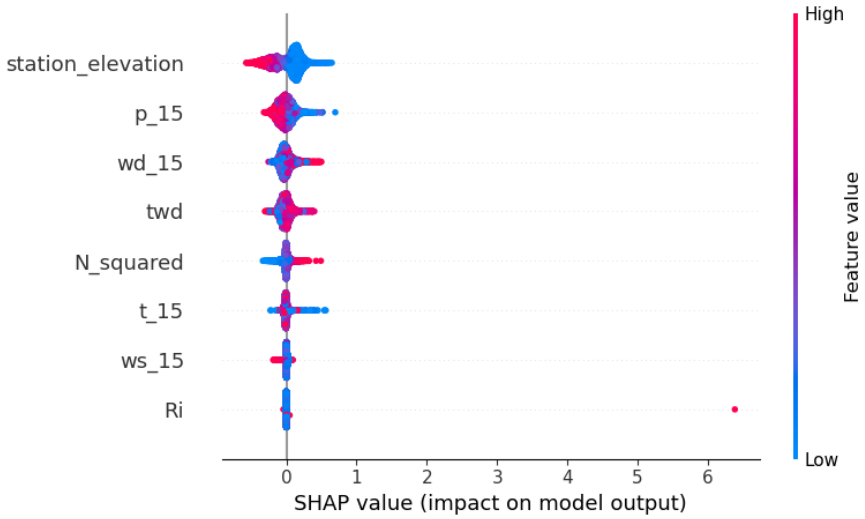

Figure A.2: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. We can see that generally multiple factors influence the prediction, with the station elevation being highly influential. There is seemingly one outlier for the Richardson 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 4.1.

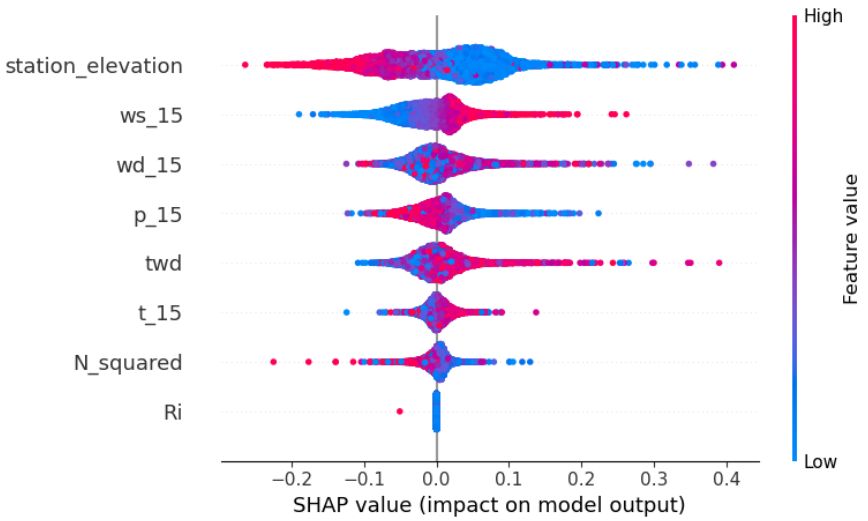

Figure A.3: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. Generally multiple factors influence the prediction, with the station elevation being highly influential. Elevation data is excluded when working with Shapley values, 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.

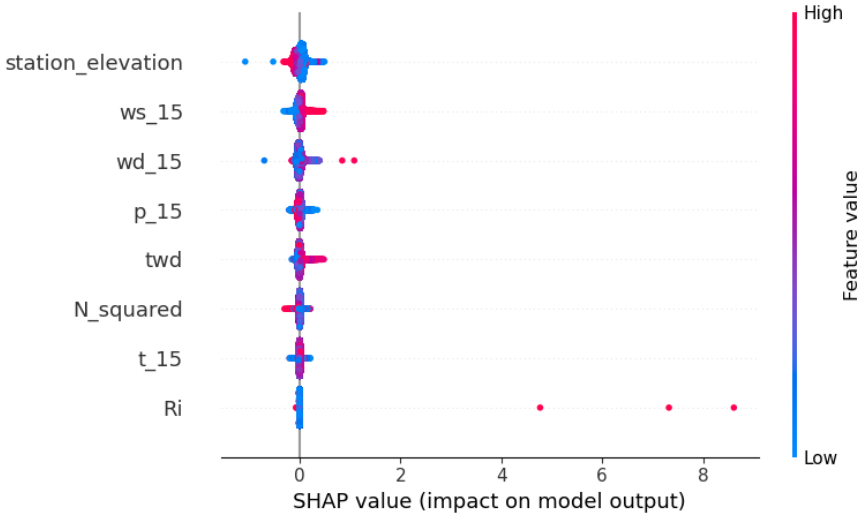

Figure A.4: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. The distribution seems to be the same as before, discounting the outliers.

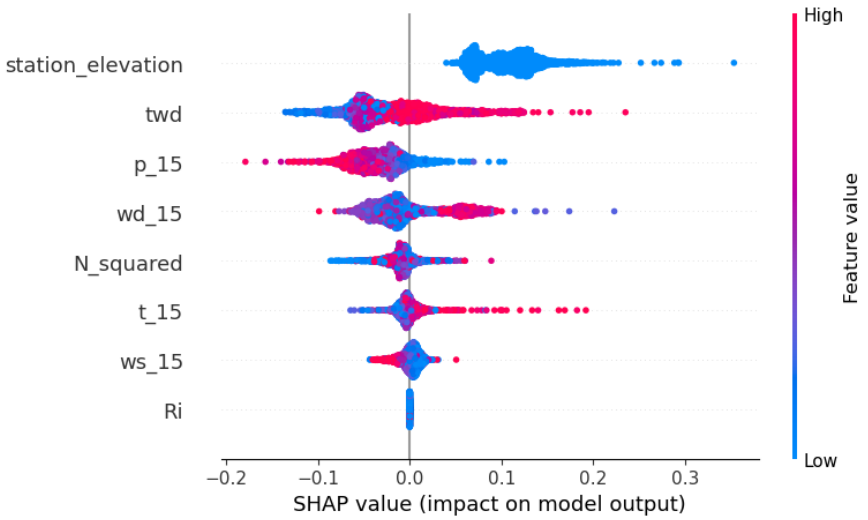

Figure A.5: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. This plot only looks at datapoints from Akrafjall. This seems to show the same distribution as previous summary plots. Station elevation is influential and Richardson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

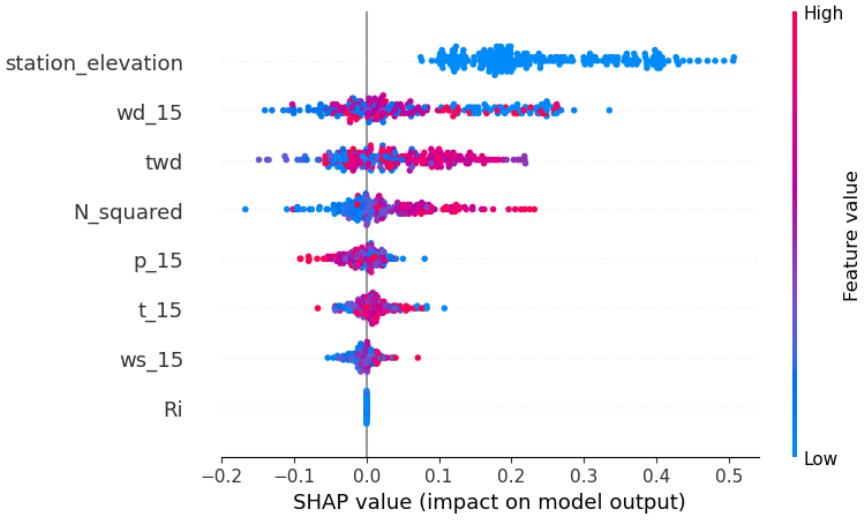

Figure A.6: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. This plot only looks at datapoints from Almannaskarð. This seems to show the same distribution as previous summary plots. Station elevation is influential and Richardson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

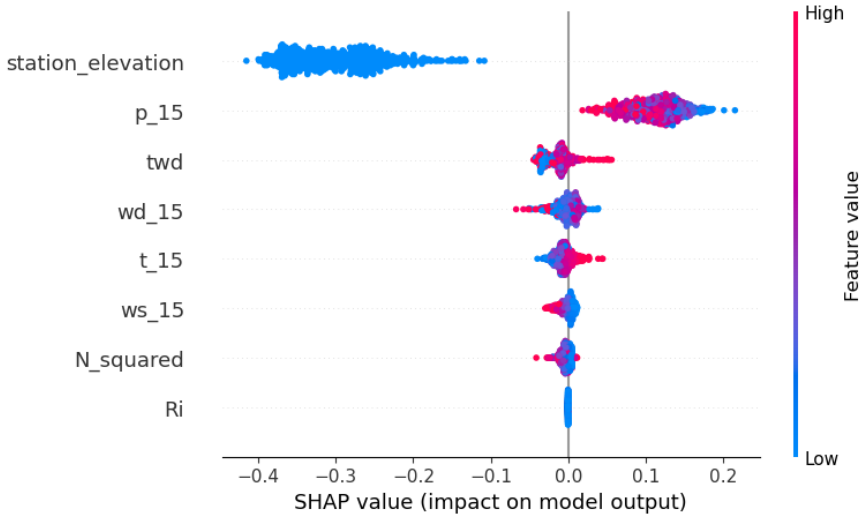

Figure A.7: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. This plot only looks at datapoints from Ásgarðsfjall. This seems to show the same distribution as previous summary plots. Station elevation is influential and Richardson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

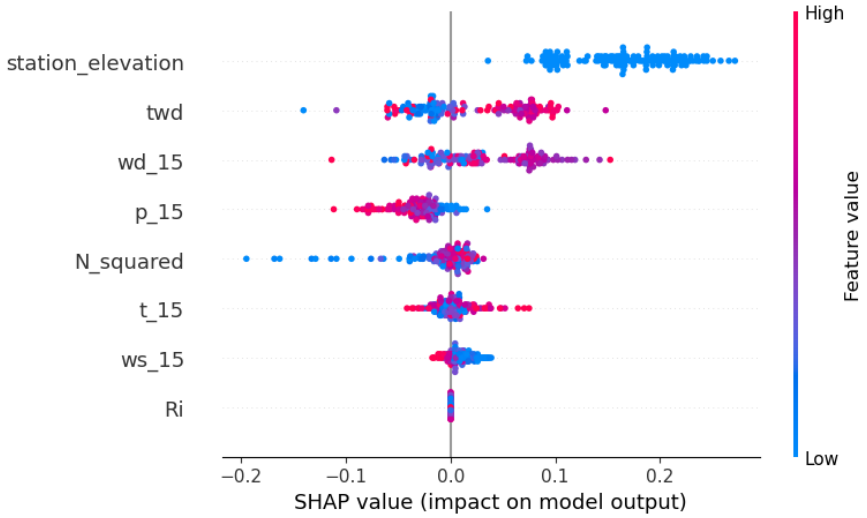

Figure A.8: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. This plot only looks at datapoints from Háahlíð. This seems to show the same distribution as previous summary plots. Station elevation is influential and Richardson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

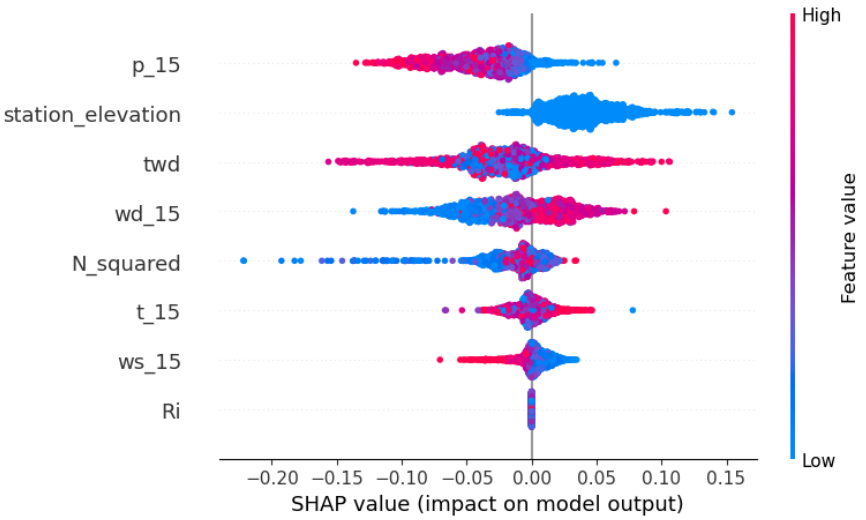

Figure A.9: Feature importance of a neural network with model architecture as described in Table 3.1 and data as described in Table 2.3. This plot only looks at datapoints from Keflavíkurflugvöllur. This seems to show the same distribution as previous summary plots. Station elevation is influential and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