MASTERARBEIT | MASTER'S THESIS

Titel | Title

The influence of meteorological parameters on wind speed extreme events in wind turbine and era5 data: Causal inference by heterogeneous Granger causality and its statistical validation

verfasst von | submitted by

Marin Seršić

angestrebter akademischer Grad | in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien | Vienna, 2024

Studienkennzahl lt. Studienblatt | Degree programme code as it appears on the student record sheet:	UA 066 921
Studienrichtung lt. Studienblatt | Degree programme as it appears on the student record sheet:	Masterstudium Informatik
Betreut von | Supervisor:	Univ.-Prof. Dipl.-Inform.Univ. Dr. Claudia Plant
Mitbetreut von | Co-Supervisor:	RNDr. CSc. Katerina Schindlerova

Acknowledgements

I would like to thank my supervisors, Prof. Claudia Plant and Dr. Katerina Schindlerova for help with this thesis. I would also like to thank Andreas Fuchs for explaining his process with the storm data.

Abstract

This thesis investigates the causal effects of meteorological parameters on the extreme values occurring in wind speed in two datasets: ERA5 data from the Andau wind park in Austria during select European wind storms, and data retrieved from the Andau park wind turbines in the period 2000.-2020. It uses Granger causal inference, namely the heterogeneous graphical Granger model (HGGM). HGGM can detect causal relations in time series with any distribution in the exponential family, unlike the classical Granger model for causal inference which supports fewer distributions. For this, a python implementation of HGGM was made based on the original MATLAB implementation. In the ERA5 storm dataset, the analysis reveals that wind direction, potential vorticity, convective available potential energy, and humidity have a causal effect on high wind speed extremes during windstorms. However, a t-test shows that only convective available potential energy and relative humidity are statistically significant. Compared to the results from a presentation using a similar method (HMML), HGGM found fewer causal links. Unlike HMML, it suffered from overestimations requiring the removal of several highly correlated variables. In the turbine data, extremes were split between high and low. To prevent overestimations, the time series duration had to be extended from the initial 96 hours to 200. The analysis found that temperature and (relative) humidity have a statistically significant effect on extremely high wind speeds. Boundary layer height, temperature and wind speed have a statistically significant effect on extremely low wind speeds. Overall, HGGM found several causal links between various meteorological variables and wind speed extremes in both datasets. However, HMML found more causal links in the ERA5 dataset and it did not suffer from issues with overestimations. The thesis finished with some suggestions for further research.

Kurzfassung

In dieser Arbeit werden die kausalen Auswirkungen von meteorologischen Parametern auf die auftretenden Extremwerte der Windgeschwindigkeit in zwei Datensätzen untersucht: ERA5-Daten aus dem Windpark Andau in Österreich während ausgewählter europäischer Windstürme und Daten, die aus den Windkraftanlagen des Windparks Andau im Zeitraum 2000.-2020. gewonnen wurden. Es wird die Granger Kausalinferenz verwendet, nämlich das heterogene grafische Granger-Modell (HGGM). Das HGGM kann kausale Beziehungen in Zeitreihen mit beliebigen Verteilungen aus der Exponentialfamilie aufdecken, im Gegensatz zum klassischen Granger-Modell für Kausalinferenz, das weniger Verteilungen unterstützt. Zu diesem Zweck wurde eine Python-Implementierung des HGGM auf der Grundlage der ursprünglichen MATLAB-Implementierung erstellt. Die Analyse des ERA5-Sturmdatensatzes zeigt, dass Windrichtung, potenzielle Vorticity, konvektive verfügbare potenzielle Energie und Feuchtigkeit einen kausalen Einfluss auf hohe Windgeschwindigkeitsextreme während Stürmen haben. Ein t-Test zeigt jedoch, dass nur die konvektiv verfügbare potenzielle Energie und die relative Luftfeuchtigkeit statistisch signifikant sind. Im Vergleich zu den Ergebnissen einer Präsentation, bei der eine ähnliche Methode (HMML) verwendet wurde, fand HGGM weniger kausale Zusammenhänge. Im Gegensatz zu HMML litt es unter Überschätzungen, die die Entfernung mehrerer stark korrelierter Variablen erforderten. In den Turbinendaten waren die Extrema zwischen hoch und niedrig aufgeteilt. Um Überschätzungen zu vermeiden, musste die Dauer der Zeitreihe von ursprünglich 96 auf 200 Stunden verlängert werden. Die Analyse ergab, dass Temperatur und (relative) Luftfeuchtigkeit einen statistisch signifikanten Einfluss auf extrem hohe Windgeschwindigkeiten haben. Grenzschichthöhe, Temperatur und Windgeschwindigkeit haben einen statistisch signifikanten Einfluss auf extrem niedrige Windgeschwindigkeiten. Insgesamt fand HGGM mehrere kausale Zusammenhänge zwischen verschiedenen meteorologischen Variablen und extremen Windgeschwindigkeiten in beiden Datensätzen. HMML fand jedoch mehr kausale Zusammenhänge im ERA5-Datensatz, und es gab keine Probleme mit Überschätzungen. Die Arbeit schließt mit einigen Vorschlägen für weitere Forschungsarbeiten.

Contents

List of Tables

List of Figures

1 Introduction

Wind turbines have become increasingly essential in meeting global renewable energy targets. However, their optimal operation depends on accurate predictions of extreme wind speeds. Wind turbines are only specified for certain wind speeds. While slower winds will cause the turbines to stop generating electricity, higher speeds than specified can damage or even destroy the turbine. For that reason, it is important to better predict extreme wind speeds. Finding out the causes of extreme winds can help with the development of better models for wind speed prediction.

To address this issue, this thesis analyses causalities in wind speed data from the Andau Wind Park in Austria. The data is split into extreme and control time series which are analysed using the Heterogeneous Graphical Granger Model (HGGM)[1] to find possible causes of extreme wind speeds. The thesis uses two datasets: era5 data recorded during 12 European wind storms, and meteorological data recorded by the wind farm's turbines between the years 2000. and 2020.

To achieve this, a Python implementation HGGM was developed, which was then compared with a reference Matlab implementation to assess its performance. Additionally, the results obtained using HGGM with the era5 dataset were compared with those produced by another method called Heterogeneous Minimum Message Length (HMML)[2].

The first dataset contains era5 data from the area of the Andau wind park taken during 12 wind storms. The data is the same as the data used in a presentation[3] made for the 2022. European Geosciences Union (EGU) assembly that uses HMML and is analysed in a similar way in order to allow for a comparison of the results. The extreme time series start 72 hours before the storm begins and ends 24 hours after the storm ends. The total duration is 96 hours or more, depending on the length of the storm. The controls are time series of the same length as the accompanying extreme series. They are taken during calm periods shortly before or after the storm.

After the dataset is split into time series, the HGGM betas and beta proportionalities (beta divided by the sum of all betas) are calculated. The statistical significance of the difference between the means of the storm and control beta proportionalities is calculated, as is the difference between the average storm and control beta proportionalities. The statistical significance of the difference in the average wind speed in the storms and controls is calculated using a t-test. Finally, the results from the HGGM experiments are compared with the results from the HMML experiments in the EGU presentation.

The second dataset contains measurements taken from wind turbines in the Andau wind park in the period from 01.2000. to the end of 2020. It starts with the analysis of the 2020. data only to focus on the differences between the readings on different turbines. Wind speeds over 14.8 m/s are classified as high extremes, and wind speeds under 0.2 m/s are classified as low extremes. The cut-offs are selected to be as high/low as possible while still allowing for enough extreme and control time series to be created. To avoid overestimations, the window for the time series starts 150 hours before an extreme and ends 50 hours after. If there are several extremes in a short period, they are combined in a single time series so that the window starts 150 hours before the first extreme and ends 50 hours after the last extreme in the cluster. There is one record every hour so the length of the time series is at least 200 or more, depending on the amount clustered extremes. HGGM is applied on 10 randomly selected time series of each type (high extreme, low extreme, control), and the results are discussed.

Because there is more data in the full dataset, the cut-off for the high extremes is raised to 16 m/s. The full dataset is analysed and HGGM betas and beta proportionalities are calculated from 50 randomly selected time series of each type. The statistical significance of the difference in the average wind speed in the storms and controls is calculated using a t-test and the causalities are discussed.

By analysing wind speed extremes using HGGM and comparing the outcomes from different methods, this thesis aims to improve our understanding of extreme wind speeds and help with the development of better predictive models for future wind energy applications.

The thesis has the following structure: Chapter 2 goes through the technical background of the HGGM and HMML algorithms. Chapter 3 discusses some related papers[4][5][6][7] from Jakob Runge. Chapter 4 describes the python implementation of HGGM that is used for the experiments. Chapter 5 has the experiments with the era5 data during European storms and a comparison with the results in the EGU presentation[3]. Chapter 6 has the experiments with the wind turbine data. Chapter 7 discusses possible future research and summarizes the thesis.

2 Foundations

This chapter is an overview of Granger causality[8] and the Heterogeneous Graphical Granger Model (HGGM)[1].

2.1 Granger Causality

C.W.J. Granger introduced the concept of "Granger causality" in his 1969. paper titled "Investigating Causal Relations by Econometric Models and Cross-spectral Methods"[8]. Given two time series X and Y, X Granger causes Y if predicting values of Y from past values of Y *and* X provides better results than predicting values of Y using only past values of Y.

Despite the name Granger causality does not determine if X causes Y, only if X predicts Y. Since it is debatable if Granger causality determines the true causality between time series in all cases(cf. [9],[10]), the concept is usually referred to as "Granger causality " to distinguish it from other definitions of causality. For time series X and Y it is then said that X "Granger causes" Y.

2.1.1 Mathematical Definition

According to Granger[8], Y causes X if $\sigma^2(X|U) < \sigma^2(X|\overline{U-Y})$, where U is all of the information in the universe, U - Y is all of the information in the universe apart from the series Y, and $\sigma^2(A|B)$ is the variance of the predictive error series $\epsilon_t(A|B) = A_t - P_t(A|B)$ where $P_t(A|B)$ is the optimum predictor of A_t using B_t. In other words, Y causes X if the prediction of X is more accurate with all available information than with all information apart from Y. This is denoted as $Y_t \Rightarrow X_t$.

Since it is not possible to have all the information in the universe and a vast majority of it is not relevant, the information can be restricted to only the series X and Y. Then if $\sigma^2(X|\overline{X})$ is the minimum predictive error variance of X using only past X, and $\sigma^2(X|\overline{X},\overline{Y})$ is the minimum predictive error variance of X using past X and Y, Y causes X if $\sigma^2(X|\overline{X}) > \sigma^2(X|\overline{X},\overline{Y})$.

In the paper, this argument was based on the identifiability of the linear model:

$$A_0 X_t = \sum_{j=1}^{d} A_j X_{t-j} + \epsilon_t \tag{2.1}$$

where $\mathrm{X_t}$ is a vector of p variables $X_t = (x_{1t}, x_{2t}, ..., x_{pt})^T$ at time t, $\mathrm{A_0}$, $\mathrm{A_1}$ ..., $\mathrm{A_d}$ are $p \times p$ lag matrices (coefficients) and d is a finite or infinite lag. ϵ_t is a p-dimensional error term with a diagonal or non-diagonal covariance matrix.

If the matrix A_0 is diagonal, this model shows Granger causalities as nonzero entries in the coefficients.

2.1.2 Vector Autoregressive Models

In the original paper, Granger used a bivariate (i.e. p = 2) version of the Structural Vector Autoregressive model (SVAR):

$$X_t = \sum_{j=1}^{d} a_j X_{t-j} + \sum_{j=1}^{d} b_j Y_{t-j} + \epsilon_t \tag{2.2}$$

$$Y_t = \sum_{j=1}^{d} c_j Y_{t-j} + \sum_{j=1}^{d} d_j X_{t-j} + \eta_t \tag{2.3}$$

where ϵ_t and η_t are uncorrelated white noise series. In this model, $\mathrm{Y_t}$ Granger causes $\mathrm{X_t}$ if some of the $\mathrm{b_t}$ are not zero. Likewise, $\mathrm{X_t}$ Granger causes $\mathrm{Y_t}$ if some of the $\mathrm{c_t}$ are not zero. If both are true, there is feedback between $\mathrm{X_t}$ and $\mathrm{Y_t}$.

2.1.3 Multivariate Granger Causality

The original paper only contained examples for bivariate and trivariate cases. It has since been expanded to include multiple variables[9], commonly by extending the VAR model described in 2.22.3 For a vector x of variables at time t $x_t = x_{1t}, x_{2t}, ..., x_{pt}$

$$X_t = \sum_{j=1}^{d} A_j x_j + \epsilon_t \tag{2.4}$$

where the parameters and variables are defined as in equation 2.1, a series $\mathrm{x_i}$ Granger causes series $\mathrm{x_j}$ if and only if $A^k_{ji} \neq 0$ for some $1 \leq k \leq d$. Since the resulting coefficient matrices can be dense, penalties are often used to punish overly complex models.

2.2 Heterogeneous Graphical Granger Model

The paper by Behzadi, Hlaváčková-Schindler, and Plant[1] proposes a Heterogeneous Graphical Granger Model (HGGM) using a penalised VAR-based algorithm to detect causal links. It uses generalised linear models (GLM) and an adaptive lasso penalty to ensure consistency. Unlike other models which are designed for homogeneous data, HGGM allows for the detection of Granger causalities in heterogeneous datasets.

The model starts with a VAR model as in equation 2.4 where ϵ_t is white noise. This model is penalized with the adaptive lasso penalty:

$$R(\beta_i) := \sum_{j=1}^{p} \omega_j |\beta_j \quad where \quad \omega_j = \frac{1}{|\hat{\beta}_j^{(mle)}|^{\omega}} \tag{2.5}$$

where ω_j is the weight vector for some $\omega > 0$ and $\hat{\beta}_j^{(mle)}$ is the maximum likelihood estimate of the parameters.

In order to detect Granger causalities between heterogeneous time series, HGGM uses the GLM framework introduced by Nelder and Baker[11]. It is an extension of linear regression to any distribution from the exponential family. With that relations between variables in the regression are not linear but depend on the link function g, depending on the time series' concrete distribution. The resulting objective function is:

$$\hat{\beta}_i = arg\,\min_{\hat{\beta}_i} \sum_{T=d+1}^{n} [-x_i^T(X_{T,d}^{Lag}.\beta_i) + g_i^{-1}(X_{T,d}^{Lag}.\beta_i)] + \lambda. \sum_{j=1}^{p} \omega_j|\beta_j|. \tag{2.6}$$

where $X_{T,d}^{Lag}$ is a vector of lagged variables up to time T in a time window of length d $X_{T,d}^{Lag} = \{x_i^{T-t}|i = 1, ..., p; t = 1, ..., d\}$, $\hat{\beta}$ is a vector of the regression coefficients $\hat{\beta}_1, ..., \hat{\beta}_p$ corresponding to the time series $x_1, ..., x_p$, and λ is the regularisation parameter.

In the paper, HGGM was compared to three other methods that are applicable to time series: transfer entropy(TEN)[12], Crack[13], and SFGC[14].

On synthetic data, HGGM has a higher F-measure than the other methods on all tested distributions (Gamma-Bernoulli, Gamma-Gaussian, Gaussian-Bernoulli, Poisson-Bernoulli, Gamma-Poisson, Poisson-Gaussian). When tested on a Poisson-Gaussian combination, HGGM outperforms the others on dependencies larger than 0.3 regardless of the number of features and causal relations. HGGM is also the fastest on time series longer than 2000.

The algorithm also outperforms the others on real data. The dataset consisted of 6 measurements: temperature, sunshine hours, altitude, precipitation, longitude, and latitude for 394 weather stations in Germany.

3 Related Work

This chapter discusses several other works that apply various causal methods to climate data and an alternative method[2] for detecting Granger causalities.

3.1 Causalities in Climate Data

The application of causal methods to climate data is an emerging field, with a potential for advancing our understanding of the relationships between various climate factors. Many questions can be framed as causal inference problems. But, since experiments with the climate are unfeasible, observational data can be used with various machine learning or statistical methods to find causalities between various systems. Several methods have been developed for analysing causal relationships between climate phenomena. Reviews[4][5][6] of some of these methods have been made by Jakob Runge and associates.

According to [6], the strengths of Granger causality and its non-parametric extensions are: "significance assessment and non-parametric versions", and weaknesses are: "dealing with contemporaneous effects and feedback cycles, high-dimensionality, deterministic dependencies, synergistic effects, time scales, unobserved variables".

Other methods mentioned are nonlinear state-space methods, conditional independence-based algorithms, the PCMCI (Peter–Clark (PC) algorithm[15] for condition selection with a momentary conditional independence (MCI) test for false positives)[5] algorithm, information-theoretic algorithms, structural causal models, invariance-based methods, and Bayesian score-based approaches. Due to a lack of papers comparing various methods, a platform (causeme.net)[16][6] was proposed to provide data for causality benchmarking.

Another paper[6] compared several methods for causal effect estimation. The performance of correlation, FullCI (full conditional independence), the PC(Peter–Clark) algorithm, the PCMCI algorithm, and Lasso regression were compared on a synthetic dataset created to model the challenges of analysing time series in complex systems. PCMCI has the highest detection power in all cases. FullCI, a method similar to Granger causality, has a detection power of 80% for 5 variables but drops to 40% for 20. It also cannot be applied if the number of variables exceeds the sample size.

3.2 Heterogeneous Graphical Granger Causality by Minimum Message Length

In the case of shorter time series, the inference in HGGM[1] can suffer from overestimation. To remedy this, a method[2] similar to HGGM was developed using the minimum message length principle to find causal connections in a heterogeneous graphical Granger model.

HMML assumes that when there are several models with equal measure of fit accuracy, the one with the most concise explanation is correct.

Two methods for finding the time series that have a causal connection with the target series were proposed:

- A genetic-type algorithm - HMMLGA
- An exhaustive search algorithm – exHMML

The minimisation of HMML using a genetic algorithm is based on another genetic algorithm (MMLGA)[17] for the Poisson graphical Granger model. In this paper, it is extended to all exponential distributions. Several experiments were done for the paper comparing the performance of HMML to Lingam[18], HGGM[1] and statistical framework Granger causality (SFGC)[19]. The experiments were done on synthetic datasets with time series of lengths between 100 and 1000. One dataset had 5 time series (2 gamma, 2 Gaussian, and 1 Poisson) and dense (graph with 18 edges) and sparse (graph with 8 edges) edge densities. The other had 8 time series (7 gamma and 1 Gaussian), and dense (graph with 52 edges) and sparse (graph with 15 edges) edge densities. HMML outperformed the others in all cases, with HMMLGA performing better in some cases and exHMML in others.

While HMML provides better results than HGGM, its computational complexity is higher. With p time series, m population size, n generations, d lag and $M = max\{(pd)^3, \frac{(n-d)(n-d+1)}{2}\}$ HMMLGA has a complexity of $\mathcal{O}(p^2 mnM)$ and exHMML has a complexity of $\mathcal{O}(p2^p M)$. Compared to HGGM with a complexity of $\mathcal{O}(np^2d^2)$, both HMML methods are slower.

An analysis of wind speed data was made using HMML for a presentation at the European Geosciences Union[3]. The results of this analysis are discussed at 5 and compared to the results from HGGM on the same data.

4 HGGM python implementation

For this thesis, the HGGM algorithm[1] was implemented using python. Initially, an Elastic net penalty was used instead of the adaptive lasso used in the HGGM paper and the MATLAB implementation. This was due to a lack of widely used python libraries that have regressions with an adaptive lasso penalty function, or that allow the usage of a custom penalty function. However, the results from this were far worse than the MATLAB implementation.

To solve this problem, the regressions are calculated using a MATLAB script. Matlab engine is used to pass the data from the main python code to the MATLAB script. It then returns the results of the regression to the main python code. The python code is written to allow an easy replacement of the dependency on the MATLAB script to allow a future pure python implementation in case an adaptive Lasso python library is developed.

Note that each version of MATLAB is only compatible with certain versions of python, defined on the official website[20]. The python version of the HGGM algorithm was implemented and tested using python version 3.10 and MATLAB version 2022b.

The MATLAB and python implementations were both tested on two datasets:

- A Gaussian dataset used in the HGGM paper.
- A synthetic dataset with 3 Gaussian and 3 gamma series of 1000 samples each.

The lag and threshold for causality in each dataset were found by testing the classification performance of HGGM on lags from 0 to 20 and thresholds from 0.0 to 0.1 (step 0.01) and selecting the best ones. 50 iterations of each HGGM implementation were then run for each dataset, and the f-measures were averaged. The results are:

Dataset	Lag	Threshold	Matlab	Python
Gaussian	4	0.01	0.4545	0.4762
Synthetic	1	0.0	0.2	0.2

Table 4.1: F-measures of the MATLAB and python HGGM implementations

The python implementation of HGGM has similar accuracy as the MATLAB implementation on the tested datasets.

4.0.1 Python libraries

MATLAB Engine API for Python

The MATLAB Engine API for Python[21] provides a Python package named MATLAB that enables calling MATLAB functions from Python. It requires MATLAB and a matching version of python to be installed[20]. After starting a MATLAB process in the python code with the command eng=matlab.engine.start_matlab(), MATLAB functions can be called with the provided engine eng. The library is used to call the MATLAB script for the adaptive lasso regression.

NumPy

NumPy[22] is an open-source project that enables numerical computing with Python. It contains multidimensional array and matrix data structures. It also provides ndarray, a homogeneous n-dimensional array object, with methods to efficiently operate on it. This object is used in multiple other libraries like scikit-learn, statsmodels, and scipy. It is also used to represent matrix data in the python HGGM implementation as it is faster and more compact than the default python lists.

scipy

Scipy[23] is a python library for scientific and technical computing. It extends NumPy to provide additional tools for array computing and specialized data structures. It also provides a python wrapper for highly optimized implementations of algorithms written in compiled languages such as C, C++, and FORTRAN.

The stats module of scipy is used in the python HGGM for the automatic fitting of distributions. Scipy is also a foundation for scikit-learn.

scikit-learn (aka sklearn)

Scikit-learn[24] is a python library for machine learning. It started as an extension for scipy and uses NumPy data structures for high-performance linear algebra and array operations. Among other things, it provides python extensions for regression algorithms. However, it does not support adaptive lasso penalties so a MATLAB script was used instead to calculate the regressions for HGGM.

statsmodels

statsmodels[25] is a Python module that provides classes and functions for the estimation of many different statistical models, as well as for conducting statistical tests, and statistical data exploration. The VAR model from statsmodels.tsa.api is used to find the best fitting lag using the VAR.select_order function.

4.0.2 MATLAB script

A MATLAB script was developed to calculate the regressions for HGGM. It is based on the original MATLAB implementation, with the addition of support for the inverse Gaussian distribution. It receives the matrix phi, lag, lambda, t, and the distribution of the time series. It returns the regression coefficients for the time series.

4.1 Installation instructions

4.1.1 Requirements

- Licensed Matlab installation
- Python version compatible with the installed MATLAB [26]
- Python libraries:
  - scipy
  - NumPy
  - matlab.engine
  - scikit-learn
  - statsmodels

Note that the MATLAB engine is incompatible with virtual environments (like those created by eg. conda). The system installation of python must be used or the MATLAB engine might not work properly.

4.1.2 Installation and setup

1. Install MATLAB and a compatible python version
2. Pull the git repository from https://github.com/MaSersic/hggm
3. Install the required python libraries by running bash or command prompt in the folder and running the commands:
   a) python -m pip install –upgrade pip
   b) pip install -r requirements.txt

4.1.3 Use

To calculate HGGM, the init_matlab() function must be used first to start a MATLAB instance and provide an engine eng. The provided engine is used for the hggm_matlab function, along with the time series data, max lag, lambda, and distributions. The actual lag is automatically calculated using a VAR model with Akaike information criteria based selection. The continuous distributions (Gaussian, inverse Gaussian, gamma) can be automatically detected from the data using the helpers.fit_data function.

4.1.4 Helper functions

The helpers.py python file contains functions to find the best fitting distribution and lag for the data. It also has functions to scale the data and the HGGM betas.

Distribution fitting

If the data has unknown continuous distributions, the distributions can be calculated using the helpers.fit_data function. It uses the kstest function from scipy.stats for each time series in the data to perform the Kolmogorov-Smirnov test for goodness of fit. The best fitting distribution of the following is assigned: Gaussian, inverse Gaussian, gamma. If the data is not positive, it is assigned a Gaussian distribution as the inverse Gaussian and gamma cause errors with negative or zero values.

Lag calculation

The best fitting lag is calculated using the find_lag_aic(max_lag, data) function. It uses the VAR model from statsmodels.tsa.api to create a VAR model from the data. The VAR.select_order function is used with the "aic" (i.e. Aikake information criterion) criterion to find the best fitting lag up to the provided maximum. To prevent errors in the lag calculation, the maximum lag cannot be greater than 12 * (length(data) / 100)$^{1/4}$. If the calculated lag is 0, the function returns a lag of 1.

5 Era5 data experiments

This chapter describes the analysis of the first dataset: era5 data in the area of the Andau wind park taken during 12 wind storms. The data is the same as the data used in a presentation[3] that uses HMML[2] and is analysed in a similar way. The HGGM[1] betas and beta proportionalities (beta divided by the sum of all betas) are then calculated. The statistical significance of the difference between the means of the storm and control beta proportionalities is then calculated, as is the difference between the average storm and control beta proportionalities. The statistical significance of the difference in the average wind speed in the storms and controls is calculated using a t-test. Results from HGGM are compared with the results from HMML in the EGU presentation.

5.1 Data

For this analysis, era5 data taken during 12 European wind storms in the Andau wind park in Austria was used. The time series were extracted from this data in the same way as in the EGU presentation, in order to allow a comparison between HGGM and HMML. The time frame starts 72 hours before the storm begins and ends 24 hours after the storm ends. The total length is at least 96 hours or more, depending on the duration of the storm.

Each storm has an accompanying control series. It has the same length as the storm, and occurs shortly before or after the storm.

Storm name	Storm duration	Control duration
Jeanett	2002.10.25 - 2002.10.31	2002.10.08 - 2002.10.14
Kyrill	2007.01.15 - 2007.01.24	2007.02.07 - 2007.02.16
Paula	2008.01.24 - 2008.01.27	2008.02.10 - 2008.02.13
Emma	2008.02.28 - 2008.03.07	2008.03.16 - 2008.03.24
Andrea	2012.01.03 - 2012.01.09	2012.01.24 - 2012.01.30
Niklas	2015.03.29 - 2015.04.06	2015.03.15 - 2015.03.23
Egon	2017.02.12 - 2017.02.13	2017.01.31 - 2017.02.01
Herwart	2017.10.29 - 2017.10.29	2017.11.09 - 2017.11.09
Ana	2017.12.10 - 2017.12.16	2017.11.25 - 2017.12.01
Friederike	2018.01.17 - 2018.01.19	2018.01.08 - 2018.01.10
Hervé	2020.02.02 - 2020.02.08	2020.01.08 - 2020.01.14
Ciar	2020.02.08 - 2020.02.16	2020.02.19 - 2020.02.27

Table 5.1: Durations of wind storms.

The best fitting distribution for the times series is found using a Kolmogorov-Smirnov test.

The lag for each time series is determined using a var model with Akaike information criteria based selection.

Each time series has 14 variables:

- blh – boundary layer height
- cape – convective available potential energy
- d – divergence
- cc – fraction of cloud cover
- z – geopotential
- o3 – ozone mass mixing ratio
- pv – potential vorticity
- r – relative humidity
- q – specific humidity
- t – temperature
- vo – relative vorticity
- w – vertical velocity
- wd – wind direction
- ws – wind speed

5.2 Distributions and lags

Each storm series and accompanying control have the distribution and lag calculated separately. The lags and wind speed distributions are from the EGU paper are:

Storm	Series	Lag	Distribution (ws)
Jeanett	Storm	15	Gaussian
	Control	15	Inverse Gaussian
Kyrill	Storm	16	Inverse Gaussian
	Control	16	Inverse Gaussian
Paula	Storm	10	Gamma
	Control	10	Gamma
Emma	Storm	16	Inverse Gaussian
	Control	16	Gamma
Andrea	Storm	15	Gaussian
	Control	15	Gamma
Niklas	Storm	16	Gaussian
	Control	16	Gamma
Egon	Storm	7	Inverse Gaussian
	Control	7	Gaussian
Herwart	Storm	5	Gaussian
	Control	5	Gamma
Ana	Storm	15	Gamma
	Control	15	Gaussian
Friederike	Storm	8	Inverse Gaussian
	Control	8	Inverse Gaussian
Hervé	Storm	15	Gaussian
	Control	15	Gamma
Ciar	Storm	16	Gaussian
	Control	16	Gaussian

Table 5.2: Lags and wind speed time series distributions for each Storm. From [3]

The lags and wind speed series distributions calculated for the HGGM experiments are mostly similar to the ones from the HMML experiments.

Storm	Series	Lag	Distribution (ws)
Jeanett	Storm	15	Gaussian
	Control	15	Inverse Gaussian
Kyrill	Storm	16	Inverse Gaussian
	Control	2	Inverse Gaussian
Paula	Storm	11	Gamma
	Control	11	Gamma
Emma	Storm	16	Inverse Gaussian
	Control	16	Gamma
Andrea	Storm	15	Gaussian
	Control	15	Gamma
Niklas	Storm	16	Gaussian
	Control	16	Gamma
Egon	Storm	8	Inverse Gaussian
	Control	8	Gaussian
Herwart	Storm	6	Gaussian
	Control	6	Gaussian
Ana	Storm	15	Gamma
	Control	15	Gamma
Friederike	Storm	10	Gamma
	Control	10	Inverse Gaussian
Hervé	Storm	15	Gaussian
	Control	15	Gamma
Ciar	Storm	16	Gaussian
	Control	16	Gaussian

Table 5.3: Lags and wind speed time series distributions for each Storm. Calculated for the HGGM experiments.

5.3 HMML results

HMML was applied using the genetic algorithm (HMMLGA) for optimisation. The betas for wind speed were scaled resulting in beta proportionalities = beta_{max}i / sum(beta_{max}).

Variables	Storm beta proportionalities	Ctr beta proportionalities
blh (boundary layer height)	0.000	0.025
cape (convective available potential energy)	0.015	0.033
d (divergence)	0.039	0.000
cc (fraction of cloud cover)	0.087	0.121
z (geopotential)	0.155	0.269
o3 (ozone mass mixing ratio)	0.002	0.054
pv (potential vorticity)	0.273	0.025
r (relative humidity)	0.165	0.133
q (specific humidity)	0.076	0.061
t (temperature)	0.003	0.009
vo (relative vorticity)	0.039	0.101
w (vertical velocity)	0.000	0.102
wd (wind direction)	0.099	0.000
Ws (wind speed)	0.047	0.065

Table 5.4: Beta proportionalities from HMML. Results from [3].

HMML detects some causality to wind speeds in most cases except for boundary layer height and vertical velocity in storms, and divergence and wind direction in controls.

Variables with the strongest effect on wind speed in storms are potential vorticity, relative humidity, geopotential.

Variables with the strongest effect on wind speed in controls are geopotential, relative humidity, and fraction of cloud cover.

5.4 HGGM

Initial tests with HGGM resulted in over-estimations (extremely high betas larger than 10^{300}). To avoid over-estimations several variables with high correlation had to be removed.

A Pearson was used test to find correlations between the variables. Variables with a high correlation (correlation > 0.5 and p < 0.05) are:

- blh and ws (correlation = 0.652, p = $7.6*10^{-11}$)
- d and w (correlation = 0.759, p = $8.8*10^{-6}$)
- pv and vo (correlation = 0.741, p = $1.7*10^{-10}$)
- t and q (correlation = 0.528, p = 0.0029)

Blh, d, and vo were removed. T and q were left in the dataset as the correlation is just above the cut-off and much lower than with the other variables.

Boundary layer height[14] is the height of the atmospheric layer in which wind speed and direction are affected by frictional interaction with objects on the Earth's surface. It could have an impact on wind speed, which explains the correlation.

Divergence[15] are pressure fluctuations that affect the vertical movement of air and could impact the vertical velocity of wind.

5.4.1 Results

To get more consistent results, 20 applications of HGGM were run on each storm and control series, and the results were averaged. Then the average beta for all storms and controls was calculated. The average raw betas for wind speed across all storms are:

	cape	cc	z	o3	pv	r	q	t	w	wd	ws
Storm	0.0051	0	0.00019	0	0.00694	0	0.00733	0	0	0.16872	0.73584
Control	0.00093	0.00619	7,00E-05	0	0.00041	0	0.00194	0	0	0.00952	0.61651

Table 5.5: Raw HGGM betas.

Since most variables have some effect on wind speeds, beta proportionalities (beta divided by the sum of all betas) were calculated to better see the relative magnitude of the effect each variable has on wind speeds. The "beta proportionalities" from both HGGM and HMML are:

	HGGM		HMML	
	Storm	Control	Storm	Control
blh	-	-	0	0.025
cape	0.00552	0.00147	0.015	0.033
d	-	-	0.039	0
cc	0	0.00973	0.087	0.121
z	0.00021	0.0001	0.155	0.269
o3	0	0	0.002	0.054
pv	0.00751	0.00064	0.273	0.025
r	0	0	0.165	0.133
q	0.00793	0.00305	0.076	0.061
t	0	0	0.003	0.009
vo	-	-	0.039	0.101
w	0	0	0	0.102
wd	0.18258	0.01499	0.099	0
Ws	0.79626	0.97002	0.047	0.065

Table 5.6: HGGM and HMML beta proportionalities.

During both storms and controls, wind speed has the highest effect. In storms, however, wind speed has a relatively lower effect. Wind direction has the second highest effect on wind speeds during storms, possibly because stronger winds come from certain directions. Convective available potential energy, potential vorticity, specific humidity, and geopotential also have a causal effect on wind speed in storms. Cloud cover, ozone mass mixing ratio, relative humidity, temperature, and vertical velocity were not found to be causal.

In controls, the wind speed has by far the highest effect followed by wind direction, cloud cover, specific humidity, convective available potential energy, potential vorticity, and geopotential. Ozone mass mixing ratio, relative humidity, temperature, and vertical velocity were not found to be causal.

Due to the dropped variables, the data can't be directly compared to the data from HMML. However, HGGM found fewer variables that are causal to wind speed than HMML did. This is consistent with the findings in the HMML paper.

5.4.2 Difference

The difference in HGGM beta proportionalities between the storms and the controls is:

	Storm	Control	Diff
cape	0.00552	0.00147	0.0041
cc	0	0.00973	-0.0097
z	0.00021	0.0001	0.0001
o3	0	0	0
pv	0.00751	0.00064	0.0069
r	0.01058	0.00011	0.0105
q	0.00793	0.00305	0.0049
t	0	0	0
w	0	0	0
wd	0.18258	0.01499	0.1676
ws	0.79626	0.97002	-0.1738

Table 5.7: Difference in HGGM beta proportionalities between the storms and the controls.

The biggest difference in causality between storms and controls is in the wind direction (much higher in storms) and wind speed (higher in controls). Cloud cover is causal in controls but not in storms. Convective available potential energy, potential vorticity, specific humidity, and geopotential have a higher causal effect on wind speed in storms than in controls.

5.4.3 HGGM statistical validation

A t-test was done comparing the sets of storm beta proportionalities with the control beta proportionalities to check that there is a statistically significant difference between the average beta proportionalities in the storms and controls. The testing is done using the scipy.stats.ttest_ind function. The null hypothesis is that the average of the storm beta proportionalities is the same as the average of the control beta proportionalities.

	Difference	p-value
cape	0.00405	**0.002**
cc	-0.00973	nan
z	0.00011	0.769
o3	0	nan
pv	0.00687	0.588
r	0.0105	**0.012**
q	0.00488	0.418
t	0	0.566
w	0	nan
wd	0.16759	0.981
Ws	-0.17376	0.253

Table 5.8: Difference in HGGM beta proportionalities and the p-value. P-values lesser than 0.05 are in bold.

The t-test could not return a result in some cases, mostly where the difference is 0. Only convective available potential energy (cape) and relative humidity (r) have a statistically significant difference between the average of the storm beta proportionalities and the average of the control beta proportionalities ($p<0.05$). Despite having the highest difference in average beta proportionalities, wind direction, and wind speed are not statistically significant.

6 Turbine data experiments

This chapter describes the analysis of the second dataset, measurements taken from wind turbines in the Andau wind park in the period from the beginning of 2000. to the end of 2020. It starts with the analysis of the 2020. data only to focus on the differences between the readings on different turbines. The full dataset is then analysed and HGGM betas and beta proportionalities are calculated. The statistical significance of the difference in the average wind speed in the storms and controls is calculated using a t-test and the causalities are discussed.

6.1 2020. data

In order to determine possible causes of extremely high and extremely low wind speeds, data from wind turbines was analysed using the HGGM method to find the Granger causalities.

The data comes from 38 turbines in the Andau wind park. The measurements were taken from January 2000. to the end of 2020. For the start, only the data from the year 2020. was analysed. The shorter time frame is used to make it easier to find ways to split the data into time series that can be processed by HGGM.

Each measurement has 10 variables:

- 'blh' - Boundary layer height
- 'd' – Divergence
- 'z' – Geopotential
- 'o3' - Ozone mass mixing ratio
- 'pv' - Potential vorticity
- 'relative_humidity' - Relative humidity
- 't' – Temperature
- 'w' - Vertical velocity
- 'vo' - Vorticity (relative)
- 'wdir100m' - Wind direction at 100m
- 'wspeed100m' - Wind speed at 100m

There are two types of wind speed extremes, high (speed above 14.8 m/s) and low (speed below 0.2 m/s). 0.273% of measurements (912 in total) are high extremes, and 0.095% (318 in total) are low extremes.

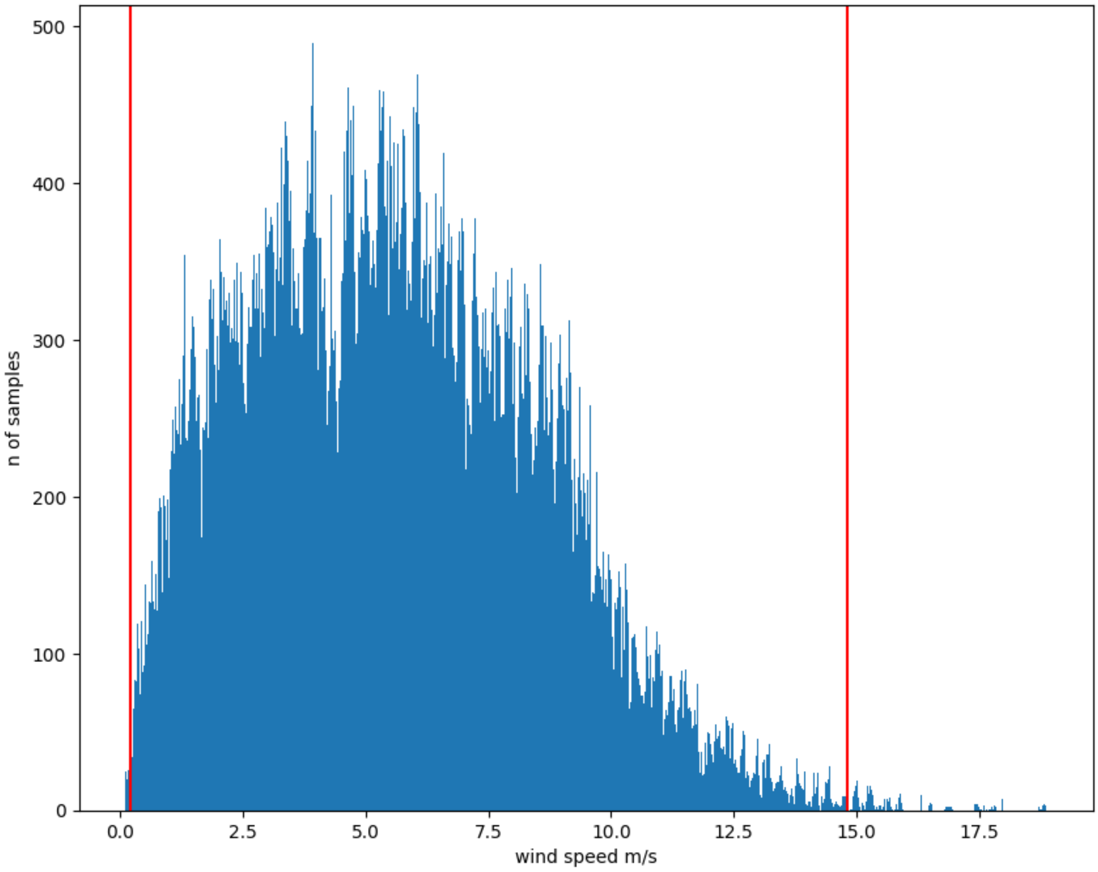

Figure 6.1: Wind speed distribution and low/high extreme cut-offs

In order to find the Granger causalities using HGGM, the data is split into time series centred around the extremes. These time series are run through the HGGM python implementation and the resulting coefficients (betas) are used to determine causalities. Initially, the time series started 72 hours before the extreme and ended 24 hours after. However, these time series were too short and resulted in HGGM outputting extremely high betas (over 10^{300}).

To minimise this overestimation in the HGGM calculation, various combinations of lengths of before and after time were tested. The time frame with no overestimations starts 150 hours before the extreme and ends 50 hours after the extreme. Significantly longer or shorter time series caused issues with overestimated HGGM coefficients. The same issue occurred with different ratios of hours before to hours after (e.g. a time frame that starts 100 hours before and ends 100 hours after the extreme).

In order to avoid overlapping data, if the time frames of multiple extreme series overlap they are combined. Then, the time series starts 150 hours before the first extreme and ends 50 hours after the last extreme in the time series.

The total length of the extreme series is at least 200 hours. It can be longer if several overlapping extremes are combined. The length then depends on the number of extremes in the series.

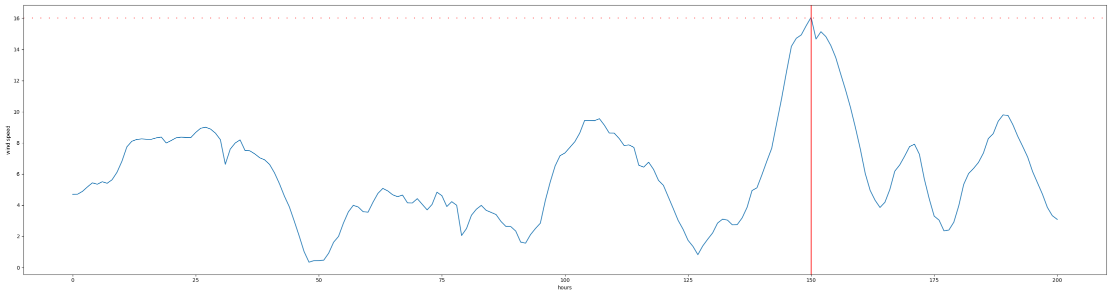

Figure 6.2: Wind speed plot for a time series with one extreme. The total duration is 200 hours (150 before and 50 after).

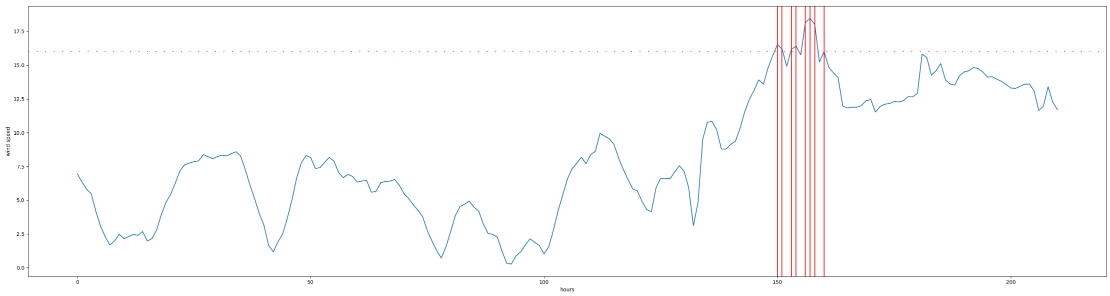

Figure 6.3: Wind speed plot for a time series with multiple extremes. The total duration is approx. 230 hours (150 before the extremes, 50 after, and approx. 30 in between).

There are three kinds of time series analysed:

High extreme time series – speed above 14.8 m/s

Low extreme time series - speed below 0.2 m/s

Control time series – 200h long time series that start 24 hours after the end of each extreme series. They are only used if they do not contain extremes. Time series which contain extremes are removed from the control set.

The time series are also separated by season into summer (01.04. - 31.09.) and winter (01.01. – 30.03. and 01.10. – 31.12) time series.

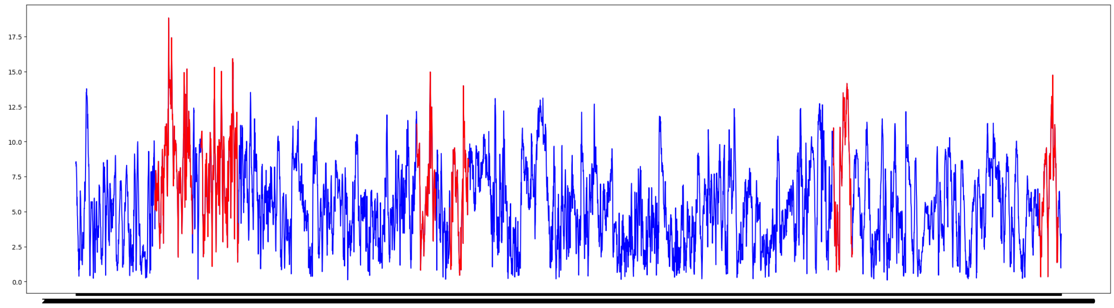

Figure 6.4: Wind speeds on turbine 0 with high extreme series in red.

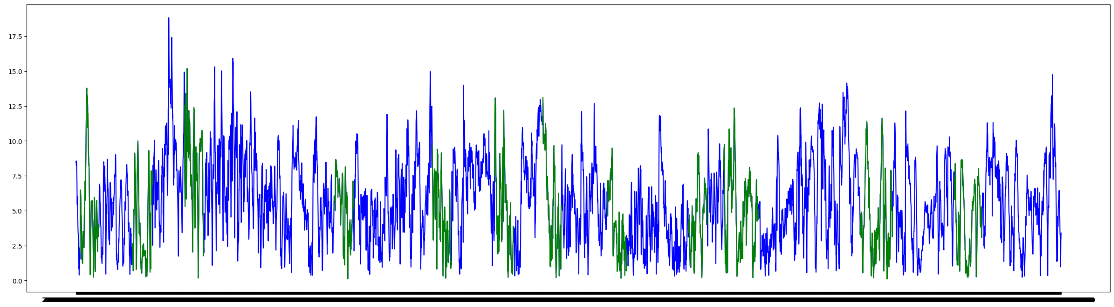

Figure 6.5: Wind speeds on turbine 0 with low extreme series in green.

The wind speeds are almost identical for all turbines. When the wind speeds for several turbines are plotted, they almost perfectly overlap:

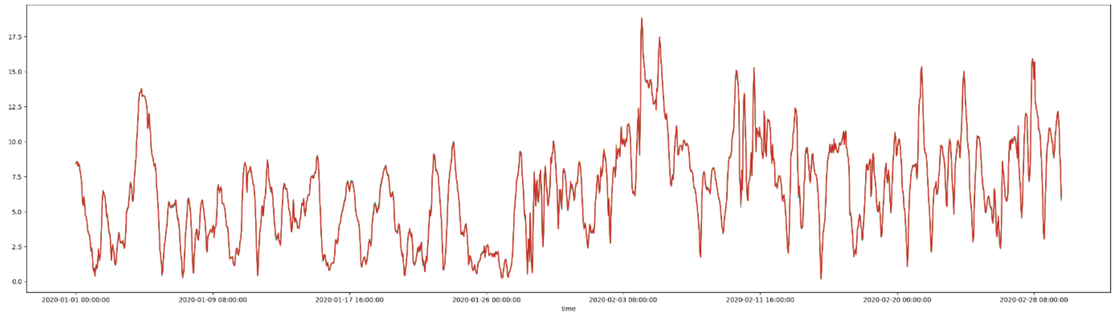

Figure 6.6: Wind speed plots for 5 different turbines.

Since the wind speeds are similar across turbines, the time series from all turbines are combined. In 2020. the number of time series is:

	Summer	Winter
Low	167	35
Control	38	43
High	38	76

Table 6.1: Number of time series per season and type

10 time series of each type are randomly selected for the HGGM calculation. The best fitting distribution for the wind speed is found using a Kolmogorov-Smirnov test. The control series are all Gaussian, the high extreme series are mostly inverse Gaussian, and the low extreme series are mostly gamma in summer and mostly inverse Gaussian in winter.

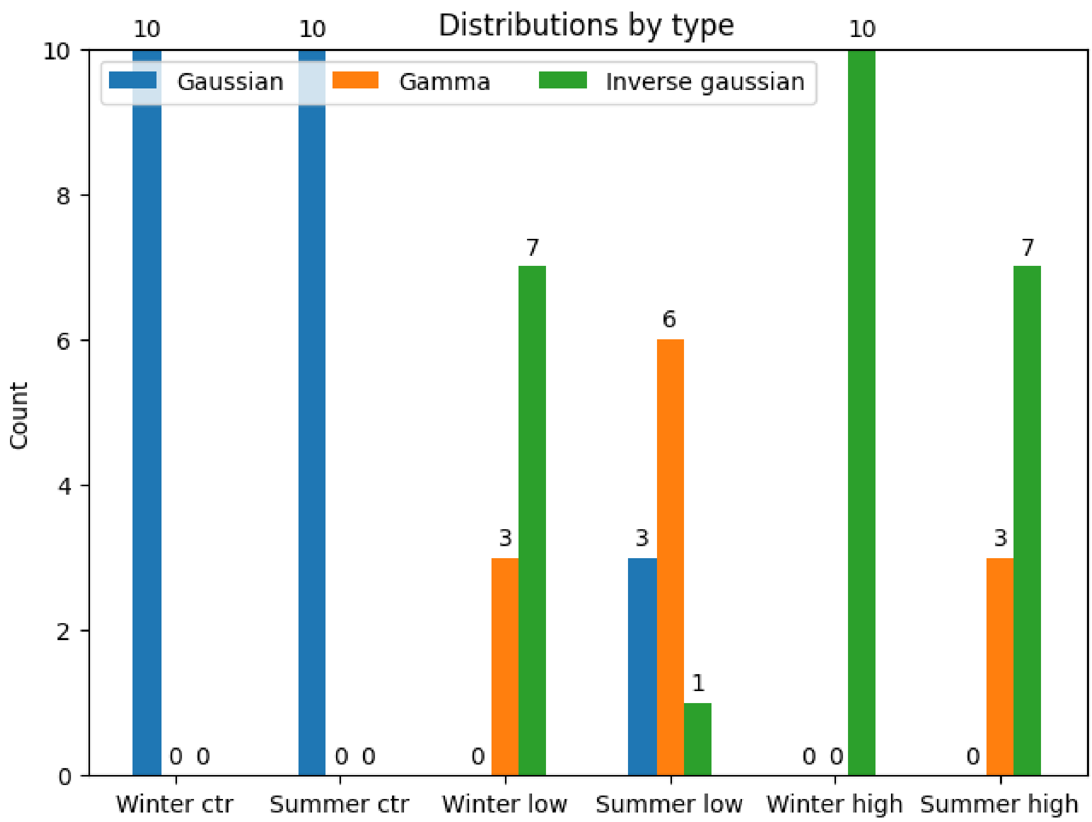

Figure 6.7: Distributions of the wind speed time series by type.

The lag for each time series is determined using a var model with Akaike information criteria based selection. Winter low and high series mostly have lower lags from 2 to 4.

Other series have a lag of 14.

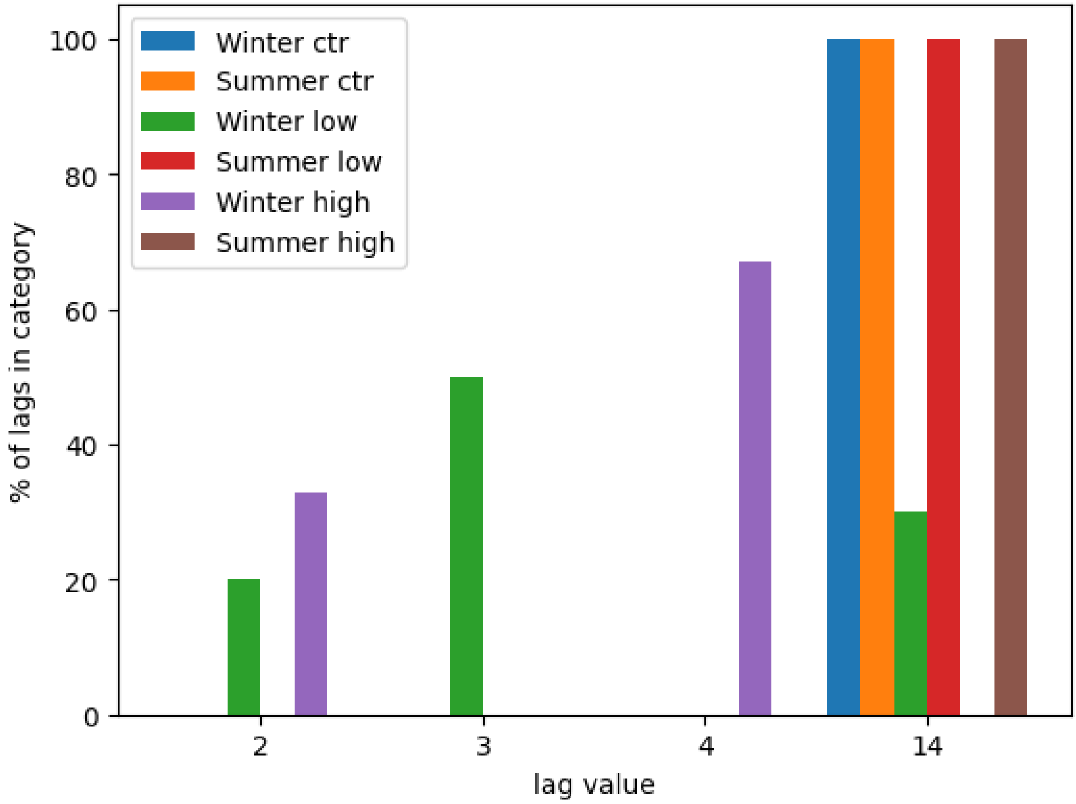

Figure 6.8: Percentage of lags in each time series by type.

6.1.1 2020. HGGM results

The HGGM results were calculated using the distributions and lags assigned by the Kolmogorov-Smirnov test and the Akaike information criteria based selection as with the storm data.

HGGM is performed on the 50 time series and the results are averaged. These are the betas for the wind speed:

vars	Winter			Summer		
	High	Control	Low	High	Control	Low
blh	0.00000	0.00000	0.00003	0.00000	0.00010	0.00000
d	0.00000	0.00000	-0.0001	0.00000	0.00000	0.00000
z	0.00000	0.00010	0.00011	0.00000	0.00017	0.00004
o3	0.00000	0.00000	0.05542	0.00000	0.00000	0.00000
pv	0.00000	0.00000	0.00565	0.00000	0.00000	0.00000
rel_h	0.00000	0.00000	0.39916	0.00000	0.03119	0.00000
t	0.00294	0.00054	0.01299	0.00448	0.01660	0.00000
w	0.00000	0.00000	-0.0000	0.00000	0.00000	0.00000
vo	0.00000	0.00000	0.00005	0.00000	0.00000	0.00000
wdir	0.00000	0.00000	0.00004	0.00000	0.00000	0.00000
wspeed	0.70397	0.97191	0.30399	0.61661	0.84715	0.37079

Table 6.2: Raw HGGM betas for the 2020. data.

To better understand the relative impact of each variable, beta proportionalities (beta divided by the sum of all betas) were also calculated. The beta proportionalities are:

vars	Winter			Summer		
	High	Control	Low	High	Control	Low
blh	0.00000	0.00000	0.00004	0.00000	0.00011	0.00001
d	0.00000	0.00000	-0.0002	0.00000	0.00000	0.00000
z	0.00000	0.00010	0.00014	0.00000	0.00019	0.00010
o3	0.00000	0.00000	0.07128	0.00000	0.00000	0.00000
pv	0.00000	0.00000	0.00727	0.00000	0.00000	0.00000
rel_h	0.00000	0.00000	0.51343	0.00000	0.03484	0.00000
t	0.00416	0.00055	0.01671	0.00721	0.01854	0.00001
w	0.00000	0.00000	-0.0000	0.00000	0.00000	0.00000
vo	0.00000	0.00000	0.00007	0.00000	0.00000	0.00000
wdir	0.00000	0.00000	0.00005	0.00000	0.00000	0.00000
wspeed	0.99584	0.99935	0.39101	0.99279	0.94632	0.99989

Table 6.3: Beta proportionalities for the 2020. data.

Unsurprisingly, wind speed is mostly caused by wind speed except for winter low extreme time series. In that case, relative humidity has the highest causal effect on wind speed, but wind speed still has a significant effect.

Since the wind speed is so dominant in the results, the beta proportionalities were recalculated for the variables with the wind speed removed.

vars	Winter			Summer		
	High	Control	Low	High	Control	Low
blh	0.00000	0.00000	0.00007	0.00000	0.00203	0.06722
d	0.00000	0.00000	-0.0003	0.00000	0.00000	0.00000
z	0.00001	0.15230	0.00024	0.00000	0.00357	0.85655
o3	0.00000	0.00000	0.11705	0.00000	0.00000	0.00000
pv	0.00000	0.00000	0.01193	0.00000	0.00000	0.00000
rel_h	0.00000	0.00000	0.84309	0.00000	0.64898	0.00000
t	0.99999	0.84770	0.02744	1.00000	0.34542	0.07622
w	0.00000	0.00000	-0.0000	0.00000	0.00000	0.00000
vo	0.00000	0.00000	0.00011	0.00000	0.00000	0.00000
wdir	0.00000	0.00000	0.00009	0.00000	0.00000	0.00000

Table 6.4: . Beta proportionalities without wind speed for the 2020. data.

In winter high extreme and control time series, the temperature has the highest effect on wind speed. In winter low extreme series, relative humidity has the highest effect. Geopotential also has a significant effect on wind speed in winter control time series, and ozone mass mixing ratio has a significant effect in winter low extreme time series. Other variables have little to no effect.

In the summer high extreme time series, temperature is the only variable with any effect on wind speed. This might be an outlier due to limited data in 2020. only. Relative humidity has the highest effect on wind speed in the summer control series, followed by temperature.

In summer low extreme time series wind speed is mostly caused by geopotential, and temperature and boundary layer height have some effect. However, if we look at the betas, HGGM found very little causal effects other than wind speed. This might also be an outlier due to limited data.

6.2 2000. – 2020. data

The same process of splitting the dataset into time series was then applied to the full dataset. Due to the data from October 2018. missing from the dataset, measurements from that month were excluded.

With cut-offs of 14.8 m/s for the high extremes and 0.2 m/s for the low extremes, there are 19628 (0.319%) high extremes and 3190 (0.051%) low extremes. Because of the large number of high extremes, the high cut-off could be increased to 16 m/s and still have a sufficient number of time series for the analysis. With the higher cut-off, there are 7909 (0.128%) high extremes.

The wind speed at which turbines need to be shut down is 25 m/s, however, the wind speed in this dataset never gets that high.

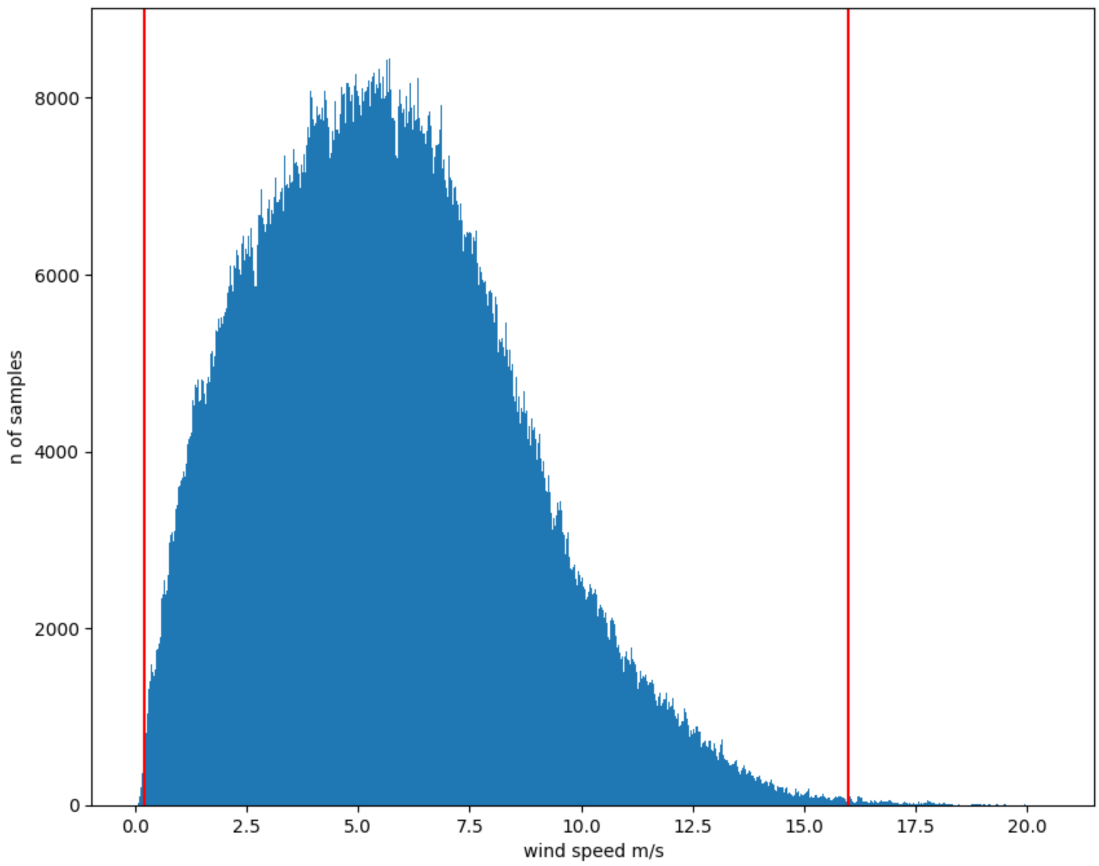

Figure 6.9: Wind speed distribution for the full dataset with the new cut-offs.

The time series were created in the same way as in the 2020. data analysis. The number of time series by type is:

	Summer	Winter
Low	684	684
Control	117	132
High	380	714

Table 6.5: Number of time series per season and type.

Due to the calculation time, 50 time series of each type were randomly selected. The best fitting distributions for the wind speed were found using a Kolmogorov-Smirnov test:

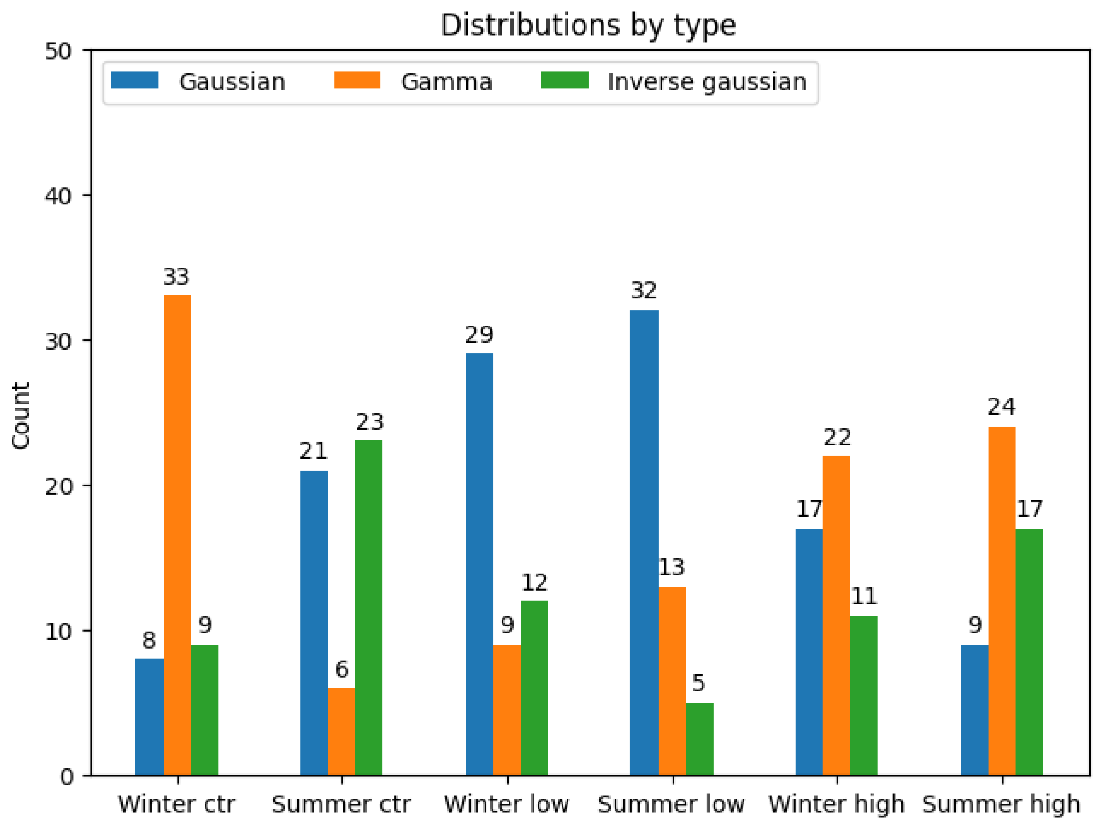

Figure 6.10: Distributions of the wind speed time series by type.

With the expanded dataset and more time series the winter control series are mostly gamma, and the summer controls are mostly Gaussian and inverse Gaussian. The low extreme series are mostly Gaussian, unlike in the 2020. data when they were mostly gamma and inverse Gaussian. The high extreme series are a mix of all distributions but mostly gamma.

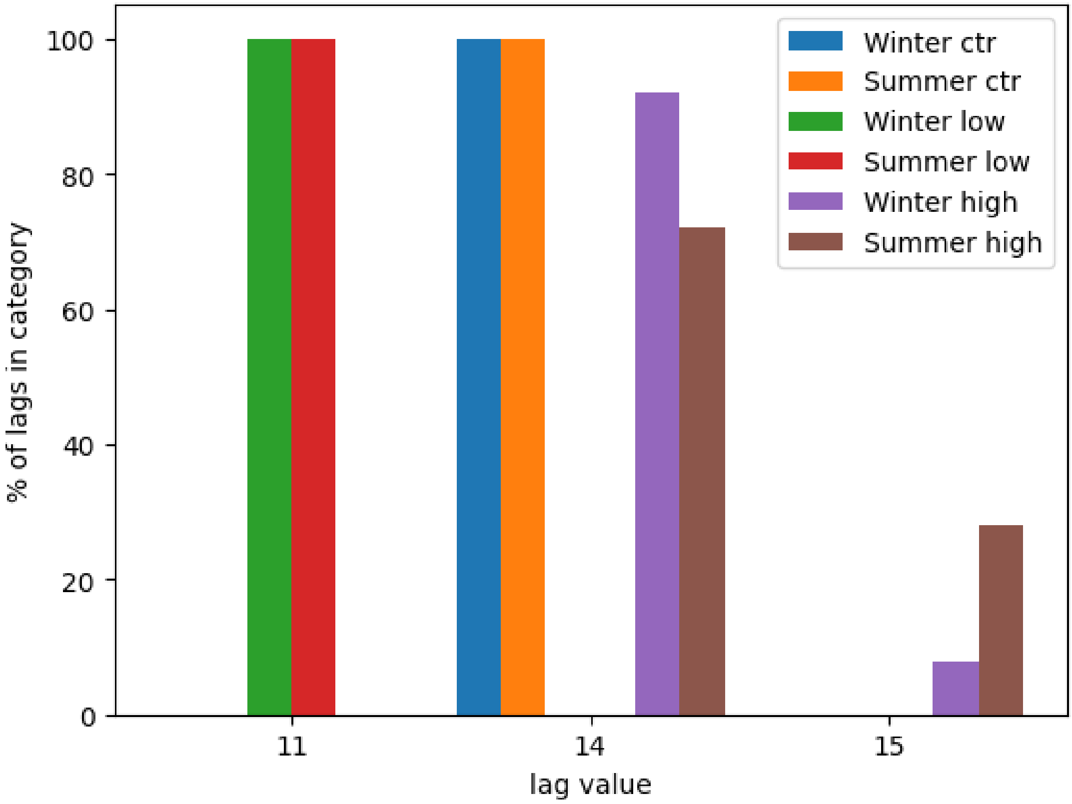

Figure 6.11: Distributions of the wind speed time series by type.

The low extreme time series have a slightly lower lag of 11. The rest have lags of 14 or 15. Unlike the 2020. data, there are no time series with a lag of 3 or 4.

6.3 HGGM results

The average betas for the sample are:

vars	Winter			Summer		
	High	Control	Low	High	Control	Low
blh	0.00024	0.00007	0.00024	0.00009	0.00010	0.00011
d	0.00000	0.00000	0.00000	0.00003	0.00000	0.00000
z	0.00004	0.00003	0.00016	0.00020	0.00016	0.00002
o3	0.00000	0.00004	0.00000	0.00029	0.00003	0.00000
pv	0.00000	0.00012	0.00000	0.00175	0.00012	0.00000
rel_h	0.00000	-0.0000	0.00715	0.06381	0.01615	0.00000
t	0.00663	0.01010	0.01664	0.01862	0.00349	0.01005
w	0.00000	0.00005	-0.0000	0.00000	0.00000	0.00000
vo	0.00000	0.00005	0.00000	0.00002	0.00000	0.00000
wdir	0.00003	0.00026	0.00033	0.00012	0.00004	0.00063
wspeed	0.99306	0.98927	0.97548	0.91507	0.97990	0.98919

Table 6.6: Raw HGGM betas for the 2000. – 2020. data.

The beta proportionalities are:

vars	Winter			Summer		
	High	Control	Low	High	Control	Low
blh	0.00014	0.00003	0.00016	0.00004	0.00009	0.00007
d	0.00000	0.00000	0.00000	0.00001	0.00000	0.00000
z	0.00002	0.00001	0.00011	0.00009	0.00015	0.00001
o3	0.00000	0.00002	0.00000	0.00014	0.00003	0.00000
pv	0.00000	0.00005	0.00000	0.00083	0.00011	0.00000
rel_h	0.00000	-0.0000	0.00464	0.03015	0.01510	0.00000
t	0.00378	0.00393	0.01081	0.00880	0.00326	0.00634
w	0.00000	0.00002	-0.0000	0.00000	0.00000	0.00000
vo	0.00000	0.00002	0.00000	0.00001	0.00000	0.00000
wdir	0.00002	0.00010	0.00021	0.00005	0.00004	0.00040
wspeed	0.56605	0.38505	0.63344	0.43235	0.91630	0.62407

Table 6.7: Beta proportionalities for the 2000. – 2020. data.

With the larger dataset, wind speed is still mostly caused by wind speed. However, now there are some seasonal differences. In winter the causal effect of wind speed is the lowest in the control group, while in summer it is the highest.

Recalculated without the wind speed:

vars	Winter			Summer		
	High	Control	Low	High	Control	Low
blh	0.03492	0.00685	0.00996	0.00110	0.00496	0.01021
d	0.00000	0.00020	0.00000	0.00036	0.00019	0.00000
z	0.00547	0.00246	0.00671	0.00231	0.00805	0.00194
o3	0.00000	0.00409	0.00000	0.00340	0.00137	0.00000
pv	0.00000	0.01095	0.00000	0.02065	0.00609	0.00000
rel_h	0.00000	-0.0012	0.29155	0.75131	0.80356	0.00000
t	0.95507	0.94199	0.67852	0.21921	0.17342	0.92967
w	0.00000	0.00435	-0.0000	0.00000	0.00000	0.00000
vo	0.00000	0.00510	0.00000	0.00029	0.00023	0.00000
wdir	0.00455	0.02413	0.01326	0.00137	0.00212	0.05817

Table 6.8: Beta proportionalities without wind speed for the 2000. – 2020. data.

Variables in high extreme and control time series have similar causalities to wind speed in both summer and winter. In summer, humidity has the biggest effect on wind speed, followed by temperature. Other variables have little to no effect. In winter, only temperature seems to have a significant effect on wind speeds.

In the low extreme time series, however, there is a significant difference from the controls. In summer, the wind speeds are caused almost exclusively by temperature and humidity has no causal effect. In winter, wind speed is still caused mostly by temperature but humidity now also has a significant effect.

6.4 Differences

6.4.1 Winter

Winter	High	Control	Diff
blh	0.00014	0.00003	0.00011
d	0	0	0
z	0.00002	0.00001	0.00001
o3	0	0.00002	-0.00002
pv	0	0.00005	-0.00005
r	0	0	0
t	0.00378	0.00393	-0.00015
w	0	0.00002	-0.00002
vo	0	0.00002	-0.00002
wdir	0.00002	0.00010	-0.00008
wspeed	0.56605	0.38505	0.18100

Table 6.9: Difference between winter high extremes and the controls.

In winter high extremes in wind speed are caused by wind speed much more than controls. Boundary layer height has a higher effect on wind speed during high extremes than during controls, and geopotential has a slightly higher effect during high extremes.

Potential vorticity, vertical velocity, ozone mass mixing ratio, and relative vorticity are causal for wind speeds during controls but not during high extremes. Wind direction has a higher effect on wind speed during controls and temperature has a slightly higher effect on wind speed during controls.

Winter	Low	Control	Diff
blh	0.00016	0.00003	0.00013
d	0	0	0
z	0.00011	0.00001	0.00010
o3	0	0.00002	-0.00002
pv	0	0.00005	-0.00005
r	0.00464	0	0.00464
t	0.01081	0.00393	0.00688
w	0	0.00002	-0.00002
vo	0	0.00002	-0.00002
wdir	0.00021	0.00010	0.00011
wspeed	0.63344	0.38505	0.24839

Table 6.10: Difference between winter low extremes and the controls.

Winter low extremes are also caused by wind speed much more than controls. Temperature, relative humidity, boundary layer height, geopotential, and wind direction also have a higher effect on wind speed during low extremes. Potential vorticity, vertical velocity, ozone mass mixing ratio, and relative vorticity are causal for wind speeds during controls but not during high extremes.

6.4.2 Summer

Summer	High	Control	Diff
blh	0.00004	0.00009	-0.00005
d	0.00001	0	0.00001
z	0.00009	0.00015	-0.00006
o3	0.00014	0.00003	0.00011
pv	0.00083	0.00011	0.00072
r	0.03015	0.0151	0.01505
t	0.0088	0.00326	0.00554
w	0	0	0
vo	0.00001	0	0.00001
wdir	0.00005	0.00004	0.00001
wspeed	0.43235	0.9163	-0.48395

Table 6.11: Difference between summer high extremes and the controls.

In summer wind speed has a significantly lower effect on wind speeds during high extremes than during controls. Relative humidity, temperature, and potential vorticity have a significantly higher effect on wind speed during high extremes than during controls. Ozone mass mixing ratio and wind direction have a slightly higher effect on wind speed during high extremes. Divergence and relative vorticity are causal for wind speed during high extremes but not during controls. Boundary layer height and geopotential have a higher effect on wind speeds during controls.

Summer	Low	Control	Diff
blh	0.00007	0.00009	-0.00002
d	0	0	0
z	0.00001	0.00015	-0.00014
o3	0	0.00003	-0.00003
pv	0	0.00011	-0.00011
r	0	0.0151	-0.0151
t	0.00634	0.00326	0.00308
w	0	0	0
vo	0	0	0
wdir	0.0004	0.00004	0.00036
wspeed	0.62407	0.9163	-0.29223

Table 6.12: Difference between summer low extremes and the controls.

Just as during high extremes, wind speed also has a lower effect on wind speed extremes than on controls. Relative humidity, geopotential, potential vorticity, ozone mass mixing ratio, and boundary layer height also have a lower effect during low extremes than during controls. Temperature and wind direction have a higher effect on wind speed during low extremes than during controls.

6.4.3 t-test

To test if the difference between the extreme and control beta proportionalities is statistically significant, a t-test was done comparing the sets of high/low extreme beta proportionalities with the control beta proportionalities. The testing is done using the scipy.stats.ttest_ind function. The null hypothesis is that the average of the high/low extreme beta proportionalities is the same as the average of the control beta proportionalities.

	WH	p	WL	p	SH	p	SL	p
blh	0.00011	0.075	0.00013	**0.011**	-0.00005	0.780	-0.00002	**0.002**
d	0.00000	0.440	0.00000	0.212	0.00001	0.314	0.00000	0.168
z	0.00001	0.093	0.00010	0.309	-0.00006	0.056	-0.00014	0.126
o3	-0.00002	0.320	-0.00002	0.320	0.00011	0.319	-0.00003	0.320
pv	-0.00005	0.450	-0.00005	0.320	0.00072	0.292	-0.00011	0.306
r	0.00000	0.226	0.00464	0.283	0.01505	**0.012**	-0.01510	0.174
t	-0.00015	0.093	0.00688	**0.016**	0.00554	**0.025**	0.00308	**0.000**
w	-0.00002	0.280	-0.00002	**0.042**	0.00000	0.319	0.00000	0.104
vo	-0.00002	0.501	-0.00002	0.369	0.00001	0.320	0.00000	0.334
wdir	-0.00008	0.155	0.00011	0.810	0.00001	0.057	0.00036	0.072
wspeed	0.18100	0.966	0.24839	0.290	-0.48395	0.813	-0.29223	**0.000**

Table 6.13: The differences between the average beta proportionalities of the high/low extremes and the controls, as well the p-value from the t-test. P-values lesser than 0.05 are in bold.

For the winter high extremes: none of the tests return a $p < 0.05$. The difference in beta proportionalities between extremes and controls is also very low across all variables.

For the winter low extremes: temperature, boundary layer height, and vertical velocity return a $p < 0.05$.

For the summer high extremes: relative humidity and temperature return a $p < 0.05$.

For the summer low extremes: wind speed, temperature, and boundary layer height return a $p < 0.05$.

6.4.4 Causality

Focusing only on variables with a p-value under 0.05:

	WH	WL	SH	SL
blh	nr	0.00013	nr	-0.00002
r	nr	nr	0.01505	nr
t	nr	0.00688	0.00554	0.00308
w	nr	-0.00002	nr	nr
wspeed	nr	nr	nr	-0.29223

Table 6.14: Beta proportionalities from HGGM. Cases with $p>0.05$ are marked as not relevant (nr).

For the winter high extremes, none of the variables were found to have a statistically significant difference in the average value compared to the controls. This is possibly due to the wind speeds in winter being generally higher than in summer, leading to a smaller difference between high extremes and controls.

Despite having by far the largest beta proportionalities, wind speed is only statistically significant for low extremes in summer.

Temperature is statistically significant in all cases except with winter high extremes and has consistently high beta proportionalities.

Relative humidity is only statistically significant for summer high extremes, but it has the highest beta proportionalities in that case.

Boundary layer height has a statistically significant but small effect on wind speed during low extremes in both summer and winter.

Vertical velocity has a statistically significant but small effect on wind speed during low extremes in winter.

7 Conclusion and Future Work

7.1 Conclusion

This thesis investigated the use of the Heterogeneous Graphical Granger Model (HGGM)[1] to find causal relationships in two datasets. The first uses era5 data recorded during 12 European wind storms, and the second uses meteorological data recorded by the wind farm's turbines between 2000. and 2020.

For the purpose of this thesis, a python implementation of HGGM was made. Unfortunately, no well-supported python libraries allow the use of generalized linear models with an adaptive Lasso penalty. Because of this, a MATLAB script had to be used for the regression and MATLAB engine was used to communicate with the python code.

For the era5 storms dataset, some highly correlated variables had to be dropped in order to avoid overestimations of HGGM coefficients. These problems did not occur in the analysis done for the EGU presentation using HMML[2], suggesting that HMML might be less prone to overestimations when applied to shorter time series. HGGM also finds fewer causal relationships to wind speed than HMML does on the era4 storm dataset, though the results are not strictly comparable due to the dropped variables.

In the era5 storm dataset, HGGM finds that wind direction, potential vorticity and convective available potential energy have a higher effect on wind speed during wind storms than during the control series. However, only the convective available potential energy and relative humidity are statistically significant.

In the turbine data, summer and winter were analysed separately. In both summer and winter, wind speed has the highest effect on wind speeds. However, in summer it has the highest effect during control winds, and in winter it is the opposite. Excluding wind speed, relative humidity has the highest effect on wind speeds during high and control winds, and temperature has the highest effect during low winds. In winter, however, temperature has the highest effect in all cases, and humidity only has a significant effect during low winds. In addition to these meteorological parameters boundary layer height, potential vorticity, and wind direction were found to have a smaller but still significant effect in some cases.

Despite HGGM finding some causalities between most meteorological parameters and wind speed, only the following meteorological parameters were found to cause wind speed extremes with $p < 0.05$:

- Temperature: During winter low extremes and summer high and low extremes.
- Boundary layer height: During winter and summer low extremes.
- Relative humidity: During summer high extremes.

- Vertical velocity: During winter low extremes.
- Wind speed: During summer low extremes

While they are statistically relevant, boundary layer height and vertical velocity have very low beta proportionalities. Therefore, their effect on wind speed extremes is statistically significant but low.

Temperature, (relative) humidity, and wind speed have generally high beta proportionalities and therefore a high effect on wind speeds during extremes.

7.2 Future Work

Overall, the wind speeds in the Andau wind park are relatively low. In the observed period they never reach the 25 m/s max speed at which the turbines need to be shut down. Analysing data from areas with higher winds might result in more causalities being found. Having a higher cut-off point for the high wind extremes might also solve the issue with HGGM finding no statistically relevant beta proportionalities during winter high extremes.

Currently, there are no python libraries that allow the use of adaptive Lasso as a regression penalty or at least the use of a user-defined penalty. If such a library was developed the use of MATLAB for the regression would not be necessary and a fully python based implementation could be made.

Bibliography

[1] Sahar Behzadi, Kateřina Hlaváčková-Schindler and Claudia Plant. 'Granger Causality for Heterogeneous Processes'. en. In: *Advances in Knowledge Discovery and Data Mining*. Ed. by Qiang Yang et al. Vol. 11441. Series Title: Lecture Notes in Computer Science. Cham: Springer International Publishing, 2019, pp. 463–475. ISBN: 978-3-030-16142-2. DOI: 10.1007/978-3-030-16142-2_36. URL: http://link.springer.com/10.1007/978-3-030-16142-2_36 (visited on 09/05/2022).

[2] Kateřina Hlaváčková-Schindler and Claudia Plant. 'Heterogeneous Graphical Granger Causality by Minimum Message Length'. In: *Entropy* 22.12 (2020). ISSN: 1099-4300. DOI: 10.3390/e22121400. URL: https://www.mdpi.com/1099-4300/22/12/1400.

[3] Hlavackova-Schindler (Schindlerova), K. et al. *The influence of meteorological parameters on wind speed extreme events: A causal inference approach*. Vienna, Austria, May 2022. URL: https://doi.org/10.5194/egusphere-egu22-5756.

[4] Jakob Runge et al. 'Causal inference for time series'. en. In: *Nature Reviews Earth & Environment* 4.7 (June 2023), pp. 487–505. ISSN: 2662-138X. DOI: 10.1038/s43017-023-00431-y. URL: https://www.nature.com/articles/s43017-023-00431-y (visited on 06/05/2024).

[5] Jakob Runge et al. 'Detecting and quantifying causal associations in large nonlinear time series datasets'. en. In: *Science Advances* 5.11 (Nov. 2019), eaau4996. ISSN: 2375-2548. DOI: 10.1126/sciadv.aau4996. URL: https://www.science.org/doi/10.1126/sciadv.aau4996 (visited on 06/05/2024).

[6] Jakob Runge et al. 'Inferring causation from time series in Earth system sciences'. en. In: *Nature Communications* 10.1 (June 2019), p. 2553. ISSN: 2041-1723. DOI: 10.1038/s41467-019-10105-3. URL: https://www.nature.com/articles/s41467-019-10105-3 (visited on 06/05/2024).

[7] Jakob Runge, Vladimir Petoukhov and Jürgen Kurths. 'Quantifying the Strength and Delay of Climatic Interactions: The Ambiguities of Cross Correlation and a Novel Measure Based on Graphical Models'. en. In: *Journal of Climate* 27.2 (Jan. 2014), pp. 720–739. ISSN: 0894-8755, 1520-0442. DOI: 10.1175/JCLI-D-13-00159.1. URL: http://journals.ametsoc.org/doi/10.1175/JCLI-D-13-00159.1 (visited on 06/05/2024).

[8] C. W. J. Granger. 'Investigating Causal Relations by Econometric Models and Cross-spectral Methods'. In: *Econometrica* 37.3 (Aug. 1969), p. 424. ISSN: 00129682. DOI: 10.2307/1912791. URL: https://www.jstor.org/stable/1912791?origin=crossref (visited on 05/05/2024).

[9] Ali Shojaie and Emily B. Fox. 'Granger Causality: A Review and Recent Advances'. en. In: *Annual Review of Statistics and Its Application* 9.1 (Mar. 2022), pp. 289–319. ISSN: 2326-8298, 2326-831X. DOI: 10.1146/annurev-statistics-040120-010930. URL: https://www.annualreviews.org/doi/10.1146/annurev-statistics-040120-010930 (visited on 05/05/2024).

[10] Mariusz Maziarz. 'A review of the Granger-causality fallacy'. en. In: *Journal of Philosophical Economics* Volume VIII Issue 2.Articles (May 2015), p. 10676. ISSN: 1844-8208. DOI: 10.46298/jpe.10676. URL: https://jpe.episciences.org/10676 (visited on 05/05/2024).

[11] J. A. Nelder and R. W. M. Wedderburn. 'Generalized Linear Models'. In: *Journal of the Royal Statistical Society. Series A (General)* 135.3 (1972), p. 370. ISSN: 00359238. DOI: 10.2307/2344614. URL: https://www.jstor.org/stable/10.2307/2344614?origin=crossref (visited on 05/05/2024).

[12] Thomas Schreiber. 'Measuring Information Transfer'. In: *Physical Review Letters* 85.2 (July 2000). arXiv:nlin/0001042, pp. 461–464. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.85.461. URL: http://arxiv.org/abs/nlin/0001042 (visited on 05/05/2024).

[13] Alexander Marx and Jilles Vreeken. *Causal Inference on Multivariate and Mixed-Type Data*. arXiv:1702.06385 [cs, stat]. Oct. 2017. URL: http://arxiv.org/abs/1702.06385 (visited on 05/05/2024).

[14] Sanggyun Kim et al. 'A Granger Causality Measure for Point Process Models of Ensemble Neural Spiking Activity'. en. In: *PLoS Computational Biology* 7.3 (Mar. 2011). Ed. by Karl J. Friston, e1001110. ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1001110. URL: https://dx.plos.org/10.1371/journal.pcbi.1001110 (visited on 05/05/2024).

[15] Peter Spirtes, Clark Glymour and Richard Scheines. *Causation, Prediction, and Search*. Ed. by J. Berger et al. Vol. 81. Lecture Notes in Statistics. New York, NY: Springer New York, 1993. ISBN: 978-1-4612-2748-9. DOI: 10.1007/978-1-4612-2748-9. URL: http://link.springer.com/10.1007/978-1-4612-2748-9 (visited on 12/05/2024).

[16] J. Muñoz-Marí et al. *CauseMe: An online system for benchmarking causal discovery methods*. 2020.

[17] Kateřina Hlaváčková-Schindler and Claudia Plant. 'Poisson Graphical Granger Causality by Minimum Message Length'. en. In: *Machine Learning and Knowledge Discovery in Databases*. Ed. by Frank Hutter et al. Vol. 12457. Series Title: Lecture Notes in Computer Science. Cham: Springer International Publishing, 2021, pp. 526–541. ISBN: 978-3-030-67658-2. DOI: 10.1007/978-3-030-67658-2_30. URL: https://link.springer.com/10.1007/978-3-030-67658-2_30 (visited on 12/05/2024).

[18] Shohei Shimizu et al. *DirectLiNGAM: A direct method for learning a linear non-Gaussian structural equation model*. arXiv:1101.2489 [stat]. Apr. 2011. URL: http://arxiv.org/abs/1101.2489 (visited on 12/05/2024).

[19] Sanggyun Kim et al. 'A Granger Causality Measure for Point Process Models of Ensemble Neural Spiking Activity'. en. In: *PLoS Computational Biology* 7.3 (Mar. 2011). Ed. by Karl J. Friston, e1001110. ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1001110. URL: https://dx.plos.org/10.1371/journal.pcbi.1001110 (visited on 12/05/2024).

[20] *Versions of Python Compatible with MATLAB Products by Release*. URL: https://www.mathworks.com/support/requirements/python-compatibility.html (visited on 27/11/2023).

[21] *Install MATLAB Engine API for Python - MATLAB & Simulink*. URL: https://www.mathworks.com/help/matlab/matlab_external/install-the-matlab-engine-for-python.html (visited on 27/11/2023).

[22] Charles R. Harris et al. 'Array programming with NumPy'. en. In: *Nature* 585.7825 (Sept. 2020). Number: 7825 Publisher: Nature Publishing Group, pp. 357–362. ISSN: 1476-4687. DOI: 10.1038/s41586-020-2649-2. URL: https://www.nature.com/articles/s41586-020-2649-2 (visited on 27/11/2023).

[23] Pauli Virtanen et al. 'SciPy 1.0: fundamental algorithms for scientific computing in Python'. en. In: *Nature Methods* 17.3 (Mar. 2020). Number: 3 Publisher: Nature Publishing Group, pp. 261–272. ISSN: 1548-7105. DOI: 10.1038/s41592-019-0686-2. URL: https://www.nature.com/articles/s41592-019-0686-2 (visited on 27/11/2023).

[24] Fabian Pedregosa et al. 'Scikit-learn: Machine Learning in Python'. In: *Journal of Machine Learning Research* 12.85 (2011), pp. 2825–2830. ISSN: 1533-7928. URL: http://jmlr.org/papers/v12/pedregosa11a.html (visited on 27/11/2023).

[25] Skipper Seabold and Josef Perktold. 'Statsmodels: Econometric and Statistical Modeling with Python'. en. In: Austin, Texas, 2010, pp. 92–96. DOI: 10.25080/Majora-92bf1922-011. URL: https://conference.scipy.org/proceedings/scipy2010/seabold.html (visited on 27/11/2023).

[26] *System Requirements for MATLAB Engine API for Python - MATLAB & Simulink*. URL: https://www.mathworks.com/help/matlab/matlab_external/system-requirements-for-matlab-engine-for-python.html (visited on 27/11/2023).