Universiteit Leiden
The Netherlands

Opleiding Informatica

Anomaly detection on pollutant levels in European air quality

Ronan Smedeman

First supervisor and second supervisor:
Matthijs van Leeuwen & Zhong Li

BACHELOR THESIS

Leiden Institute of Advanced Computer Science (LIACS)
www.liacs.leidenuniv.nl
12/08/2025

Abstract

The industrial revolution sparked the start of an increasing amount of air pollution in the world. Research later showed the detrimental impact that pollutants can have on human health. These developments sparked the start of a lot of research into this area, which can not be done without the availability of data. Therefore, a significant amount of data has been collected over the years, among which a dataset from the European Environment Agency (EEA). This dataset keeps track of large amounts of time series data with regards to outdoor European air quality.

Research has shown that it is possible to detect anomalies in indoor air quality by using an LSTM-AE model. This ability can allow policy-makers to adopt measures in an attempt to reduce harmful concentrations of air pollutants. Such data is however very different from outdoor data due to its clear spikes in concentrations that are related to room occupancy. If it would also be possible to efficiently detect anomalies in time series data for outdoor air pollutant concentrations, it could give people better insight into the possible sources of these pollutants. This could subsequently help policy-makers in the adoption of measures to reduce the negative impact of harmful pollutant concentrations.

This thesis therefore aims to find whether it is possible to effectively detect anomalous values in outdoor air quality data, with a focus on European data from the EEA dataset. An attempt will also be made to find out which anomaly detection models are most suited for such a problem and which will deliver the best results.

Due to the unlabeled nature of this dataset, a ground truth first needs to be established. This has been done by artifically injecting anomalies into time series data from the dataset. A selection of five different models, consisting from statistical, classical machine learning and deep learning models, has subsequently been applied to these time series. With the creation of a ground truth it has therefore become possible to evaluate these models based on confusion matrices and their resulting performance metrics.

It was found that it was possible to detect anomalies quite effectively depending on the specific pollutant. In the best case, almost three-quarters of the anomalies were detected. In some cases however, depending on the characteristics of the data, none of the anomalies could be detected. It was however found that some models have specific shortcomings when used in anomaly detection on such data. Global forecasting models like a Recurrent Neural Network, that made use of LSTM cells, generally achieved optimal results.

Contents

1 Introduction

The industrial revolution, and the subsequent use of combustion engines, sparked the start of an increasing amount of air pollution in the world. Realisation later set in that certain materials and particles, now increasingly found in the air, could have a deteriorating effect on human health [Hol99]. Pollutants like particulate matter (i.e. $PM_{2.5}$ and PM_{10}) have for example become primary indicators of air pollution due to their prevalence in urban atmospheres and their effect on respiratory and cardiovascular issues [DPBM25]. The World Health Organisation mentioned in 2021 that air pollution is a "silent killer" that produces the premature death of almost seven million people each year [Reu21]. Reducing air pollution has thus subsequently become an important area for research in an attempt to reduce its negative impact.

Among the research that has been done towards air pollutants and air quality in general, numerous papers have focused on the forecasting of air quality. Méndez et al. [MMN23] have written a survey about the use of machine learning algorithms in the forecasting of air quality. They remark, for example, that air pollution forecasting is very useful in being able to inform people about the pollution level, which allows policy-makers to adopt measures for reducing its impact. Machine learning techniques have become the most common methods in the forecasting of air pollution. Machine learning algorithms have both been used to predict Air Quality Indices (AQI) and the concentration of specific pollutants. Besides machine learning methods, it is also possible to apply classical regression-based algorithms, like ARIMA, on such data in order to obtain a forecast.

Forecasting methods can also be used to detect anomalies, data points/sequences in a time series that would not be expected in normal situations [SWP22]. An example of an anomalous data point in outdoor air quality could for example originate from heavier traffic than what would usually be the case. In indoor air quality the presence of more people in a room might cause unexpected spikes in CO_2 concentrations [WJJX+23]. If such anomalies can be detected it would give people better insight into possible sources of pollutants. This knowledge might also increase peoples ability to reduce air pollutant concentrations and thus improve their health.

Wei et al. [WJJX+23] have used an LSTM-AE model for anomaly detection on the earlier mentioned indoor air quality data. This research has shown that anomalies can accurately be detected in this type of data by using such models. The used indoor air quality data, however, shows clear spikes in concentrations due to room occupancy. Where this research can thus be of help in improving indoor air quality, its result do not necessarily translate to outdoor air quality where such spikes might not occur. There is thus a gap in existing knowledge as it is not known whether anomaly detection models can also be used to accurately detect anomalies in outdoor air quality data.

The goal of this thesis is therefore to find out whether it is possible to effectively detect anomalies in time series data of outdoor air quality. If it this would prove to be possible, it could help people take precautions against anomalously high concentrations of hazardous air pollutants. It could also aid policy-makers in attempts to lower the frequency with which harmful concentration levels are reached. To find out how effectively such anomalies can be detected, multiple anomaly detection models should be tested. It is namely not guaranteed that a deep learning model that works well on indoor air quality, also produces good results on outdoor air quality.

To find out whether anomalies can effectively be detected in outdoor air quality, multiple types of anomaly detection models have to be applied on such data. These models will subsequently be evaluated based on how many anomalies they can accurately detect. The models that will be evaluated have been selected from a python library called Darts [HLP+22]. This library contains a large amount of models ranging from statistical to deep learning models. Without data to apply these anomaly detection models on, this research would not be possible. Therefore a dataset from the European Environment Agency (EEA) has been selected that has been collecting data on air pollutant concentrations in 41 different countries [Eur22]. This dataset consists of hourly measurements of the concentration of all sorts of pollutants for many different measuring stations over a 10-year time period. From this dataset it is possible to extract different time series data for different locations and pollutants on which it can also be attempted to apply anomaly detection/outlier detection [CBK09]. This dataset is thus very suitable for use in research towards anomaly detection on outdoor air quality.

By evaluating models from this Darts library on data from the EEA, this thesis will attempt to fill gaps in the existing knowledge of anomaly detection on air quality data. This dataset contains data that has been collected from locations in Europe. This thesis will therefore specifically focus on anomaly detection on outdoor European air quality. Results from this thesis will hopefully allow future steps toward improving air quality and reducing the negative impact that pollutants have on human health.

1.1 Thesis overview

Chapter 1 gives an introduction of the subject. It will give background information about research into air quality and its impact on human health. The importance of being able to accurately forecast air pollutant concentrations is also explained here. An overview of the contents of this thesis is subsequently also given. Chapter 2 discusses related work to this subject and explains technical terms/concepts that will regularly be used in the remainder of the thesis. Finally, this chapter also gives a problem statement and the resulting research questions. The method of this thesis is subdivided into two main parts, namely exploratory data analysis and the actual anomaly detection. In Chapter 3, exploratory data analysis is performed in order to find out what the EEA dataset actually looks like and what data is the most useful for the application of anomaly detection. In Chapter 4 it is explained how the selected models work and how they have been applied to the dataset. Chapter 5 subsequently presents the results of applying the different anomaly detection models on the selected data from the dataset. Chapter 6 discusses both the limitations of this research and possible future research while the final chapter, Chapter 7, gives the final conclusion of this thesis.

2 Preliminaries

This chapter introduces key technical terms and concepts used throughout this thesis and provides an overview of related work. It also presents the problem statement and outlines the research questions that the thesis aims to address.

2.1 Definitions

This section explains some technical terms/concepts that will often be mentioned in this thesis.

- **Anomaly detection** refers to the problem of finding patterns in data that do not conform to expected behavior. These nonconforming patterns are often referred to as anomalies or outliers [CBK09].
- **Univariate time series** is a sequence of data points that are measured over time with each data point only consisting of one variable. In the case that each data point has multiple variables, the time series would be multivariate [SWP22].
- **Air Quality Indices (AQI)** are systems often used by countries/organizations to give an overall indication of the air quality [CJZ07] (i.e. by presenting a single number from 1-10) instead of an overview of all pollutant concentrations.
- **Seasonality** is the periodic recurrence of fluctuations [BW20]. As an example, one might think of a time series of daily average outdoor temperatures. Such a time series would show a yearly seasonal pattern with high values in the summer and low values in the winter.
- **Trend** refers to when the mean μ of a time-series is not constant, but increases or decreases over time. A trend can both be linear or non-linear [BW20].
- **Global forecasting models** are models that can be trained on multiple time series like deep learning models [HLP$^+$22].
- **Local forecasting models** are models that can be trained on a single target series only. In the Darts library this tends to be simpler statistical models like ARIMA [HLP$^+$22].

2.2 Related work

2.2.1 Air quality

Air pollution is difficult to measure in a single number as there exist many different air pollutants which all have a varying impact on human health in the short and long term. Cairncross et al. [CJZ07] have thus proposed an index system based on the relative risk of increased daily mortality, the amount of deaths that occur in a population in a 24-hour time period, associated with short-term exposure to five common air pollutants. These air pollutants are sulphur dioxide (SO_2), particulate matter (PM), nitrogen oxides (NO_x), carbon monoxide (CO) and ozone (O_3). Other air quality indices are often also based on these pollutants. These thus seem to be the most relevant focus point for anomaly detection due to these pollutants being the ones most often used for determining overall air quality.

Research by Dimitrou et al. [DPK13] uses the same dataset from the EEA to determine the most important air quality stressors. They looked at the mentioned five pollutants, that are also used to determine the European air quality index, at 14 different monitoring stations to determine their contribution to the total increased risk of mortality. This research has shown that both CO and SO_2 had a relatively low contribution to this while the remaining three: O_3, PM_{10} and NO_2 turned out to be important public health stressors. It is important to note that the contribution of O_3 varied significantly depending on the location of the measurement stations as measuring stations at locations with heavy traffic saw lower values.

2.2.2 Anomaly detection

Anomaly detection refers to the problem of finding patterns in data that do not conform to expected behavior. Chandola et al. have done a general survey about anomaly detection techniques [CBK09]. Schmidl et al. [SWP22] have taken a closer look at anomaly detection with regards to time series specifically. They have presented an overview of different methods for both univariate and multivariate time series and for unsupervised and supervised methods. This overview gives a good starting point to explore methods that can be applied to the unlabeled dataset from the EEA. The fact that this dataset is unlabeled also creates the need to find a way to evaluate the performance of the anomaly detection methods. Campos et al. [CZS+16] have done research on possible ways to evaluate unsupervised anomaly detection methods. One method for evaluating unsupervised anomaly detection methods is through injecting anomalies into an existing time series. Keogh and Wu [WK23] have done research into what are optimal practices regarding this technique.

As mentioned in Chapter 1, Wei et al. [WJJX+23] have previously done research into using an LSTM-AE model to detect anomalies in a similar indoor air quality dataset. The steps taken here could be useful in performing anomaly detection on outdoor air quality data as well. For the actual applying of the anomaly detection techniques it would be a good option to make use of models that are part of the Darts python library [HLP+22]. This library contains a combination of multiple anomaly detection techniques which can be applied to the EEA dataset.

Braei and Wagner [BW20] indicate that all anomaly detection methods on time-series data can be divided into the following three main categories.

- **Statistical methods** assume that the data is generated by a specific statistical model. Examples of this are exponential smoothing [Gar06] and ARIMA [ACADL18].

- **Classical machine learning methods** generally consider the data generation process as a black box and try to learn from the data only. The machine learning methods are based on the implicit assumption that the underlying data generation process is not relevant as long as the machine learning methods are able to produce accurate predictions. Examples of machine learning models are support vector regression and K-nearest neighbours regression (KNN) [MMN23].

- **Deep learning methods** are models that make use of neural networks. Examples of models that fall under this category are Recurrent Neural Networks (RNN) or Multiple Layer Perceptrons (MLP) [MMN23].

The boundary between statistical and machine learning approaches are vague however, which could lead to situations where it is not immediately clear where a model should be placed.

It will also be important to determine the type of anomalies that will be focused on. Chandola et al. [CBK09] originally came up with a suggestion to divide anomalies for tabular data into three different categories. Braei and Wagner [BW20] have subsequently given a similar division, but with regards to anomalies in univariate time series. They categorize the anomalies in the following three categories:

- **Point anomalies** are single data points that deviate significantly from the rest of the data.
- **Collective anomalies** are cases where individual points are not anomalous, but a sequence of points are labeled as an anomaly.
- **Contextual anomalies** are data points that can be normal in a certain context, while detected as an anomaly in a different context.

2.3 Problem statement

As mentioned in the previous section, research has already been done towards indoor CO_2 levels in classrooms [WJJX$^+$23]. This research has shown that a deep-learning model can be used to detect anomalies in time series of air pollutant concentrations. However, this indoor data contained clear peaks in concentration levels due to classroom occupancy varying quite suddenly. As outdoor air quality generally shows different patterns, this means that the results of this research do not necessarily translate to such data and that this could thus be an interesting point of focus for new research. Because the dataset of the EEA contains large amounts of air quality data for different pollutants and regions, this dataset is a good starting point for such new research. The data from this dataset similarly consists of univariate time series where the anomalies have not been labeled previously. The fact that the dataset is unlabeled causes the need to apply either unsupervised anomaly detection techniques or to find a method to label anomalies. The above information spawns the following research question:

"How effective can anomalies be detected in univariate time series data of air pollution in Europe."

If it is shown that anomalies can be detected in such time series data, it will also be of interest to see which technique can most efficiently achieve this. Given that a univariate time series in this situation is an ordered set $T = \{T_1, T_2, ..., T_m\}$ of m real-valued, one-dimensional data points with $T_i \in \mathbb{R}$ [SWP22], anomaly detection models $M = \{M_1, M_2, ..., M_n\}$ can be applied on a time series T to forecast the same value T_{m+k} with $k \in \mathbb{N}$. By making a selection of different types of models from the categories mentioned in Chapter 2.2.2 ($M_{Type} = \{M_{Stat}, M_{ML}, M_{DL}\}$), their performance in anomaly detection can subsequently be compared. It has already been shown that deep learning models, M_{DL}, can deliver promising results on indoor air quality, but this does not necessarily translate to outdoor air quality data. Braei and Wagner [BW20] have also shown that classical machine learning methods, M_{ML}, can deliver promising results on similar types of data.

This spawns the question which model would achieve the most promising results and whether M_{ML} and M_{DL} can actually achieve significantly better results than statistical models, M_{Stat}. This leads to the second research question:

"Which anomaly detection methods achieve the best performance in detecting anomalies in univariate time series data of air pollution in Europe?"

3 Exploratory data analysis

Before applying anomaly detection models to the EEA dataset, it is necessary to first explore the data to understand its structure and characteristics. If we have more information about the data, it can be determined which types of anomalies would be the most interesting point of focus and what actually constitutes as an anomaly. A decision can then hopefully also be made on what type of anomaly detection models would be likely to deliver the most promising results in the detection of such anomalies. In this step multiple types of models should be compared, like statistical and deep learning models. A method also needs to be found to evaluate these models as the dataset lacks a ground truth, which increases the difficulty of evaluation. The data exploration step should also be helpful in determining how such a ground truth can possibly be established.

3.1 Dataset

Table 1 shows characteristics of the EEA dataset. The dataset covers a large amount of measuring stations across 41 countries. These measuring stations measure 350 different air pollutants. Since the data from 2013 to 2023 has been verified, it is the most suitable for applying anomaly detection. The metadata of the dataset also lists numerous characteristics of the measuring stations like coordinates and altitude. Besides this, the area and type of each measurement station is also listed as shown in Table 1. Apart from basic information about the measuring stations, little is initially known about the actual data, such as the distribution of the measurements. The first step is thus to find out what would be the most interesting segments of the dataset to actually apply anomaly detection to.

Characteristic	**Value**
Amount of countries	41
Amount of pollutants	350
Time period	2013-2023 (hourly measurements)
Amount of measuring stations	60124
Possible station areas	Urban, Suburban, Rural
Possible station types	Background, Industrial, Traffic

Table 1: An overview of relevant characteristics of the EEA dataset.

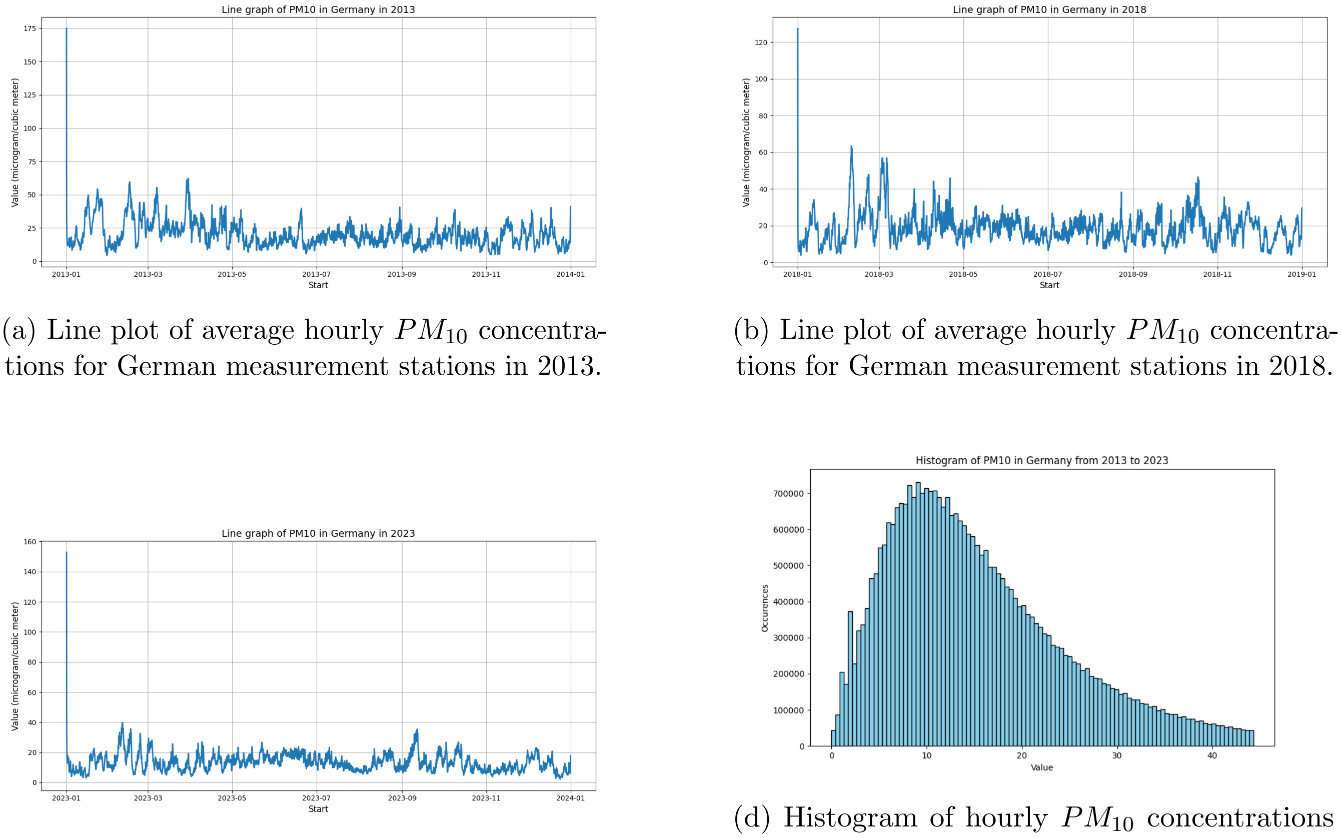
AQI's are often used to give an indication of the overall air quality and its current possible impact on human health. For most of these AQI's, the pollutants that are taken into account are very similar with only slight variations [CJZ07]. The EEA also has its own Air Quality Index which takes into account the following five pollutants: PM_{10}, $PM_{2.5}$, SO_2, O_3 and NO_2. As the research by Dimitrou et al. [DPK13] has shown, PM_{10}, O_3 and NO_2 contribute the most to an increased risk of mortality. For that reason the most logical step is to look into what the exact data for these three pollutant looks like.

Due to the datasets large size it takes a very long time to process all the data. By using a representative selection of the dataset it should however be possible to get a good overview of the dataset. To obtain such a selection, all the measuring stations of three different countries were selected. The selected countries, France, Germany and Austria, were among the ones with the largest amount of measuring stations. Both France and Germany also have a significant amount of measuring stations for all possible areas and types. Austria has less measuring stations than the other two countries, but the stations are often at higher altitudes which makes it possible to look at the effect that altitude has on the concentration of different pollutants.

3.2 Exploring PM_{10}

The first pollutant of interest is PM_{10} of which the long-term health effects of exposure are associated with shortening of life expectancy, increased rates of bronchitis and reduced lung function [CJZ07]. To generate the charts in Figure 1 the average measurements of every hour were taken for all German measurement stations to get an insight into any possible patterns. It is evident that each year begins with significantly higher values compared to the rest of the year. A likely explanation would be that this is caused by fireworks on new year's eve. Other, less extreme, points in the charts also show sudden increases and decreases which indicates that it should be possible to detect anomalies in this dataset. Figure 1d shows a histogram of all the hourly PM_{10} measurements from 2013-2023 that are within 2 standard deviations from the mean. This histogram indicates that the data has the shape of a normal distribution that is skewed slightly to the left.

To examine the impact of various station attributes, histograms of PM_{10} values were plotted in Figure 2 across different areas, altitudes, and station types. From these histograms it becomes clear that all these attributes have an impact on the skewness of the distribution with especially high altitudes seeing a significant decrease in concentrations. From these plots it becomes clear that the attributes of the measuring stations should be taken into account when selecting a training set for a deep learning model. When applying anomaly detection on data from a rural area, the model should for example also be trained on data from rural areas to achieve optimal performance.



(a) Line plot of average hourly PM_{10} concentrations for German measurement stations in 2013.

(b) Line plot of average hourly PM_{10} concentrations for German measurement stations in 2018.

(c) Line plot of average hourly PM_{10} concentrations for German measurement stations in 2023.

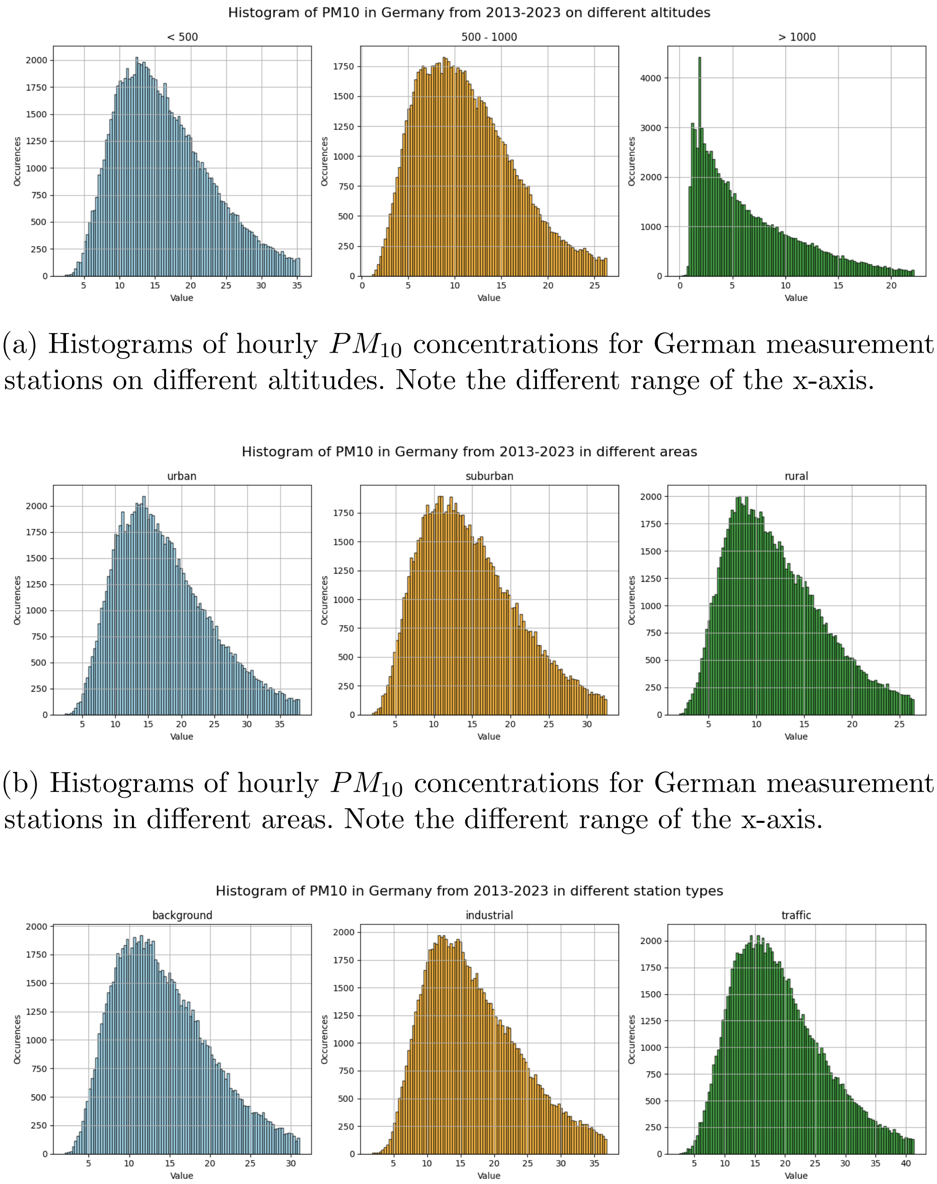
(d) Histogram of hourly PM_{10} concentrations for German measurement stations from 2013 to 2023.

Figure 1: Charts describing hourly PM_{10} concentrations in Germany.

3.3 Exploring O_3

Figure 3 shows the average concentrations of ozone (O_3) as measured by the German stations in the period from 2013 to 2023. From the line plot in Figure 3a it becomes clear that this time series exhibits seasonality, with higher concentrations generally being measured in the summer months. This seasonality also contributes to a distribution in Figure 3b that appears multimodal, with peaks around both 0 and 70 μg/m^3. Figure 4 shows the effect of different measuring station attributes. It becomes evident that O_3 concentrations are influenced by altitude, area, and station type in a manner similar to PM_{10}. The main difference here is that the impact of the altitude was not as extreme here. A higher altitude mostly resulted in an increasing mean and a decreasing variance. This once again shows the importance of training a model on relevant data for a specific type of measuring station.

What also becomes clear from line plots of the O_3 concentrations at different measuring stations, and what Figure 3 also shows, is that there is a lot more fluctuation in this data compared to the PM_{10} data. This potentially increases the difficulty of detecting anomalous values as it is less uncommon for a data point to significantly increase or decrease compared to its previous value.

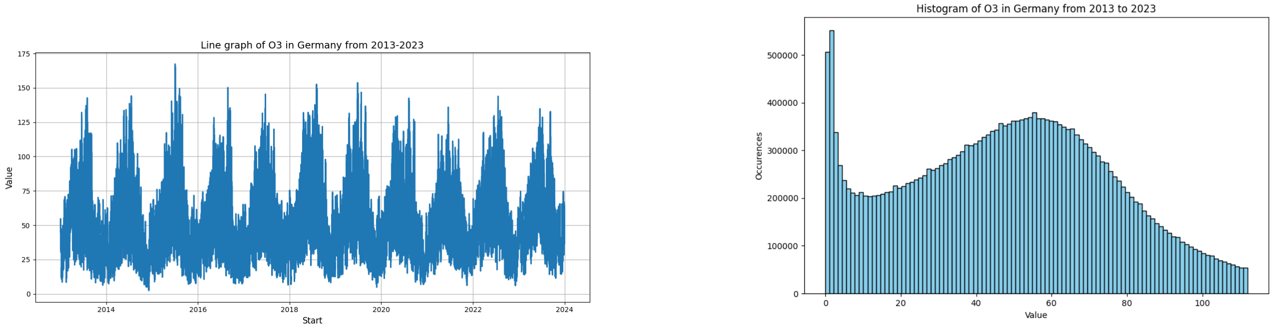


(a) Histograms of hourly PM_{10} concentrations for German measurement stations on different altitudes. Note the different range of the x-axis.

(b) Histograms of hourly PM_{10} concentrations for German measurement stations in different areas. Note the different range of the x-axis.

(c) Histograms of hourly PM_{10} concentrations for German measurement stations with different station types. Note the different range of the x-axis.

Figure 2: Histograms describing PM_{10} concentrations for measuring stations with different attributes in Germany.



(a) Line plot of average hourly O_3 concentrations for German measurement stations from 2013 to 2023.

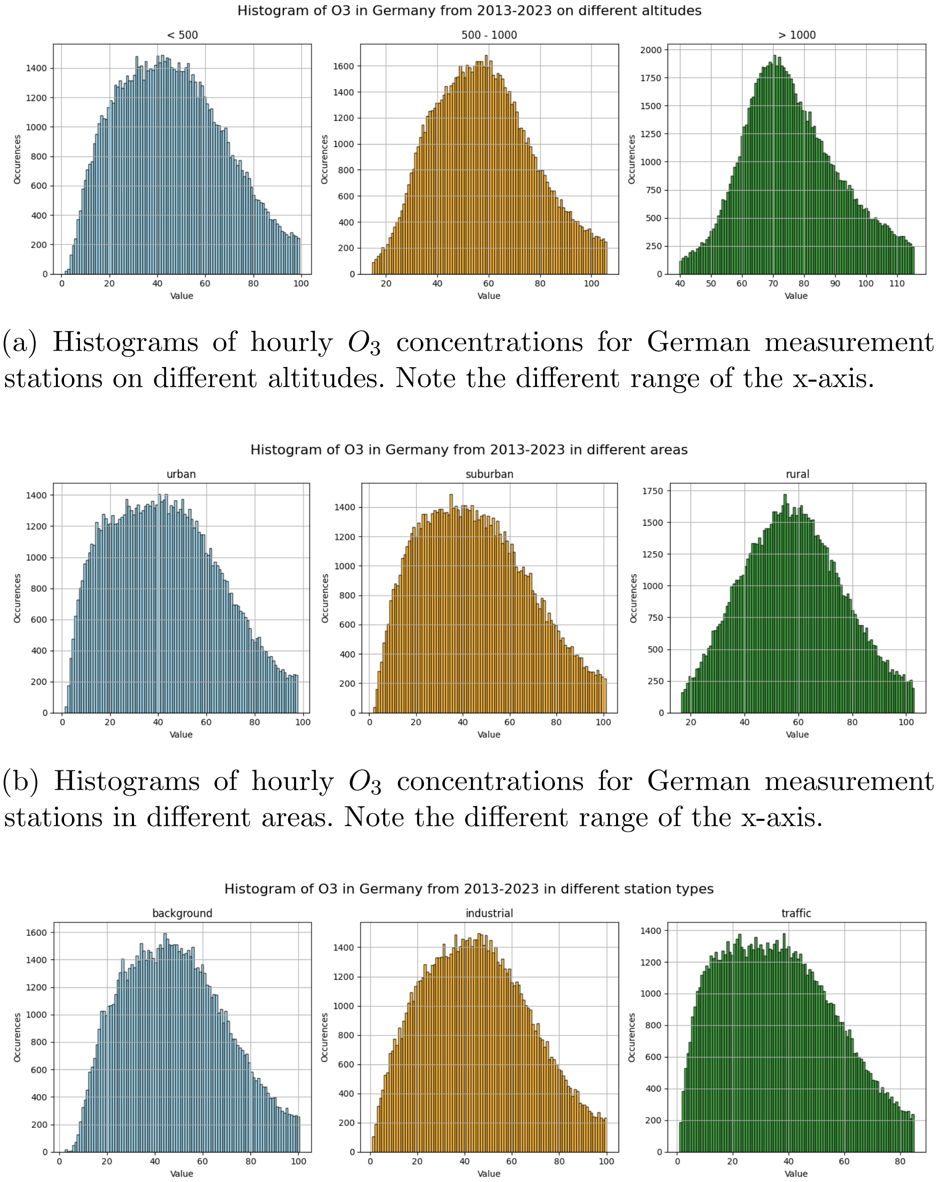
(b) Histogram of hourly O_3 concentrations for German measurement stations from 2013 to 2023.

Figure 3: Charts describing hourly O_3 concentrations in Germany.

3.4 Exploring SO_2

Figure 5 shows the average concentration of SO_2 in Germany from 2013 to 2023. These charts seem to indicate that the data contains a significantly higher amount of measurements for certain values. Quite a lot of data points had also been set to -9999 in the SO_2 time series which would seem to indicate missing data. Even though the O_3 and PM_{10} time series also contain missing data points, these do occur significantly less frequently. With so many data points seemingly containing incorrect values, it might not be the most suitable to use for anomaly detection as these values would have to be estimated, thus potentially creating non-existent anomalies or removing actual anomalies. For a time series where an individual data point is missing, it would be possible to take the average of the surrounding data points to fill in the gap. This would allow us to fill the gaps with values that would appear non-anomalous, thus reducing the negative impact of these missing points on the evaluation of the anomaly detection models. With subsequent data points having no value, it becomes increasingly difficult to fill these gaps by taking the average value of the surrounding data points. Due to this reason it will be very difficult to accurately evaluate anomaly detection models on SO_2 time series compared to the series of the other two pollutants.

An interesting observation from the average hourly SO_2 values is the clear declining trend over the ten-year period. This declining trend differentiates SO_2 from the other two pollutants which did not show a clear trend. Due to this property, SO_2 might be an interesting pollutant to focus on for the evaluation of anomaly detection models as it might provide an insight into the effect that such a trend has on their performance. SO_2 time series also show sudden spikes in values which suggests that anomalous values occur quite frequently. These characteristics might lead to SO_2 being an interesting air pollutant for anomaly detection in the case that the amount of missing data points can be reduced.

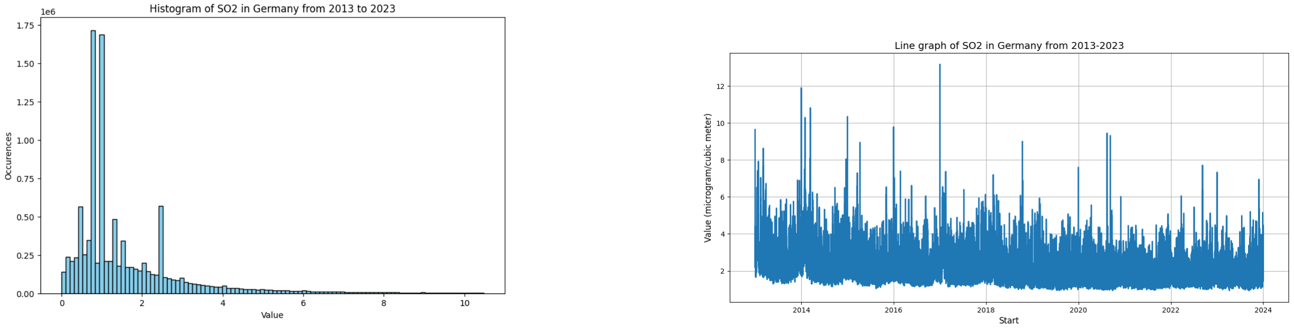


(a) Histograms of hourly O_3 concentrations for German measurement stations on different altitudes. Note the different range of the x-axis.

(b) Histograms of hourly O_3 concentrations for German measurement stations in different areas. Note the different range of the x-axis.

(c) Histograms of hourly O_3 concentrations for German measurement stations with different station types. Note the different range of the x-axis.

Figure 4: Histograms describing O_3 concentrations for measuring stations with different attributes in Germany.



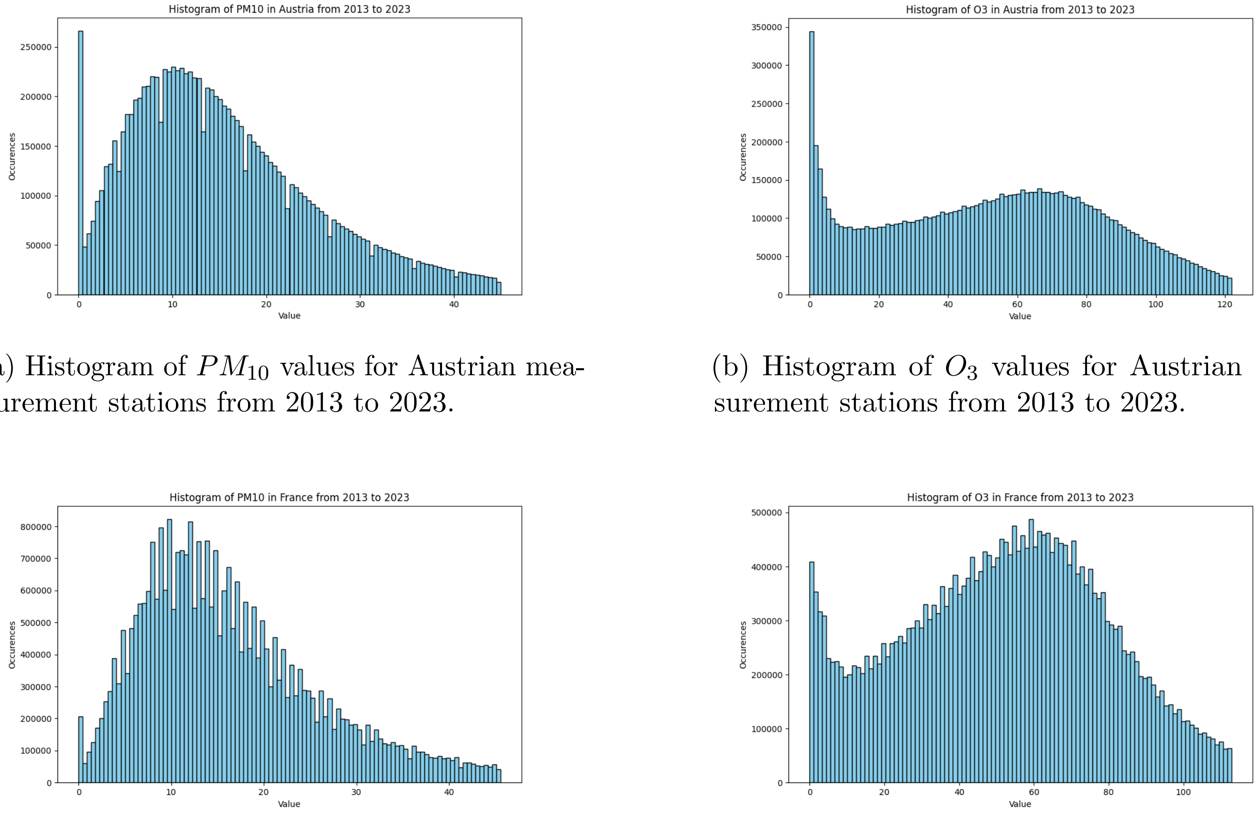
(a) Histogram of average hourly SO_2 concentrations for German measurement stations from 2013 to 2023.

(b) Line plot of hourly SO_2 concentrations for German measurement stations from 2013 to 2023.

Figure 5: Charts describing hourly SO_2 concentrations in Germany.

3.5 Comparing countries

If measurements throughout different countries in Europe show different patterns, it would become necessary to train models on data that is sampled from multiple countries. This would be needed to be able to drawn an overall conclusion on how well the models perform on European outdoor air quality. If data however turns out to be similar throughout Europe, it might be sufficient to only use data from one country. Data exploration has shown that Germany has the largest amount of measuring stations with a large diversity in station types and areas. These characteristics make Germany the most likely country to deliver the most representative results in the situation that a model should be trained on data from a single country. To be able to validate that Germany is representative for European data in general, data from France and Austria was also analyzed as these are also countries with a large amount of stations in multiple types of locations. Data from these countries show similar distributions to the data that is collected in Germany for both O_3 and PM_{10} as shown in Figure 6. Note that the size of the dataset for different countries varies. This results in some values occurring more frequently in some countries than others as seen in Figure 6. Even though the values on the y-axis are different for each chart, the general distribution remains similar for the different countries. For SO_2 the histograms once again seem to indicate that certain values occur more frequently than might be expected which is possibly caused by missing values in these time series. Therefore it is hard to judge whether the distributions of this pollutant show similarities across countries.



(a) Histogram of PM_{10} values for Austrian measurement stations from 2013 to 2023.

(b) Histogram of O_3 values for Austrian measurement stations from 2013 to 2023.

(c) Histogram of PM_{10} values for French measurement stations from 2013 to 2023.

(d) Histogram of O_3 values for French measurement stations from 2013 to 2023.

Figure 6: Charts describing hourly O_3 and PM_{10} concentrations in France and Austria.

4 Anomaly detection

4.1 Data exploration results

From all the pollutants that are being measured by the measuring stations, PM_{10} and O_3 have shown to be the most interesting for anomaly detection due to not only their impact on human health, but also the characteristics of these time series. The histogram of hourly PM_{10} values tends to follow a normal distribution that is slightly skewed to the left and its line plot does not show any general trend with values staying around the mean. The line plot, however, also makes clear that this data contains point anomalies as there are sudden increases and decreases in the data. A global point outlier that should be detectable by the models can be observed on new years eve. Within such an unlabeled dataset, new years eve would be the easiest to label as an anomaly as it clearly has a higher value than any other data points. While PM_{10} does not exhibit a clear seasonal pattern, O_3 clearly does. Because of the seasonality of this time series, it will be more interesting to focus on contextual anomalies instead of global point outliers as a high value in the winter months might not be a global outlier and vice versa.

The data exploration has also shown that the data from German measuring stations generally follows a similar pattern to other countries. As Germany has measurement stations of all different types and in all areas and altitudes, applying anomaly detection algorithms on data from these stations will likely give representative results for European air quality in general. It will also be interesting to compare model performance on data from different areas and station types as these measuring station characteristics have shown to have a clear impact on the measured concentrations. For these reasons the models will only be applied to data from Germany as this would likely give a good overview of their general performance on European air quality data. The impact of various measuring station characteristics on model performance will also be examined.

4.2 Model selection

Data exploration has lead to a selection of data that is most suitable for anomaly detection. However, now that this step has been completed, a selection of anomaly detection models still needs to be made. For the selection of these models a python library called Darts [HLP+22] will be used. This library contains numerous models that can be used for anomaly detection on time series data. The five selected models from the available library are listed in Table 2. The models in this table have also been categorized according to a categorization as suggested by Braei and Wagner [BW20]. They mention that generally, statistical approaches assume that the data is generated by a specific statistical model while machine learning/deep learning methods consider the data generation process as a black box and try to learn from the data only.

To obtain some sort of a baseline performance, the first model that will be used is Exponential Smoothing [Gar06]. Due to its extensive documentation, this model is relatively simple to implement and should give an impression whether it is possible to detect anomalies in the dataset. The model in this library is a wrapper around the Holt-Winters' Exponential Smoothing method as found in the Statsmodels library [SP10]. Holt-Winters' Exponential Smoothing is also able to take trends

Model	**Category**
Exponential Smoothing	Statistical
ARIMA	Statistical
Linear Regression Model	Machine Learning
RNN Model	Deep Learning
Regression Ensemble Model	Deep Learning

Table 2: List of the used anomaly detection models.

and seasonality into account which possibly increases the accuracy compared to the more basic version.

The second model that will be compared is the ARIMA [ACADL18] model which is a more modern statistical method compared to Exponential Smoothing. This model should hopefully be more accurate on complex time series like the air quality data that is used here. Numerous hyperparameters can be selected for the model including seasonal parameters which results in the seasonal ARIMA (SARIMA) model. Due to this property, ARIMA could be a good option for application on time series both with and without a clear seasonal component. ARIMA should thus be able to give a good indication of how well anomalies can be detected by using only statistical models.

The third selected model is the Linear Regression Model. This is a model that uses linear regression [NK21] of some of the target series' lags to forecast future values. The hyperparameters of this model can also be set to make use of covariate series lags to improve the accuracy of the forecasts. Covariates allow for the model to focus on specific data points that might contain valuable information with regards to the prediction of future data points. This ability differentiates this model from the first two statistical models for which no covariates could be set in the hyperparameters. It will be interesting to observe whether this machine learning model will lead to improved performance in anomaly detection compared to the previous statistical models.

The fourth model that will be used is the Recurrent Neural Network (RNN) Model that makes use of long short-term memory (LSTM) cells [HS97]. The use of LSTM cells in RNN's helps with solving the vanishing gradient problem, making them capable of learning long-term dependencies [Cal20]. This model can also be applied without LSTM cells, but due to LSTM's use on indoor air quality time series and the successful results there, the decision was made to apply this RNN model with such cells. Just like the Linear Regression Model, this model can make use of covariates. It is also the first global forecasting model as it needs to be trained on similar time series in order to accurately forecast a series.

The final selected model is the Regression Ensemble Model which uses a regression model for ensembling individual models their predictions using the stacking technique [Wol92]. This model thus uses a regression model to optimally combine the forecasts of multiple base models. The base models can be set in the hyperparameters and in the case that all individual models can make use of covariates, these can also be passed to these selected models. This model is selected to be able to observe whether an ensemble model might perform better than standalone models like the previous four. The next paragraphs of this chapter will go into further detail about how these models will be evaluated and how they have been implemented.

4.3 Model evaluation

This chapter describes the steps that need to be taken to be able to evaluate the performance of the different models. To be able to compare them, two important steps need to be taken. Namely, a ground truth of the dataset needs to be acquired and promising hyperparameter settings for the models need to be found.

4.3.1 Establishing Ground Truth in Unlabeled Time Series

Due to the absence of ground-truth labels in the air quality dataset, a direct comparison between the detected anomalies and true anomalies is not feasible. To be able to evaluate these five models, a solution thus needs to be found if we want to determine which type of model performs better. There are multiple possibilities to obtain a ground truth for an unlabeled time series like this one however. One possibility is to manually label the existing data. Due to the dataset being very large however, this would be very time consuming and thus not practical for this thesis. Better options are to mark values that reach a certain threshold as anomalies, for example values that are above certain health standards, or to inject synthetic anomalies into the time series.

This first option would likely make the application of anomaly detection models redundant as it would already be clear in this case when a measurement is an anomaly, namely when the threshold is reached. The second option, of injecting synthetic anomalies, thus looks like the better option as this will show whether the models will be able to detect anomalies or not. Another possibility might be to combine both options by evaluating the models on time series where all values are below the threshold of a health standard, but where synthetic anomalies have been injected that are above this threshold. This can potentially be achieved by replacing data points that are not anomalous with a point that has a high value that is above this threshold.

4.3.2 Model Calibration and Hyperparameter Selection

Before anomalies can be detected, it should first be determined whether a model is accurately forecasting a time series. If it is known that a model can forecast a non-anomalous time series with complete accuracy, an anomalous data point can easily be detected. Only anomalous data points would in such a situation have a value that is different than the forecasted value for the same data point. In a real world situation however, it is very difficult to obtain a completely accurate forecast, but an attempt does need to be made to obtain forecasts that are as accurate as possible. The less accurate a forecast is, the more difficult it will be to determine whether a data point is anomalous or if the forecasts is inaccurate instead. To determine the accuracy of forecasting models, the Mean Squared Error (MSE) of forecasts on non-anomalous time series can be calculated [IA24]. A lower MSE indicates that the values of the target series and the forecasted series are close to each other, while high values indicate the opposite.

To determine whether our models deliver accurate results, the models are thus first applied to a non-anomalous time series. Once a forecast is obtained on this time series, the mean squared error (MSE) is calculated by comparing the predicted values to the actual values from the target series. In the case of PM_{10} air quality, the values measured at measuring point *SPO.DE_DEHE024_PM1_dataGroup1* from 17/10/2019 to 16/3/2020 were used to evaluate the accuracy of the forecasts. This series was selected as non-anomalous due to its low variance and lack of sudden spikes in values. In order to find optimal hyperparameter values for each model, multiple configurations are tested on this series. The set of values that yields the lowest MSE is then selected for use in anomaly detection and comparison with the other models. The optimization of hyperparameters is a complicated research area of itself (especially in an unsupervised learning) [AAS+22][ZRA21][LWvL25] and outside of the scope of this thesis. Therefore the usage of MSE will most likely not produce optimal models, however for this use case it should be sufficient as it does give an indication of how accurate the forecasts are.

For O_3 a non-anomalous time series also needs to be selected. Due to O_3 data visibly having a higher variance than PM_{10} it is challenging to determine what constitutes as non-anomalous. The decision was made to use the values measured at measuring point *SPO.DE_DEHE028_O3_dataGroup1* as this point is found in a rural area and is of the background type. Such measuring stations generally see lower values and less sudden changes due to a lack of traffic and/or industry affecting the measurements. The selected PM_{10} series was also chosen due to its measuring station being of this type and area. For O_3, multiple measuring stations were explored from which a series was chosen that looked to follow the most regular pattern without significant outliers. The selected hyperparameters for PM_{10} series were also applied to O_3 series to determine whether they also provided accurate forecasts on a different pollutant.

4.3.3 Synthetic Anomaly Injection and Evaluation Framework

Once hyperparameters have been found that give accurate results in the forecasting of a non-anomalous time series, the next step is to inject anomalies into the time series. Injecting anomalies entails that values are inserted into a time series at a position where they normally would not be expected. To be able to achieve accurate results it is important to inject realistic values as anomalies [WK23] as the results would otherwise not represent real world situations. To test whether the models can actually be used for the detection of anomalies, the first step is to select certain data points at random from a time series with high average concentrations of pollutants. These randomly selected data points are then placed at the corresponding position in the non-anomalous time series. Every data point is subsequently assigned an anomaly score by using the below equation. This equation calculates the squared error for a data point by squaring the difference between the forecasted and actual value, to subsequently divide it by the average squared error of all data points in the target series.

$$s_i = \frac{(\hat{x}_i - x_i)^2}{\frac{1}{N}\sum_{j=1}^{N}(\hat{x}_j - x_j)^2},$$

where the notation is as follows:

s_i: Anomaly score at data point i.

$\hat{x}_i$: Forecasted value at data point i.
x_i: Actual value at data point i.
N: The amount of data points in the time series for which an anomaly score is calculated.
j: The current data point over which the sum is iterating.

By injecting anomalies from a time series with high average concentrations into a non-anomalous time series like the one from *SPO.DE_DEHE024_PM1_dataGroup1*, the resulting anomalies should receive significantly higher anomaly scores than the non-anomalous sections. Because a ground truth has been created by injecting these anomalies, a confusion matrix can now be used to evaluate a models performance. To create such a confusion matrix a threshold needs to be set for the minimum anomaly score for which a data point is classified as an anomaly. Setting this threshold too low might result in false positives, while setting it too high might results in false negatives. Once a sufficient threshold has been determined by attempting to optimize the amount of true positives and false positives that models achieve, the confusion matrix can be used to calculate a selection of evaluation metrics. After initially testing the models on extreme anomalies, such as injecting data points from an urban measuring station into data from a rural station, they should also be tested on more subtle, realistic, anomalies. By for example injecting data points from measuring stations with a similar type and area it is possible to achieve more realistic anomalies and thus more realistic results.

4.3.4 Evaluation Metrics

Once a confusion matrix has thus been obtained by comparing the ground truth to the detected anomalies, the selected evaluation metrics need to be calculated. For the evaluation of the models, the following three evaluation metrics have been chosen: precision, recall and F1-score. Precision describes the amount of true positives compared to the total amount of positives. This metric will show lower values when a lot of false positives are detected. False positives in this case are data points that are not an injected anomaly, but are detected as one. The recall value indicates the amount of true positives compared to the total amount of true positives and false negatives. In this situation, the metric shows how many of the injected anomalies have actually been detected. The F1-score is used to give an overall indication of a models performance by combining the precision and recall scores. The combination of these evaluation metrics should give an impression of how well these models are able to detect anomalies in such an air quality dataset. The formulas that are shown below, show how the performance metrics are calculated using the values from the confusion matrix.

$$\textit{Precision} = \textit{True Positives}/(\textit{True Positives} + \textit{False Positives})$$

$$\textit{Recall} = \textit{True Positives}/(\textit{True Positives} + \textit{False Negatives})$$

$$\textit{F1-score} = 2 * \textit{True Positives}/(2 * \textit{True Positives} + \textit{False Positives} + \textit{False Negatives})$$

Based on these metrics it might be possible to come to a conclusion on which type of model performs optimally on European outdoor air quality data.

4.4 Model implementation

This section provides a detailed description of the five selected models. For each model, the forecasting process for a given target series is explained, along with the hyperparameters that can be configured. The optimal hyperparameter settings for each model are also presented.

4.4.1 Exponential smoothing

The first anomaly detection method to implement is the Exponential Smoothing [Gar06] forecasting model from the Darts library. Exponential smoothing is a statistical method that assigns weights to previous data points to be able to forecast future points, with more recent values being given higher weights. As mentioned earlier, this model is a wrapper around the Holt-Winters' Exponential Smoothing method as found in the Statsmodel library [SP10]. This model is able to take trends and seasonality into account which improves the accuracy of the model on the air quality time series compared to a model that does not have this ability. After testing multiple hyperparameter configurations, the model that achieved the highest accuracy used the settings shown in Table 3. Formula 1 to 4, as shown below, are the ones used by Exponential Smoothing for a damped additive trend and additive seasonality, as is the case with the selected hyperparameters.

$$S_t = S_{t-1} + \phi T_{t-1} + \alpha e_t, \tag{1}$$

$$T_t = \phi T_{t-1} + \alpha\gamma e_t, \tag{2}$$

$$I_t = I_{t-p} + \delta(1-\alpha)e_t, \tag{3}$$

$$\hat{X}_t(m) = S_t + \sum_{i=1}^{m} \phi^i T_t + I_{t-p+m}, \tag{4}$$

where the notation is as follows:

α: Smoothing parameter for the level of the series.
γ: Smoothing parameter for the trend.
δ: Smoothing parameter for seasonal indices.
ϕ: Autoregressive or damping parameter.
S_t: Smoothed level of the series, computed after X_t is observed. Also the expected value of the data at the end of period t in some models.
T_t: Smoothed additive trend at the end of period t.
I_t: Smoothed seasonal index at the end of period t.
X_t: Observed value of the time series in period t.
e_t: One-step-ahead forecast error. $e_t = X_t - \hat{X}_{t-1}$
m: Number of periods in the forecast lead-time.
p: Number of periods in the seasonal cycle.
$\hat{X}_t(m)$: Forecast for m periods ahead from origin t.

With these settings the model has a sliding window of 168 data points (hours) which it uses to predict the amount of data points of the forecast horizon. After every iteration the sliding window moves ahead by 1 hour and performs the calculation again on the data points that are in the sliding window. As one might notice, this means that most data points will find themselves in the forecast horizon for multiple subsequent iterations in case of a forecast horizon that is larger than 1. As only 1 value needs to remain for the final forecast, an average of these forecasted values is taken after the sliding window has reached the end of the time series. Applying the model with these settings, on the non-anomalous PM_{10} validation series, results in an MSE of 1.66 which looks to be the optimal result for this model and time series.

The O_3 concentration time series exhibits distinct characteristics, most notably a clear yearly seasonal pattern, which is not present in the PM_{10} series. There also appear to be more sudden changes in values between subsequent data points. Due to this difference between both pollutants, an MSE of 26.99 was achieved for O_3 concentrations using the same hyperparameters as shown in Table 3

Parameter	**Value**
`trend`	`ADDITIVE`
`seasonal`	`ADDITIVE`
`seasonal_periods`	`24`
`damped`	`True`
`window_size`	`168`
`forecast_horizon`	`1`

Table 3: Selected hyperparameter configuration for applying the exponential smoothing model on a PM_{10} and O_3 time series.

4.4.2 ARIMA

ARIMA [Gar06], AutoRegressive Integrated Moving Average, is a more modern statistical method. With seasonal ARIMA (SARIMA) it is also possible to forecast time series with a seasonal component as is for example the case in O_3 air quality data. The ARIMA implementation in the Darts library has four required parameters of which the optimal settings are shown in Table 4. The parameter p sets the number of time lags of the autoregressive model (AR), d sets the order of differentiation (I) and q sets the size of the moving average window (MA). The main part of the ARIMA model combines AR and MA polynomials into a complex polynomial [ACADL18] using the following equation:

$$y_t = \mu + \sum_{i=1}^{p}(\sigma y_{t-1}) + \sum_{i-1}^{q}(\Theta \varepsilon_{t-1}) + \varepsilon_t,$$

where the notation is as follows:

μ: The mean value of the time series data.
p: The number of autoregressive lags.
σ: Autoregressive coefficients (AR).
q: The number of lags of the moving average process.
Θ: Moving average coefficients (MA).
ε: The white noise of the time series data.
d: The number of differences calculated from: $\Delta y_t = y_t - y_{t-1}$.

In the case that the model is applied to a time series with a seasonal component, the hyperparameter 'seasonal_order' can also be set with once again the p, d and q hyperparameters. The length of the seasonal component can also be indicated here, but was left out because it caused the maximum likelihood optimization to fail for the PM_{10} time series data. For an optimal comparison between the different models, the window size and forecast horizon have been set to the same value as this has a clear influence on the accuracy of the forecasts. Using the below settings leads to an MSE of 1.43 on the non-anomalous PM_{10} time series which is slightly lower than what could be achieved by using exponential smoothing. With the same hyperparameter configuration, an MSE of 25.04 could be achieved when applying the model on O_3 time series.

Parameter	**Value**
p	12
d	1
q	0
`seasonal`	`(0,0,0,0)`
`window_size`	`168`
`forecast_horizon`	`1`

Table 4: Selected hyperparameter configuration for applying the ARIMA model on PM_{10} and O_3 time series.

4.4.3 Linear Regression Model

While the previous models are categorized as statistical models in Darts, the linear regression model is classified under the regression models category. In the simplest case of linear regression, the model allows for a linear relationship between the forecast variable y and a single predictor variable x [HA21]:

$$y_t = \beta_0 + \beta_1 x_t + \varepsilon_t.$$

β_0 and β_1 denote the intercept and slope of the line respectively in this equation. The above equation can be extended by including more lagged values of the predictors. Table 5 shows the selected hyperparameter configuration for this model for application on PM_{10} time series. The lags parameter defines which past values are used to predict the next data point. This hyperparameter can simply be set to include a certain amount of the most recent points or it can be set as a list of covariates as is the case here. The fact that this model can make use of covariates also differentiates it from the previous two models which did not have such an option. Data points from both 1 and 2 hours ago are now taken into account as well as values from exactly 1 and 2 days ago. By taking values from exactly 1 and 2 days ago into account, it might be possible to capture any longer patterns in the data that span over 1 or 2 days. Due to the large amount of possible settings for this hyperparameter it is likely that there are more promising settings to be found, but the selected setting performed quite well on the non-anomalous data. The size of the sliding window has once again been set to 168 data points, one week, for a fair comparison with the first two models. The ‘output_chunk_length’ has been set equal to the ‘forecast_horizon’ as only one data point is needed per iteration as output in this case. The selected hyperparameter settings resulted in an MSE of 1.34 on the non-anomalous PM_{10} time series and thus resulted in a more accurate forecast than the previous models. With this hyperparameter configuration, an MSE of 23.91 could be achieved by applying the model on the non-anomalous O_3 time series.

Parameter	**Value**
`lags`	`[-1, -2, -24, -48]`
`output_chunk_length`	`1`
`window_size`	`168`
`forecast_horizon`	`1`

Table 5: Selected hyperparameter configuration for applying the linear regression model on PM_{10} and O_3 time series.

4.4.4 RNN Model

The Recurrent Neural Network model in the Darts library allows for the setting of three different module types with the model parameter which can be set to use “RNN”, “LSTM” or “GRU”. Given the promising results of LSTM-AE models in anomaly detection for indoor air quality [WJJX+23], and the ability of LSTM cells to capture long-term dependencies, the LSTM module type was chosen for this study. An LSTM cell consists of three types of gates: an input gate, an output gate and a forget gate, as shown in Figure 7. These gates enable the cell to retain information over arbitrary time intervals [HS97].

The main purpose of the forget gate is to decide which bits of the cell state are useful given both the previous hidden state and new input data [WJJX+23]. The forget gate makes use of the following equation:

$$f_t = \sigma(w_f[H_{t-1}, X_t] + b_f). \tag{5}$$

The input gate is needed to protect the memory contents stored in a unit from perturbation by irrelevant inputs. It checks whether new information should be kept in the cell state and what new information should be added. The following three equations are used by this gate to determine this:

$$\tilde{C}_t = \tanh(w_c[H_{t-1}, X_t] + b_c), \tag{6}$$

$$i_t = \sigma(w_i[H_{t-1}, X_t] + b_i), \tag{7}$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t. \tag{8}$$

The output gate on the other hand is used to protect other units from perturbation by currently irrelevant memory contents that are stored in its own cell. The following equations are used in this step:

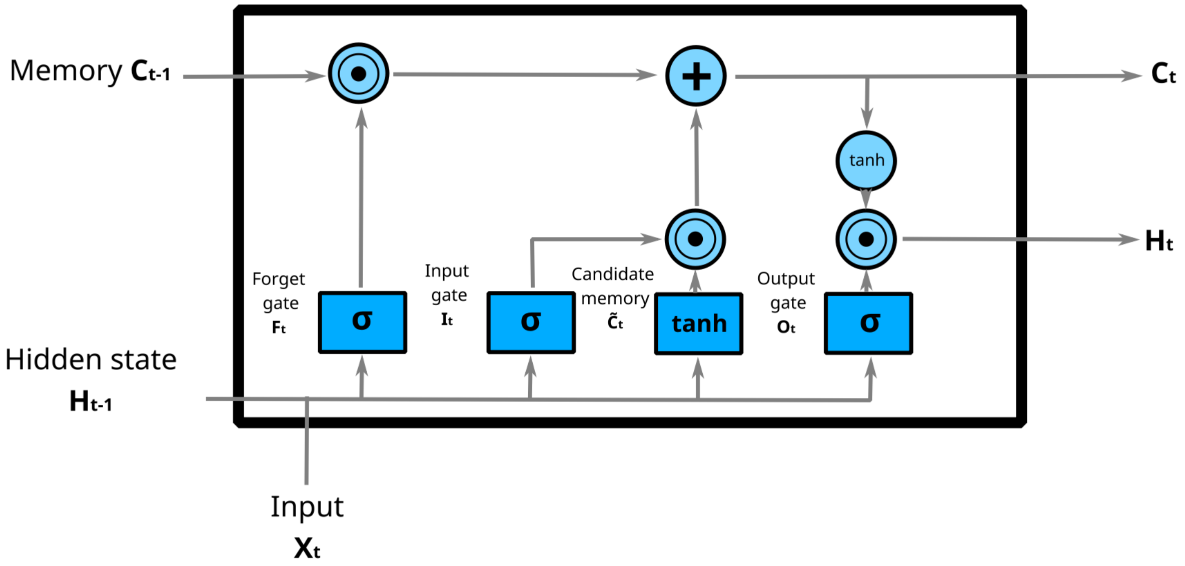


Figure 7: Illustration of an LSTM cell, adapted from: Zhang et al. [ZLLS23]

$$o_t = \sigma(w_o[H_{t-1}, X_t] + b_o), \tag{9}$$

$$H_t = o_t \odot \tanh(C_t), \tag{10}$$

where the notation for the above equations is as follows:

f_t: The result from the forget gate.
i_t: The result from the input gate.
o_t: The result from the output gate.
σ: The activation function.
$tanh$: The tanh activation function that is used.
w_x: Weights of the gate.
b_x: Bias of the gate.
H_{t-1}: Concatenation of the hidden state.
X_t: Current input.
$\tilde{C}_t$: Amount of new information.

This model has a large amount of hyperparameters that can be set. The optimal settings that were found resulted in an MSE of 1.27 on the non-anomalous PM_{10} time series and an MSE of 28.77 on the non-anomalous O_3 series. Once again, the large amount of hyperparameters makes it likely that there are settings to be found that deliver better performance. The used settings can be found in Table 6. For a fair comparison with the previous models the 'input_chunk_length' is set to the same value as the window size for the previous models, namely 168. The training set for the model consists of all time series in the dataset that span the same time frame as the target series. Only the data that has been collected from measuring stations with the same type and area have been used for training due to the impact that these characteristics have on the data. The data exploration in Section 3 clearly shows that the characteristics of these measuring stations have a significant impact on the shape of the data. Due to some types and areas being more prevalent, the size of the training set changes depending on the type and area of the target series. This difference in training set size can have a potential impact on the performance of model.

Parameter	**Value**
`model`	`LSTM`
`input_chunk_length`	`168`
`training_length`	`336`
`n_epochs`	`20`
`hidden_dim`	`50`
`n_rnn_layers`	`3`
`dropout`	`0.2`
`random_state`	`13`
`add_encoders`	`‘cyclic’: ‘future’: [‘dayofweek’, ‘month’, ‘hour’]`
`retrain`	`3`
`last_points_only`	`True`
`forecast_horizon`	`1`

Table 6: Selected hyperparameter configuration for applying the RNN model on PM_{10} and O_3 time series.

4.4.5 Regression Ensemble Model

The regression ensemble model employs a regression algorithm to combine the forecasts of individual models using the stacking technique [Wol92]. The selected hyperparameter configuration shown in Table 7 resulted in an MSE of 1.91 on the non-anomalous PM_{10} time series which is larger than all the other models. An MSE of 31.00 was achieved on the non-anomalous O_3 time series. Two models have been selected for this ensemble model, one RNN model with default cells and the RNN model with LSTM cells as described in Chapter 4.4.4. A Random Forest Regressor is subsequently applied to ensemble these two individual models. A random forest [Bre01] fits multiple decision trees on various subsamples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting [PVG+11]. Using this regression model, it is possible to improve the accuracy of an individual model such as the RNN model that used LSTM cells. The hyperparameter configuration for the individual LSTM model is identical to the one shown in Table 6, with only the model hyperparameter changed to ‘RNN’ for the standard RNN model. The training set for both models is identical to that used for the individual RNN model in Chapter 4.4.4. It therefore consists of identical time frames that have been collected from measuring stations that share the same area and type as the measuring station of the target series.

Parameter	**Value**
`forecasting_model`	`[model_rnn_lstm, model_rnn]`
`regression_train_n_points`	`84`

`regression_model`	`RandomForestRegressor(n_estimators=100,`
	`random_state=42)`

`train_forecasting_models`	`false`
`train_using_historical_forecasts`	`True`
`forecast_horizon`	`1`

Table 7: Selected hyperparameter configuration for applying the ensemble model on PM_{10} and O_3 time series.

5 Results

This chapter will show the results that have been obtained by applying all the models to time series that have been selected from the dataset. The chapter has been divided into three different sections. The first two sections cover the results on PM_{10} and O_3 data while the final section provides a general overview of the results. Both Sections 5.1 and 5.2 are split into five different subsections. The first subsection shows a selection of time series that will be used to evaluate the performance of the models. The remaining subsections will show the actual results that have been achieved by applying the models to these selected series.

At first a look will be taken at the performance of these models on unrealistic anomalies, anomalies that are not likely to occur in natural situations. It would be expected that such anomalies are the easiest to detect due to them being characterised by more extreme values. After applying the models on such unrealistic anomalies, they also need to be applied to realistic anomalies to get an impression of their performance in realistic situations. The final two subsections will look at the impact of the station area and type characteristics of the measuring stations. This is done by applying the models to data from different measuring stations with realistically injected anomalies. In this way, it is possible to study the influence of station area and type on the models' ability to detect realistically occurring anomalies.

5.1 Results for PM_{10}

Table 8 shows which specific time series have been selected for the evaluation of the models. As can be seen from this table, a selection has been made so that performance can be compared on different types and areas of measuring stations. Beyond comparing performance across different types of measuring stations, the models were also evaluated on various types of injected anomalies. The first step for example injects anomalies by selecting random data points that are above the health standard of 50μg/m^3, from a measuring station of the traffic type in an urban area. As this measuring station sees higher concentrations on average than one of the background type in a rural area, these high values can be randomly injected into this time series. To evaluate the impact of different types and areas, anomalies are injected from a measuring station with the same type and area as to have anomalous values that are as realistic as possible. Table 9 shows the measuring stations that were used to inject anomalies. The injected values were selected from the same time frame as the target series. In the subsequent paragraphs the time series will be referred to by the index that is mentioned in the tables, instead of the full sampling point ID.

Index	Sampling point ID	Area	Type	Timeframe
#1	SPO.DE_DEHE024_PM1_dataGroup1	Rural	Background	17/10/2019 - 16/3/2020
#2	SPO.DE_DEBB021_PM1_dataGroup1	Urban	Background	17/10/2019 - 16/3/2020
#3	SPO.DE_DEBB054_PM1_dataGroup1	Urban	Traffic	17/10/2019 - 16/3/2020
#4	SPO.DE_DEMV031_PM1_dataGroup1	Urban	Industrial	17/10/2019 - 16/3/2020
#5	SPO.DE_DEBB048_PM1_dataGroup1	Suburban	Background	17/10/2019 - 16/3/2020

Table 8: Time series used for evaluation. Anomalies are injected into these series.

Index	Sampling point ID	Area	Type
#6	SPO.DE_DEMV004_PM1_dataGroup1	Rural	Background
#7	SPO.DE_DEBB064_PM1_dataGroup1	Urban	Background
#8	SPO.DE_DEBB110_PM1_dataGroup1	Suburban	Background
#9	SPO.DE_DEHH015_PM1_dataGroup1	Urban	Industrial
#11	SPO.DE_DEBE061_PM1_dataGroup1	Urban	Traffic

Table 9: Time series used for evaluation. Anomalies are selected from these series and injected into a different one.

5.1.1 Injecting unrealistic anomalies

Table 10 shows the results from injecting values from time series #11 into #1. Only values that were above the health standard of $50\mu g/m^3$ were injected as these high values should be clear to detect by the models. In total, 19 different values were injected from series #11 into #1. Figure 8 shows one of the injected anomalies that was given an anomaly score larger than 15 by all five models. These data points, with an anomaly score above 15, are marked as anomalous by the models and subsequently compared to the ground truth. If a data point does not receive an anomaly score that is higher than 15, the point will be marked as normal by a model.

Model	TP	FN	FP	TN	Precision	Recall	F1-Score
Exponential Smoothing	18	1	14	3446	0.562	0.947	0.706
ARIMA	19	0	35	3425	0.352	1.000	0.521
Linear Regression model	11	8	14	3446	0.440	0.579	0.500
RNN model	19	0	9	3451	0.679	1.000	0.809
Regression Ensemble model	19	0	8	3452	0.704	1.000	0.826

Table 10: The resulting confusion matrices of injecting 19 anomalies from time series #11 (urban area, traffic type) into #1 (rural area, background type) with a threshold of 50.

Due to anomalies being undefined in our original time series, it is likely that the models will result in a relatively large number of false positives. As it is not possible to determine whether the "non-anomalous" time series actually does not contain any anomalies before the injection of actual anomalies, it is possible that some of the false positives could actually be classified as anomalies. An attempt was made to select a time series, for time series #1, with as few as possible visible

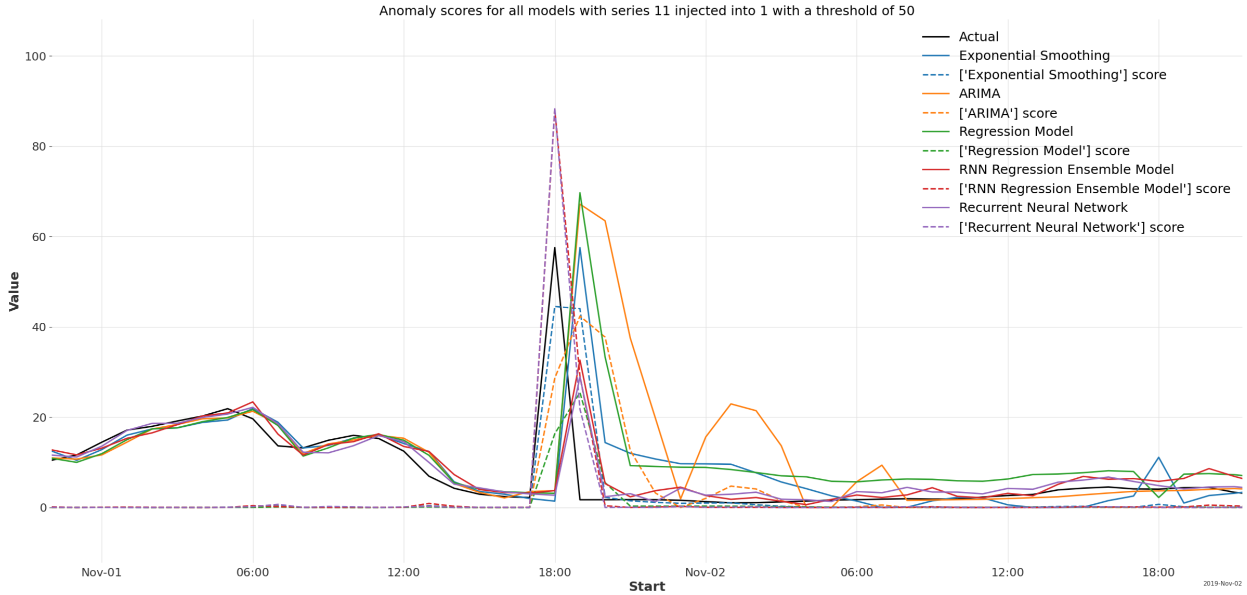


Figure 8: Zoomed in chart of the forecasts and anomaly scores of the injected anomalies from series #11 into series #1. In this chart only one anomaly is visible.

anomalies, but this unfortunately does not guarantee that there are none in the series. Therefore, the recall value gives a better indication of the models performance than precision as this does not take the false positives into account.

The results in Table 10 generally show relatively high recall scores for all the models. Only the linear regression model stands out here with a significantly lower recall score, while still having quite a large amount of false positives. ARIMA does detect all the injected anomalies, but this comes at the caveat of a significantly larger amount of false positives compared to the other four models. Overall, the regression ensemble model has performed the best on such injected unrealistic anomalies with similar scores for the RNN model.

Figure 8 presents the forecasts generated by all models, along with the corresponding anomaly scores assigned to each data point. From this plot it becomes clear why ARIMA has a larger amount of positives. Unlike the RNN, this model is not a global forecasting model. This results in ARIMA taking the anomalous data point into account for the forecasting of future data points. This results in high anomaly scores for data points that should actually be non-anomalous.

5.1.2 Injecting realistic anomalies

As noted by Keogh et al. [WK23] it is important to inject anomalies that could realistically occur within the time series to ensure accurate evaluation of the models. The most effective approach in this case is to inject values that are randomly selected from data collected at a measuring station of similar type and location. Injecting random values from series #6 into #1 without applying any threshold value, as was described in section 5.1.1, would approach realistic anomalies most accurately.

Model	TP	FN	FP	TN	Precision	Recall	F1-Score
Exponential Smoothing	7	12	14	3446	0.333	0.368	0.350
ARIMA	7	12	20	3440	0.259	0.368	0.304
Linear Regression model	7	12	16	3444	0.304	0.368	0.333
RNN model	7	12	22	3438	0.241	0.368	0.292
Regression Ensemble model	7	12	18	3442	0.280	0.368	0.318

Table 11: The resulting confusion matrices and performance metrics of injecting 19 anomalies from time series #6 into #1 (rural area) without a threshold.

The performance metrics of all five models on realistic anomalies are shown in Table 11. As most of the values in both series #1 and #6 will now be more similar, it is likely that fewer of the injected anomalies will actually be detected. As a result, all performance metrics are lower for all models compared to when unrealistic anomalies were injected. All the models generally have a similar performance with the only differences being in the amount of false positives. As earlier determined, the amount of false positives does not necessarily give any information about model performance due to possible anomalies in the original time series.

5.1.3 Impact of station area on model performance

As shown in Chapter 3, the area in which a measuring station is located has a significant impact on the distribution of the data that is collected there. For example, a measuring station located in an urban area is more likely to register high concentrations than one that is located in a rural area. Table 11, Table 12 and Table 13 show the results for all five models on time series with realistically injected anomalies from time series with the same area and type. They respectively show the results for data from rural, urban, and suburban areas. To ensure the injected anomalies are as realistic as possible, the altitude of the selected measuring stations was also matched as closely as the available data allowed.

Model	TP	FN	FP	TN	Precision	Recall	F1-Score
Exponential Smoothing	2	16	12	3450	0.143	0.111	0.125
ARIMA	0	18	10	3452	0.000	0.000	0.000
Linear Regression	0	18	4	3458	0.000	0.000	0.000
RNN Model	5	13	17	3445	0.227	0.278	0.250
Regression Ensemble	4	14	16	3446	0.200	0.222	0.211

Table 12: The resulting confusion matrices and performance metrics of injecting 18 anomalies from time series #7 into #2 (urban area and background type) without a threshold.

As seen in Table 11, the models all performed relatively similar on data from rural areas. This is however not the case for data from both suburban (Table 13) and urban (Table 12) areas. In urban areas,

Model	TP	FN	FP	TN	Precision	Recall	F1-Score
Exponential Smoothing	0	18	10	3452	0.000	0.000	0.000
ARIMA	0	18	9	3453	0.000	0.000	0.000
Linear Regression	0	18	4	3458	0.000	0.000	0.000
RNN Model	5	13	26	3436	0.161	0.278	0.204
Regression Ensemble	1	17	15	3447	0.062	0.056	0.059

Table 13: The resulting confusion matrices and performance metrics of injecting 18 anomalies from time series #8 into #5 (suburban area) without a threshold.

both the RNN and regression ensemble model give the best results with only one anomalous data point not being detected by the ensemble model compared to the RNN model. The statistical and machine learning models perform significantly worse here than the two deep learning models. Only exponential smoothing manages to detect any of the injected anomalies. It is also noticeable that the amount of false positives increases together with the amount of true positives. This might suggest that some models generally award higher anomaly scores for data points and thus show better results.

In suburban areas, the RNN model once again shows the most promising results, with only the regression ensemble model also being able to detect any of the injected anomalies as well. This data shows similar patterns as the results for urban areas, with the amount of true positives once again seeming related to the amount of true negatives. Notably, the number of false positives remains similar for the regression ensemble model, despite a reduction in true positives within the suburban data.

5.1.4 Impact of station type on model performance

Just like the area of a measuring station, its type also influences the distribution of the data. For that reason the models have also been applied on time series from all different station types with other variables of the stations being kept similar.

Model	TP	FN	FP	TN	Precision	Recall	F1-Score
Exponential Smoothing	0	19	10	3451	0.000	0.000	0.000
ARIMA	0	19	9	3452	0.000	0.000	0.000
Linear Regression	0	19	9	3452	0.000	0.000	0.000
RNN Model	3	16	26	3435	0.103	0.158	0.125
Regression Ensemble	0	19	12	3449	0.000	0.000	0.000

Table 14: The resulting confusion matrices and performance metrics of injecting 19 anomalies from time series #11 into #3 (traffic type) without a threshold.

Table 14 shows that for data from stations of the traffic type, only the RNN model was able to detect some of the injected anomalies. Similar to what was the case in the comparison of different areas, the amount of true positives seem to correlate with the amount of false positives. As discussed in Chapter 5.1.3, Table 12 did show better results for data from measuring stations of the background

Model	**TP**	**FN**	**FP**	**TN**	**Precision**	**Recall**	**F1-Score**
Exponential Smoothing	4	16	19	3441	0.174	0.200	0.186
ARIMA	4	16	25	3435	0.138	0.200	0.163
Linear Regression	5	15	22	3438	0.185	0.250	0.213
RNN Model	5	15	22	3438	0.185	0.250	0.213
Regression Ensemble	1	19	32	3428	0.030	0.050	0.038

Table 15: The resulting confusion matrices and performance metrics of injecting 20 anomalies from time series #9 into #4 (industrial type) without a threshold.

type, with three of the models detecting more of the injected anomalies. However, both the ARIMA and linear regression model remained at 0 true positives. The results for the industrial station type, as shown in Table 15, generally show better results. All the models are able to at least detect some of the injected anomalies here, with only the regression ensemble model not being able to detect four or more. The linear regression and RNN model perform the best here with exactly the same performance metrics. Both statistical models also perform similarly with the only differences being in the amount of false positives which are slightly higher for ARIMA. Most notable here are the results for the regression ensemble model as this model was only able to detect a single injected anomaly, even though this model did have the largest amount of false positives. Figure 9 shows a forecast of the regression ensemble model on a time series where 20 anomalies have been injected into series #4 from time series #9. This figure shows multiple spikes in the forecasted values where such spikes do not occur in the original target series. This causes a large amount of false positives compared to the other models.

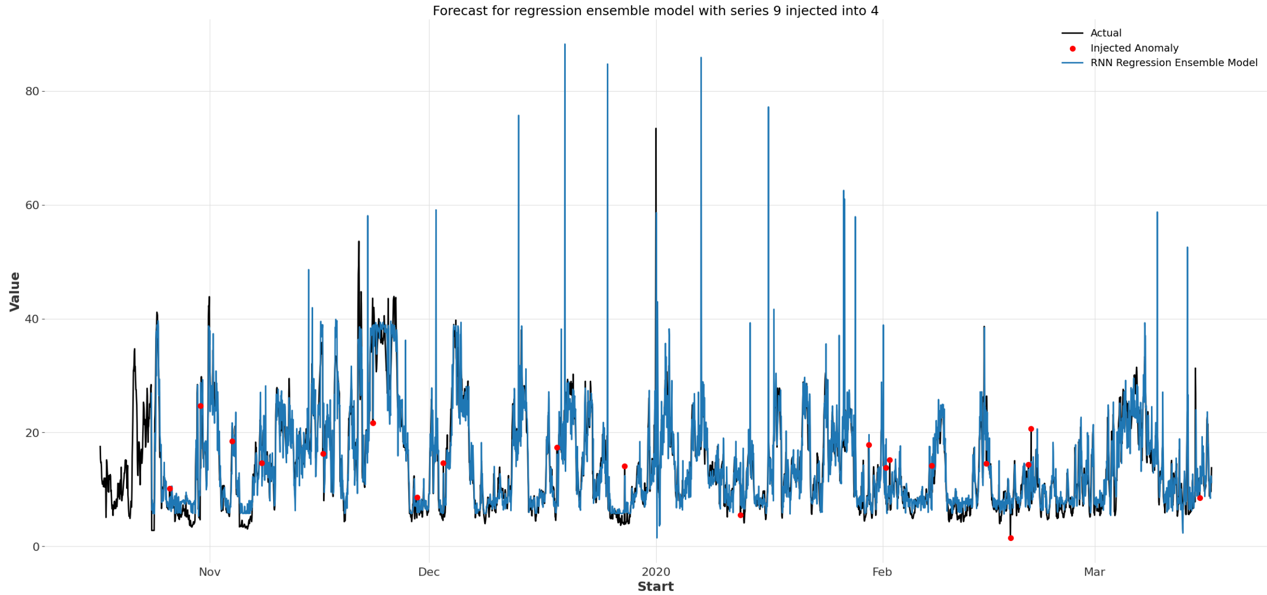


Figure 9: Chart of a forecast by the regression ensemble model on a time series where 20 anomalies have been injected from time series #9 into #4.

5.2 Results for O_3

To compare the models performance on time series data for O_3 concentrations, a similar type of evaluation was done. Table 16 shows the time series which anomalies have been injected into while Table 17 shows the time series from which anomalies were selected. Just as was the case for the evaluation of the models on PM_{10} time series, the models are first compared on unrealistic anomalies. However, since O_3 has a seasonal component, selecting values above a certain threshold based on health standards would likely make anomalies harder to detect depending on the season. Therefore, the selected anomalies were selected purely at random and without a threshold for this first comparison of the models. The same evaluation metrics based on the confusion matrices were used again for this evaluation on O_3 concentrations.

Index	Sampling point ID	Area	Type	Timeframe
#12	SPO.DE_DEHE028_O3_dataGroup1	Rural	Background	17/10/2019 - 16/3/2020
#13	SPO.DE_DEBB007_O3_dataGroup1	Suburban	Background	17/10/2019 - 16/3/2020
#14	SPO.DE_DEBB021_O3_dataGroup1	Urban	Background	17/10/2019 - 16/3/2020
#15	SPO.DE_DENW021_O3_dataGroup1	Urban	Industrial	17/10/2019 - 16/3/2020
#16	SPO.DE_DEBE065_O3_dataGroup1	Urban	Traffic	17/10/2019 - 16/3/2020

Table 16: Time series used for evaluation. Anomalies are injected in these series.

Index	Sampling point ID	Area	Type
#17	SPO.DE_DEHE042_O3_dataGroup1	Rural	Background
#18	SPO.DE_DEBB055_O3_dataGroup1	Suburban	Background
#19	SPO.DE_DEBB064_O3_dataGroup1	Urban	Background
#20	SPO.DE_DENW034_O3_dataGroup1	Urban	Industrial
#21	SPO.DE_DERP023_O3_dataGroup1	Urban	Traffic

Table 17: Time series used for evaluation. Anomalies are selected from these series and injected into a different one.

5.2.1 Injecting unrealistic anomalies

By injecting anomalies from time series #21 into #12, values from a series with higher average values will be injected into one with lower average values. In contrast to how these anomalies were injected for the evaluation of the models on PM_{10} time series, no threshold based on health standards is now taken into account due to O_3 its seasonality. This seasonality would cause the threshold value to be less anomalous for certain seasons. By selecting values purely at random from series #21, unrealistic anomalous values could be higher or lower than expected instead of only larger than the expected value. The confusion matrices that result from these unrealistic anomaly injections are shown in Table 18.

Model	TP	FN	FP	TN	Precision	Recall	F1-Score
Exponential Smoothing	8	12	17	3443	0.320	0.400	0.356
ARIMA	8	12	16	3444	0.333	0.400	0.364
Linear Regression model	9	11	20	3440	0.310	0.450	0.367
RNN model	9	11	14	3446	0.391	0.450	0.419
Regression Ensemble model	8	12	16	3444	0.333	0.400	0.364

Table 18: The resulting confusion matrices of injecting 20 anomalies from time series #21 into #12 without a threshold.

This table shows that the models perform relatively similar on these unrealistic anomalies. In contrast to the unrealistic anomalies for PM_{10} series, the linear regression model does not perform significantly worse than the other four models here. Overall, the models do perform a lot worse than with the unrealistic PM_{10} anomalies which can possibly be explained by the lack of threshold value.

5.2.2 Injecting realistic anomalies

To get an accurate depiction of how the models perform on more realistic data, random data points were selected from series with a similar area and type and injected into the same series that was used for injecting unrealistic anomalies, namely series #12. The results that all models achieved on a time series where 20 data points from series #17 where injected into series #12 are shown in Table 19.

Model	**TP**	**FN**	**FP**	**TN**	**Precision**	**Recall**	**F1-Score**
Exponential Smoothing	14	6	22	3438	0.389	0.700	0.500
ARIMA	14	6	24	3436	0.368	0.700	0.483
Linear Regression model	14	6	25	3435	0.359	0.700	0.475
RNN model	14	6	16	3444	0.467	0.700	0.560
Regression Ensemble model	14	6	12	3448	0.538	0.700	0.609

Table 19: The resulting confusion matrices of injecting 20 anomalies from time series #17 into #12 (rural area) without a threshold.

There is a clear difference in model accuracy between realistic and unrealistic anomalies for both the PM_{10} and O_3 time series, but in opposite directions: unrealistic anomalies are easier to detect in PM_{10} data, whereas realistic anomalies are detected more accurately in O_3 data. All models perform quite well here with all of them detecting 14 of the 20 injected anomalies. The only difference is found in the amount of false positives which are clearly lower for the deep learning models compared to the other three models.

The fact that the models detected more of the realistic anomalies is an unexpected result. However, some of the characteristics of O_3 data might explain this result, as time series for this pollutant show significantly more variance in subsequent data points. This characteristic creates a larger window of possible values that might not be marked as anomalous. There also seems to be a larger difference between time series from similarly classified measuring stations compared to those for PM_{10} time series. Combined with the lack of a threshold for unrealistic anomalies due to O_3 its seasonality, these characteristics increase the difficulty of injecting unrealistic anomalies. This is thus also likely to be the reason for the unexpected result where realistic anomalies were more accurately detected by all the models.

5.2.3 Impact of station area on model performance

Chapter 3 has shown that the area of measuring stations has a significant impact on the distribution of the collected data. This is not only the case for PM_{10} data, but also for O_3 data. For that reason the models should also be evaluated on O_3 data from different areas and types of measuring stations. Table 20 and Table 21 show the results for time series from suburban areas and urban areas respectively with the station type being held consistent as to have realistic anomalies injected. Table 20 shows that all models perform similarly on suburban data, just as they did on rural data. However, less of the injected anomalies are detected here than what was the case for the rural data. Exponential smoothing achieves the highest recall and F1-score here, but the deep learning models still have fewer false positives as was also the case when applying the models on the rural data. Table 21 shows the results for urban areas. Once again, the results are relatively consistent across all models, with only the exponential smoothing model scoring slightly lower on most metrics.

Model	TP	FN	FP	TN	Precision	Recall	F1-Score
Exponential Smoothing	9	10	19	3442	0.321	0.474	0.383
ARIMA	8	11	27	3434	0.229	0.421	0.296
Linear Regression model	8	11	23	3438	0.258	0.421	0.320
RNN model	8	11	18	3443	0.308	0.421	0.356
Regression Ensemble model	8	11	15	3446	0.348	0.421	0.381

Table 20: The resulting confusion matrices of injecting 19 anomalies from time series #18 into #13 (suburban area) without a threshold.

Model	TP	FN	FP	TN	Precision	Recall	F1-Score
Exponential Smoothing	7	12	20	3441	0.259	0.368	0.304
ARIMA	10	9	22	3439	0.312	0.526	0.392
Linear Regression model	10	9	22	3439	0.258	0.421	0.320
RNN model	11	8	24	3437	0.314	0.579	0.407
Regression Ensemble model	9	10	14	3447	0.391	0.474	0.429

Table 21: The resulting confusion matrices of injecting 19 anomalies from time series #19 into #14 (urban area with background type) without a threshold.

5.2.4 Impact of station type on model performance

The type of measuring station also affects the data distribution, just as the station its area does. Therefore the models once again need to be applied to data from different types of measuring stations with other characteristics kept similar. Table 21, Table 22 and Table 23 all show the results for the models on a specific type of measuring station in urban areas.

Model	TP	FN	FP	TN	Precision	Recall	F1-Score
Exponential Smoothing	6	12	16	3446	0.273	0.333	0.300
ARIMA	7	11	17	3445	0.292	0.389	0.333
Linear Regression model	6	12	19	3443	0.240	0.333	0.279
RNN model	7	11	16	3446	0.240	0.333	0.279
Regression Ensemble model	6	12	15	3447	0.286	0.333	0.308

Table 22: The resulting confusion matrices of injecting 19 anomalies from time series #20 into #15 (industrial type) without a threshold.

Table 21 shows that the models perform relatively similar on data from the background type with only exponential smoothing achieving slightly lower scores on most metrics. Overall, the scores on data from the industrial measuring stations are slightly lower than for the background type. Once again the models show very similar results in Table 22 even though they are thus slightly worse. Table 23 presents the results for data collected from traffic-type measuring stations. These results show more differences between the models with exponential smoothing scoring the highest on all

Model	**TP**	**FN**	**FP**	**TN**	**Precision**	**Recall**	**F1-Score**
Exponential Smoothing	10	7	11	3452	0.476	0.588	0.526
ARIMA	8	9	21	3442	0.276	0.471	0.348
Linear Regression model	9	8	19	3444	0.321	0.529	0.400
RNN model	6	11	15	3448	0.286	0.353	0.316
Regression Ensemble model	7	10	11	3452	0.389	0.412	0.400

Table 23: The resulting confusion matrices of injecting 19 anomalies from time series #21 into #16 (traffic type) without a threshold.

three performance metrics. On average, the deep learning models score lower here than models from the other two categories.

5.3 Overview of results

The previous results show that no single model clearly performs better across all types of data. There has been some amount of fluctuation in the results regarding the pollutant and the type and area of the measuring stations on which the models were applied. Table 24 shows the average of the previously gathered results. From this table it becomes clear that the RNN model generally generates the highest scores in the used performance metrics. This model namely achieved the highest scores for all three of these metrics, even though the amount of false positives that this model registered were not the lowest. In general the regression ensemble model also achieved quite similar scores to the RNN model. It however performed quite poorly on PM_{10} data from measuring stations in urban areas that are from the traffic or industrial type. This causes the average results to appear relatively poor for this model while it does detect relatively few false positives compared to the other models. The only model that is categorized as a machine learning model in this thesis achieved the worst results on average. This linear regression model scored quite poorly overall, especially on unrealistic anomalies in PM_{10} data.

Model	**TP**	**FN**	**FP**	**TN**	**Precision**	**Recall**	**F1-Score**
Exponential Smoothing	7.08	11.75	15.33	3445.66	0.271	0.374	0.311
ARIMA	7.08	11.75	19.58	3441.42	0.213	0.373	0.267
Linear Regression model	6.58	12.25	16.42	3444.58	0.223	0.338	0.267
RNN model	8.25	10.58	18.75	3442.25	0.300	0.431	0.353
Regression Ensemble model	7.00	11.83	15.33	3445.67	0.297	0.370	0.329

Table 24: Table containing the average values for all previously given results.

5.4 Answering research questions

Now that all results have been collected and that an overview of the results has been created, an attempt can be made to answer the research questions. This section will try to answer both research questions regarding the effectiveness with which anomalies can be detected in our data and regarding which models can achieve the best results.

5.4.1 First research question

"How effective can anomalies be detected in univariate time series data of air pollution in Europe."

The results do not seem to indicate a clear yes or no answer to this first research question. The models also appeared to perform similarly regardless of the area and type of the measuring station from which the data was collected. For realistic anomalies, which are the most interesting type of anomalies given the research question, the models do not seem to be able to detect them very well. With some types of data even resulting in none of the injected anomalies being detected by some models in the case of PM_{10} data.

For O_3 data, the models seemed to be able to detect around half of the injected anomalies regardless of whether they were realistic or unrealistic. With all models being able to detect 14 out of the 20 realistically injected anomalies in time series from a rural area with a background type, an argument can be made that the models were quite effective at detecting anomalies in such data.

A valid answer to the research question would thus likely be that it is indeed possible to detect anomalies effectively in univariate time series data of air pollution in Europe. This answer does however come with the caveat that this is not the case for all such air pollution data. This thesis has shown that several models generally failed to effectively detect realistic anomalies in PM_{10} data. Effective detection across all tested anomaly types was observed only in the case of O_3 concentration data. Even then, the term 'effectively' is debatable, as the proportion of correctly detected anomalies ranged from approximately half to three-quarters of the total.

5.4.2 Second research question

"Which anomaly detection methods achieve the best performance in detecting anomalies in univariate time series data of air pollution in Europe."

For PM_{10} air quality data, one model clearly outperformed the others: the RNN model. It consistently detected at least three of the injected anomalies, whereas the other models occasionally failed to detect any. Especially from the data shown in Table 13 it becomes clear that the RNN model significantly outperforms the other models on data from stations of the background type in suburban areas. On average, this larger amount of true positives also results in more false positives due to generally higher anomaly scores being awarded to data points. This larger amount of false positives however does not necessarily mean that de model is not performing well. As mentioned earlier, the lack of a ground truth can mean that there are anomalous data points present in the time series before any are injected. The registered amount of false positives is thus not necessarily correct due to the lack of a definition for an anomaly.

The models generally performed better on O_3 data with smaller differences in the results between the different models. Depending on both the location of the measuring station and the type of anomalies that were injected, different models achieved the best results. Due to this it is difficult to determine which models achieve the best results on such data.

The average performance of the models across all individual results have been discussed in Chapter 5.3. These average results indicate that the RNN model generally detected the highest number of anomalies, suggesting it is the most effective at detecting anomalies in univariate air pollution time series data in Europe. However, due to the differences in results between the two pollutants, this conclusion may not be generalizable to other, similar pollutants. It was however noticeable in the plots of the anomaly scores that the RNN model generally awarded higher anomaly scores to anomalous points, which might indicate that it does indeed give the best overall performance. Some of the other models like the regression ensemble model and the ARIMA model showed abnormal behaviour for certain time series. They respectively showed sudden spikes in their forecasts and kept forecasting future data points based on past anomalous points. This last remark shows the disadvantage of local forecasting models like the selected statistical and machine learning models that were selected. Their results seem to be more heavily impacted by having anomalous data points present in their sliding windows. Due to the deep learning models having been trained on a large dataset, their forecasted values do not seem to be impacted as much by such anomalous data points in their sliding window. For this reason, a likely answer to this second research question is that global forecasting models generally achieve the best results, like the selected deep learning models.

6 Discussion

This Chapter is divided into two sections. The first section discusses some of the limitations that had to be dealt with in this thesis while the second section discusses suggestions for future research in relation to this topic.

6.1 Limitations

This thesis has shown how different anomaly detection models perform on univariate time series data of European outdoor air quality. Although the results from section 5 show that the accuracy of these models vary based on measuring station characteristics and the method with which anomalies are injected, it generally seems to be possible to detect anomalies with a certain level of accuracy. There are however some important limitations regarding the method used in this thesis which might be of interest for possible future research.

As discussed previously in section 4.3, finding optimal hyperparameter configurations for a model is a complicated research area of itself. Due to limitations to the hardware that was used to run the selected models, it was not possible to test the models with a larger variety of hyperparameter configurations. Due to this reason, the models were eventually used with the same hyperparameter setting regardless of the type of time series they were applied to. If more computational power would be available for such research, it might be possible to find better performing hyperparameter configurations for individual time series.

Due to the same hardware limitations, it was also not possible to apply the anomaly detection models to longer time series segments. Due to the large amount of data that was used to train the models, adding short segments to the final time series already causes a significant increase in the time that is needed for training. For this reason the series that could be used were limited to a maximum length of around five months. To be able to obtain more accurate results this could thus be a point of focus for future research. As applying the selected models to longer or more diverse time series might give a better indication of the actual performance on outdoor air quality in Europe. Due to this limitation with regards to the length of the time series, it was relatively difficult to capture the seasonal component of the O_3 pollutant. If a longer time series could be used, the effect of the seasonality on model performance could be evaluated better.

Another limitation is found in the dataset itself. In Chapter 3 it was determined that SO_2 was an interesting pollutant to focus on due to its concentrations following a declining trend over the past 10 years. This property set it a part of the other pollutants that were discussed in this thesis. However, due to significant amounts of missing data, it was not possible to accurately apply anomaly detection models on this data. If a dataset could be created for SO_2 concentrations where there are significantly less missing data points, it would give an indication of how time series with a trend effect the performance of anomaly detection models.

The lack of a ground truth also remains an important limitation to this research as this makes it difficult to evaluate the models on the amount of false positives that they detect. Even though an attempt was made to select non-anomalous time series to inject anomalies into, the lack of a definition of an anomaly [LZVL23] creates difficulties. This creates the possibility that false positives could actually be true positives depending on what an actual definition of an anomaly would look like in such data. This limitation thus limits the extent to which the models can be evaluated. In any possible future research, a focus should be put on improving this limitation for this reason. For example, with enough resources it could be possible to use a manually labeled dataset which would allow for more accurate results.

6.2 Future research

The mentioned limitations also bring suggestions for possible future research where the impact of these limitations could potentially be reduced. Besides these suggestions for future research, it might also be of interest to focus on more pollutants and on different regions. It is not guaranteed that the results presented in Chapter 5 will generalize to data from outside Europe or to other types of pollutants. The fact that SO_2 data had a significant amount of missing data points would make this one of the most interesting directions for future research. Since the concentrations of the other pollutants do not exhibit a clear trend, further research is needed to determine whether this characteristic affects the performance of the selected models. Another question that remains after the gathered results is whether the results on European data translate to data from other regions.

There is a lot of metadata available for the EEA dataset, including information about the different measuring stations. The characteristics of the measuring stations can be used to select training sets for the models with a selection of these characteristics. This thesis has looked into the impact of some of these characteristics on the performance of these models. There are however more remaining characteristics which might also be of interest. One possibility for future research is also to take a closer look at the performance of anomaly detection models on contextual anomalies [SWJR07] [LVL23]. It could be possible to train a model on a certain selection of measuring stations and subsequently inject anomalies from this training set into a target series which has a different value for one of the characteristics. With this method it would be possibly to inject contextual anomalies which are more subtle than the unrealistically injected anomalies as seen in Chapter 5.1.1, but likely easier to detect than the realistic anomalies as described in Chapter 5.1.2.

A final suggestion for future research would be to apply more anomaly detection models on the EEA dataset. By applying more than the five selected models it might be possible to find models that provide better results. It might also be interesting to select more models from the machine learning category as only one was selected for use in this thesis.

7 Conclusion

This thesis has shown varying results in the detection of anomalies in European air quality data. The best results were achieved on unrealistic anomalies that were injected into PM_{10} data as some models managed to detect all anomalies with relatively few false positives. Applying the models on anomalies that were realistically injected into PM_{10} data, however, showed quite poor results overall. Applying the models to data that was specifically gathered from measuring stations with certain characteristics generally also showed quite poor performance. With most models failing to detect a single anomaly in data from traffic type measuring stations in urban areas, it appears that the area and type of the measuring stations had a slight impact on model performance. However, overall performance on realistic anomalies remained quite poor.

The models performed significantly better on time series for O_3 concentrations. In contrast to what was the case for the PM_{10} data, the models gave better results on realistically injected anomalies instead of unrealistic anomalies. On average, about half of the injected anomalies were correctly detected by the models. This did result in a slightly higher amount of false positives, but not by a large amount compared to the results for PM_{10}. The models also appeared to perform similarly regardless of the area and type of the measuring station from which the data was collected.

This thesis has attempted to find an answer to two research questions. The first one asked how effective anomalies can be detected in univariate time series data of European air pollution. This thesis was not able to give a clear yes or no answer to this question based on the gathered results. However, a valid answer to this question would likely be that it is indeed possible to detect anomalies effectively in univariate time series data of air pollution in Europe. This would come with the caveat that this is not the case for all such air pollution data as this thesis noticed that the models generally performed better on O_3 data than on PM_{10} data. It also remains debatable whether the term 'effectively' can be used in this answer as the proportion of correctly detected anomalies ranged from approximately half to three-quarters of the total in O_3 data.

The second research question asked which anomaly detection methods would achieve the best results in detecting anomalies in univariate time series data of air pollution in Europe. The results have indicated that of the five selected models, the Recurrent Neural Network model was able to generally detect the highest number of anomalies. It was also found that local forecasting models seemed to be more heavily impacted by having anomalous data points in their sliding windows. For this reason, a likely answer to this research question was that global forecasting models, like the two selected deep learning models, generally achieve better results than local forecasting models.

While Wei at al. [WJJX+23] have shown that anomalies can be detected in indoor air quality by using an LSTM-AE model, this thesis was arguably not able to determine whether anomalies could accurately be detected in outdoor European air quality depending on the pollutant. Even though anomaly detection methods could be useful for policy-makers in lowering pollutant levels in air quality, more research would clearly be needed to determine which types of models can most effectively achieve this for each individual pollutant.

8 Acknowledgement

I am very grateful to Zhong Li for his support throughout the writing of this thesis. I would like to thank him for being willing to take the time to both give elaborate feedback and to have regular meetings. His supervision and advice have been extremely helpful in the completion of this thesis.

I am also thankful to Matthijs van Leeuwen for helping me find a suitable subject for this thesis and for connecting me with Zhong Li for the subsequent supervision of this thesis.

References

[AAS$^+$22] Ahmad Alsharef, Karan Aggarwal, Sonia, Manoj Kumar, and Ashutosh Mishra. Review of ml and automl solutions to forecast time-series data. *Archives of computational methods in engineering*, 29(7):5297–5311, 2022.

[ACADL18] Hussan AL-Chalabi, Yamur Al-Douri, and Jan Lundberg. Time series forecasting using arima model: A case study of mining face drilling rig. 08 2018.

[Bre01] L Breiman. Random forests. *Machine Learning*, 45:5–32, 10 2001.

[BW20] Mohammad Braei and Sebastian Wagner. Anomaly detection in univariate time-series: A survey on the state-of-the-art. 2020.

[Cal20] Ovidiu. Calin. *Deep Learning Architectures : A Mathematical Approach*. Springer Series in the Data Sciences. Springer International Publishing, Cham, 1st ed. 2020. edition, 2020.

[CBK09] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM computing surveys*, 41(3):1–58, 2009.

[CJZ07] Eugene K. Cairncross, Juanette John, and Mark Zunckel. A novel air pollution index based on the relative risk of daily mortality associated with short-term exposure to common air pollutants. *Atmospheric environment (1994)*, 41(38):8442–8454, 2007.

[CZS$^+$16] Guilherme O. Campos, Arthur Zimek, Jörg Sander, Ricardo J. G. B. Campello, Barbora Micenková, Erich Schubert, Ira Assent, and Michael E. Houle. On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study. *Data mining and knowledge discovery*, 30(4):891–927, 2016.

[DPBM25] Pradeep Kumar Dongre, Viral Patel, Upendra Bhoi, and Nilesh N. Maltare. An outlier detection framework for air quality index prediction using linear and ensemble models. *Decision analytics journal*, 14:100546–, 2025.

[DPK13] K. Dimitriou, A.K. Paschalidou, and P.A. Kassomenos. Assessing air quality with regards to its effect on human health in the european union through air quality indices. *Ecological indicators*, 27:108–115, 2013.

[Eur22] European Environment Agency. Air quality e-reporting: Air quality time series (e1a & e2a data sets), 2022.

[Gar06] Everette S. Gardner. Exponential smoothing: The state of the art—part ii. *International journal of forecasting*, 22(4):637–666, 2006.

[HA21] Rob Hyndman and G. Athanasopoulos. *Forecasting: Principles and Practice*. OTexts, Australia, 3rd edition, 2021.

[HLP$^+$22] Julien Herzen, Francesco Lässig, Samuele Giuliano Piazzetta, Thomas Neuer, Léo Tafti, Guillaume Raille, Tomas Van Pottelbergh, Marek Pasieka, Andrzej Skrodzki, Nicolas Huguenin, Maxime Dumonal, Jan Kościsz, Dennis Bader, Frédérick Gusset, Mounir Benheddi, Camila Williamson, Michal Kosinski, Matej Petrik, and Gaël Grosch. Darts: User-friendly modern machine learning for time series. *Journal of Machine Learning Research*, 23(124):1–6, 2022.

[Hol99] S. T. Holgate. *Air pollution and health.* Academic Press, San Diego, CA, 1st ed. edition, 1999.

[HS97] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[IA24] Amjad Iqbal and Rashid Amin. Time series forecasting and anomaly detection using deep learning. *Computers & chemical engineering*, 182:108560–, 2024.

[LVL23] Zhong Li and Matthijs Van Leeuwen. Explainable contextual anomaly detection using quantile regression forests. *Data Mining and Knowledge Discovery*, 37(6):2517–2563, 2023.

[LWvL25] Zhong Li, Yuhang Wang, and Matthijs van Leeuwen. Towards automated self-supervised learning for truly unsupervised graph anomaly detection. *Data Mining and Knowledge Discovery*, 39(5):1–43, 2025.

[LZVL23] Zhong Li, Yuxuan Zhu, and Matthijs Van Leeuwen. A survey on explainable anomaly detection. *ACM Transactions on Knowledge Discovery from Data*, 18(1):1–54, 2023.

[MMN23] Manuel Méndez, Mercedes G. Merayo, and Manuel Núñez. Machine learning algorithms to forecast air quality: a survey. *The Artificial intelligence review*, 56(9):10031–10066, 2023.

[NK21] Yoni Nazarathy and Hayden Klok. *Statistics with Julia : fundamentals for data science, machine learning and artificial intelligence.* Springer Series in the Data Sciences. Springer, Cham, Switzerland, 2021.

[PVG$^+$11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[Reu21] Reuters. Air pollution kills 7 million a year, says who as it tightens guidelines. https://www.scmp.com/news/world/europe/article/3149735/air-pollution-kills-7-million-year-says-who-it-tightens?module=perpetual_scroll_0&pgtype=article, 2021. Accessed: 2025-08-02.

[SP10] Skipper Seabold and Josef Perktold. statsmodels: Econometric and statistical modeling with python. In *9th Python in Science Conference*, 2010.

[SWJR07] X. Song, M. Wu, C. Jermaine, and S. Ranka. Conditional anomaly detection. *IEEE transactions on knowledge and data engineering*, 19(5):631–645, 2007.

[SWP22] Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. Anomaly detection in time series: a comprehensive evaluation. *Proceedings of the VLDB Endowment*, 15(9):1779–1797, 2022.

[WJJX$^+$23] Yuanyuan Wei, Julian Jang-Jaccard, Wen Xu, Fariza Sabrina, Seyit Camtepe, and Mikael Boulic. Lstm-autoencoder-based anomaly detection for indoor air quality time-series data. *IEEE Sensors Journal*, 23(4):3787–3800, 2023.

[WK23] Renjie Wu and Eamonn J. Keogh. Current time series anomaly detection benchmarks are flawed and are creating the illusion of progress. *IEEE transactions on knowledge and data engineering*, 35(3):2421–2429, 2023.

[Wol92] David Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 12 1992.

[ZLLS23] Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola. *Dive into Deep Learning*. Cambridge University Press, 2023. https://D2L.ai.

[ZRA21] Yue Zhao, Ryan Rossi, and Leman Akoglu. Automatic unsupervised outlier model selection. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 4489–4502. Curran Associates, Inc., 2021.