

[About](#) [Projects](#) [Talks](#) [Blog](#) [Links](#)

## Using functional analysis to model air pollution data in R

Let's say you need to understand how your data changes within a day, and between different days. Functional analysis is one approach of doing just that so here's how I applied functional analysis to some air pollution data using R!

November 14, 2022

Let's say you need to understand how your data changes within a day, and between different days. For example, if you have hourly pollution data that follows a regular pattern throughout a day, but follows different patterns on a Wednesday and Saturday. Functional analysis is one approach of doing just that. During my PhD, I developed methods based on functional analysis to identify outlier demand in booking patterns for trains in railway networks. To demonstrate that those statistical methods are also applicable in other areas, I started to analyse air pollution data across the United Kingdom. This blog post will discuss the idea of using functional analysis to model air pollution data with the aim of identifying abnormal pollution days.

## Introducing the data

The data comes from DEFRA's (Department for Environment, Food and Rural Affairs) Automatic Urban and Rural Network (AURN), which reports the level of nitrogen dioxide (among other pollutants) in the air every hour at 164 different locations. For this analysis, we considered data recorded between November 6, 2018 to November 6, 2022.

The data can be downloaded from [uk-air.defra.gov.uk/data/data\\_selector\\_service](https://uk-air.defra.gov.uk/data/data_selector_service).

► Show code: reading in data

## Data wrangling

It's often said that 80% of data science is cleaning data, and this analysis was no exception. The data initially looked a little bit like this:

...1	...2	Aberdeen	...4	Aberdeen Erroll Park	...6
Date	Time	Nitrogen dioxide	Status	Nitrogen dioxide	Status
2018-11-06	01:00:00	27.56325	V ugm-3	No data	NA
2018-11-06	02:00:00	25.3542	V ugm-3	No data	NA
2018-11-06	03:00:00	22.12239	V ugm-3	No data	NA

...with another 324 columns for the remaining stations. With a little but of help from {dplyr} and {lubridate}, we can clean this up into something a little bit nicer:

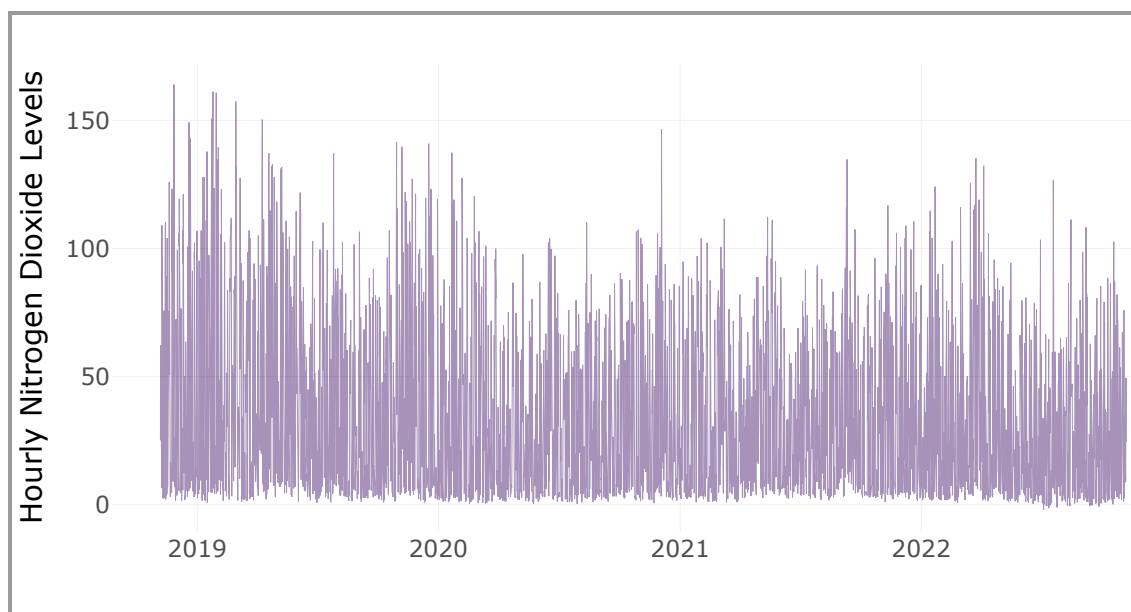
Date	Time	DateTime	Aberdeen	Aberdeen Erroll Park
2018-11-06	1H 0M 0S	2018-11-06 01:00:00	27.56325	NA
2018-11-06	2H 0M 0S	2018-11-06 02:00:00	25.35420	NA
2018-11-06	3H 0M 0S	2018-11-06 03:00:00	22.12239	NA
2018-11-06	4H 0M 0S	2018-11-06 04:00:00	17.13361	NA

► Show code: tidying the data

For this example, let's focus on a single station (we might come back to how to deal with multiple stations in a later blog post...). Here, let's choose the *Aberdeen Wellington Road* to analyse...

► Show code: selecting a station

... and have a little look at what the data looks like. It's a little bit difficult to see in this plot, but there are a few missing values. Not every hour of every day has a recorded value for the level of Nitrogen Dioxide - there are 55 days with at least one missing value.



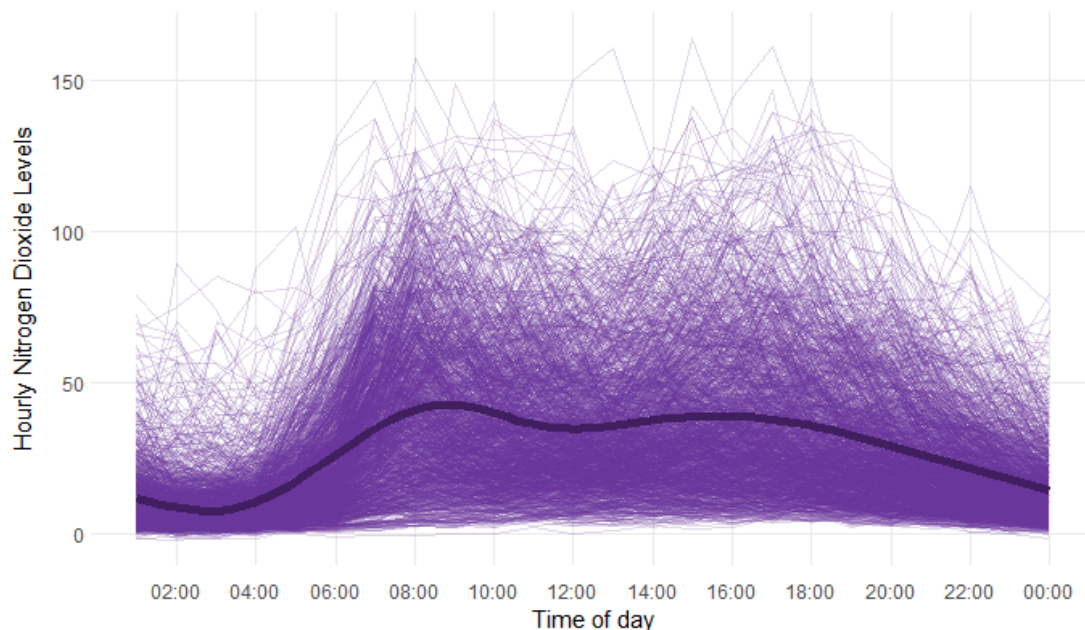
► Show code: plotting the data

There are a few different approaches to dealing with missing data - we'll utilise two different approaches here, depending on how many missing values there are. If there are more than 10 hours of data missing for a single day - we'll discard that entire day and exclude it from our data set. There are only six days in the four year period the data covers that fit this criteria, so we're not losing too much data. For the remaining days, we use mean imputation. For example, if a value is missing from 04:00 on a specific day, we'll use the mean of the non-missing values taken at 04:00 on every other day.

► Show code: dealing with missing data

## Functional approaches to modelling

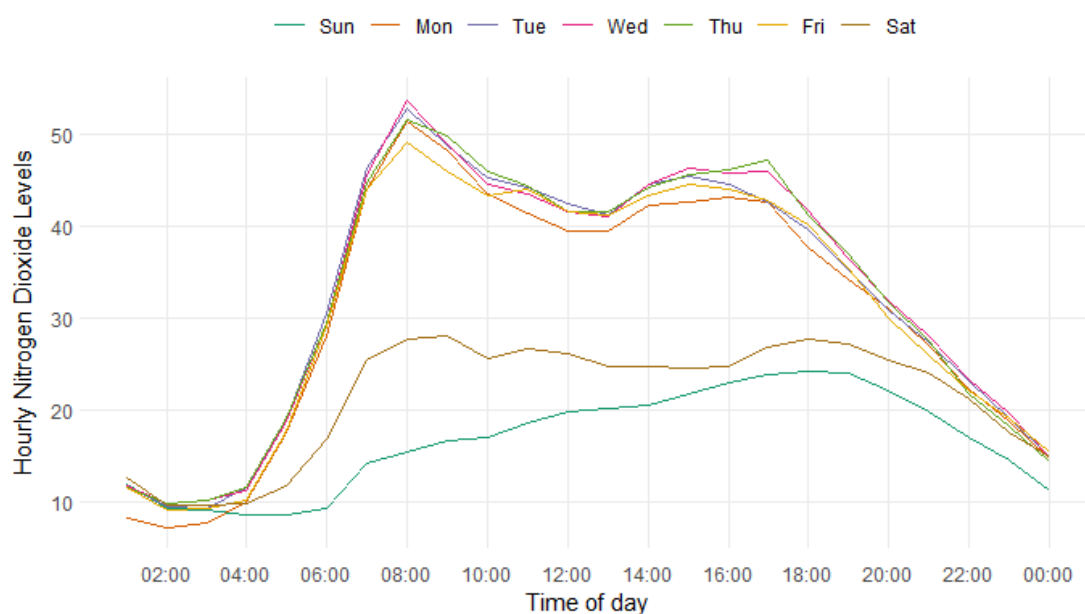
Now that we've tidied up the data and dealt with the missing values, it's time to get stuck into the modelling process! First of all, let's have another look at our data but in a slightly different way:



► Show code: plotting the data

The nitrogen dioxide levels follow a similar pattern each day, with higher levels around rush hour in the morning and evening. Functional analysis treats the pollution pattern for each day as an observation of a function over time. This allows us to compare the differences between patterns on different days, without complications from the varying pollution levels throughout each day.

We still need to deal with seasonal and trend components of the functional data. For example, since the higher levels of pollution around 8am and 5pm are most likely caused by people travelling to work, this raises the question of whether pollution levels are lower on weekends, when there are generally fewer people travelling to work. Are there different pollution patterns on different days of the week?



► Show code: plotting the data

Visualising the average daily patterns suggests that pollution patterns on weekdays are generally very similar, with Saturdays and Sundays being lower on average but still different from each other. To test this formally, a functional ANOVA test can be applied. This tests whether the average pollution pattern on a weekday is significantly different from the average pollution pattern on a Saturday or Sunday. The test returns the probability that, if the two pollution patterns were equal, the observed patterns would be this different. For this analysis, the probability was 0. Therefore, we can conclude there is a significant difference in the

pollution levels on weekdays compared to Saturdays and Sundays. We can either analyse Saturdays, Sundays, and weekends separately, or fit a model to remove the seasonality and analyse the residuals instead.

A simplified version of the regression model might look something like this:

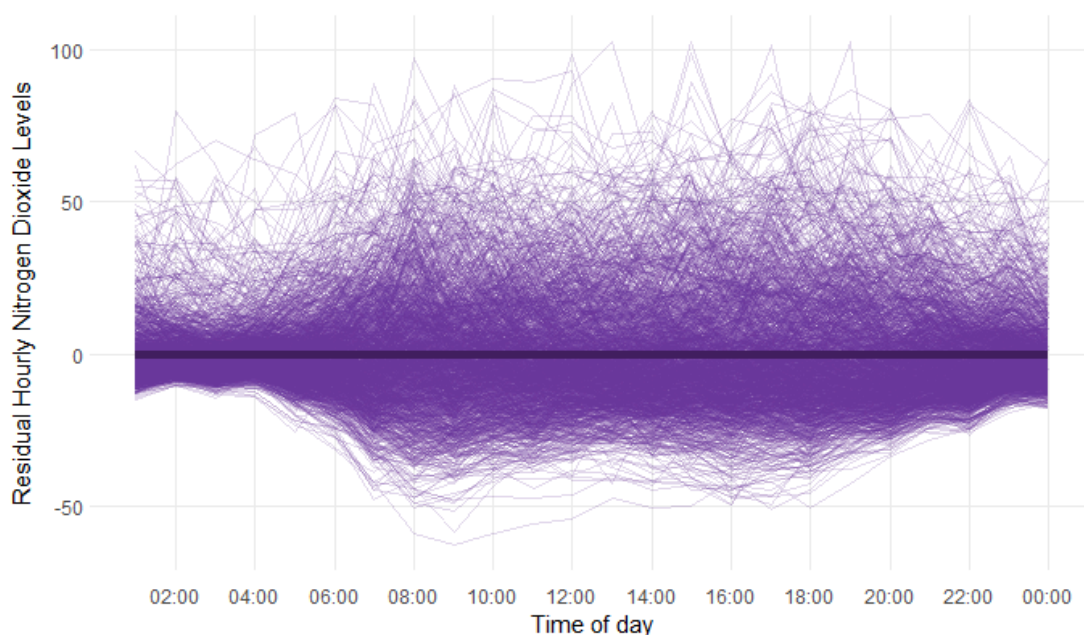
$$y_n = I_{weekday,n} + I_{Saturday,n} + e_n$$

where  $I_{weekday,n}$  is 1 if the  $n^{th}$  day is a weekday and 0, otherwise. We can use an analogous method to test if different months follow different daily patterns, and if there is a significant difference between years. Interestingly, November, December, and January all exhibit higher level of nitrogen dioxide, particularly in the evening. One possible explanation for why these months have higher levels of nitrogen dioxide is fireworks. Fireworks are commonly set off on Bonfire Night, around Christmas, and New Year's Eve. Fireworks have been shown to contribute to elevated levels of gaseous pollutants, including nitrogen dioxide.

We can account for all three types of variation (weekday, monthly, and annual) within a functional regression model.

► Show code: fitting the functional regression model

We can then analyse the residuals, that look a little bit like this:



► Show code: plotting the residuals

## Finding outliers in the data

To describe how normal (or abnormal) the (residual) pollution pattern for each day is, we calculate the functional depth. A depth measurement attributes a sensible ordering to observations, such that observations which are close to average have higher depth and those far from the average have lower depth. Days which have very low depth can be classified as outliers. Functional depth takes into account the abnormality both in the magnitude (for example, overall higher pollution levels throughout the day), and shape (for example, pollution levels peak earlier in the day than normal) of the pollution pattern.

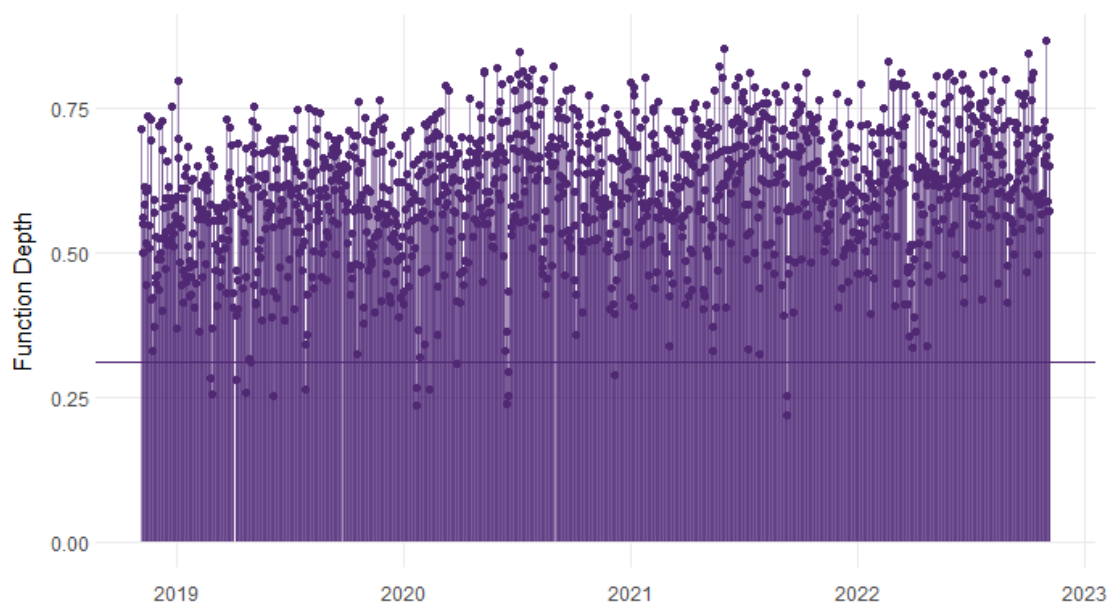
The mathematics behind the calculation of functional depth isn't super friendly so we won't go into detail here, but if you're interested in it, I'd recommend reading [this paper](#) which introduces the concept.

► Show code: calculating the depths

A threshold is then calculated for the functional depths. If the functional depth for a given day is below that threshold, the day is classified as an outlier.

► Show code: calculating the threshold

Now we can take a look at the functional depths, and the associated threshold.



► Show code: plotting the data

Much like when inspecting residuals, a scatter plot of functional depths over time should appear random. If there is evidence of seasonality, or trend in the functional depths, we would want to revisit the functional regression model. Here, the depths look alright.

The days classified as outliers, are those that fall below the threshold. In this case, 16 days were classified as outliers. As we would expect, the outliers appear randomly distributed across the period of time the data covers.

► Show code: classify the outliers

If you're super keen on finding out more about applying function depth to find outlying time series, you can read [our paper](https://nrennie.rbind.io/blog/using-functional-analysis-to-model-air-pollution-data-in-r/) on how we applied similar methodology to railway booking data.





---

For attribution, please cite this work as:

**Using functional analysis to model air pollution data in R.**

Nicola Rennie. November 14, 2022.

[nrennie.rbind.io/blog/using-functional-analysis-to-model-air-pollution-data-in-r](https://nrennie.rbind.io/blog/using-functional-analysis-to-model-air-pollution-data-in-r)

Licence: [creativecommons.org/licenses/by/4.0](https://creativecommons.org/licenses/by/4.0)

[← Automatically deploying a Shiny app for browsing #RStats tweets with GitHub Actions](#) [30 Day Map Challenge 2022](#) [→](#)

---

© 2024 Nicola Rennie. Made with [Hugo Apéro](#).