Pollution report documentation

Personal project that I took up to practice and learn how to work with Power BI better, this document folder serves as a step-by-step overview of what I did while working on the project with the data provided. This project is about government pollution in the UK, with the data being sourced directly from gov.uk, making cleaning the data essentially unnecessary in this case. This is more of a reference as to how I got my report working and functioning, along with the way I think data should be modeled and visualized for a report. For this project, I focused mainly on particulate matter, nitrogen oxides and ammonia released into the atmosphere, though there is more data present for other emissions such as methane on their website.

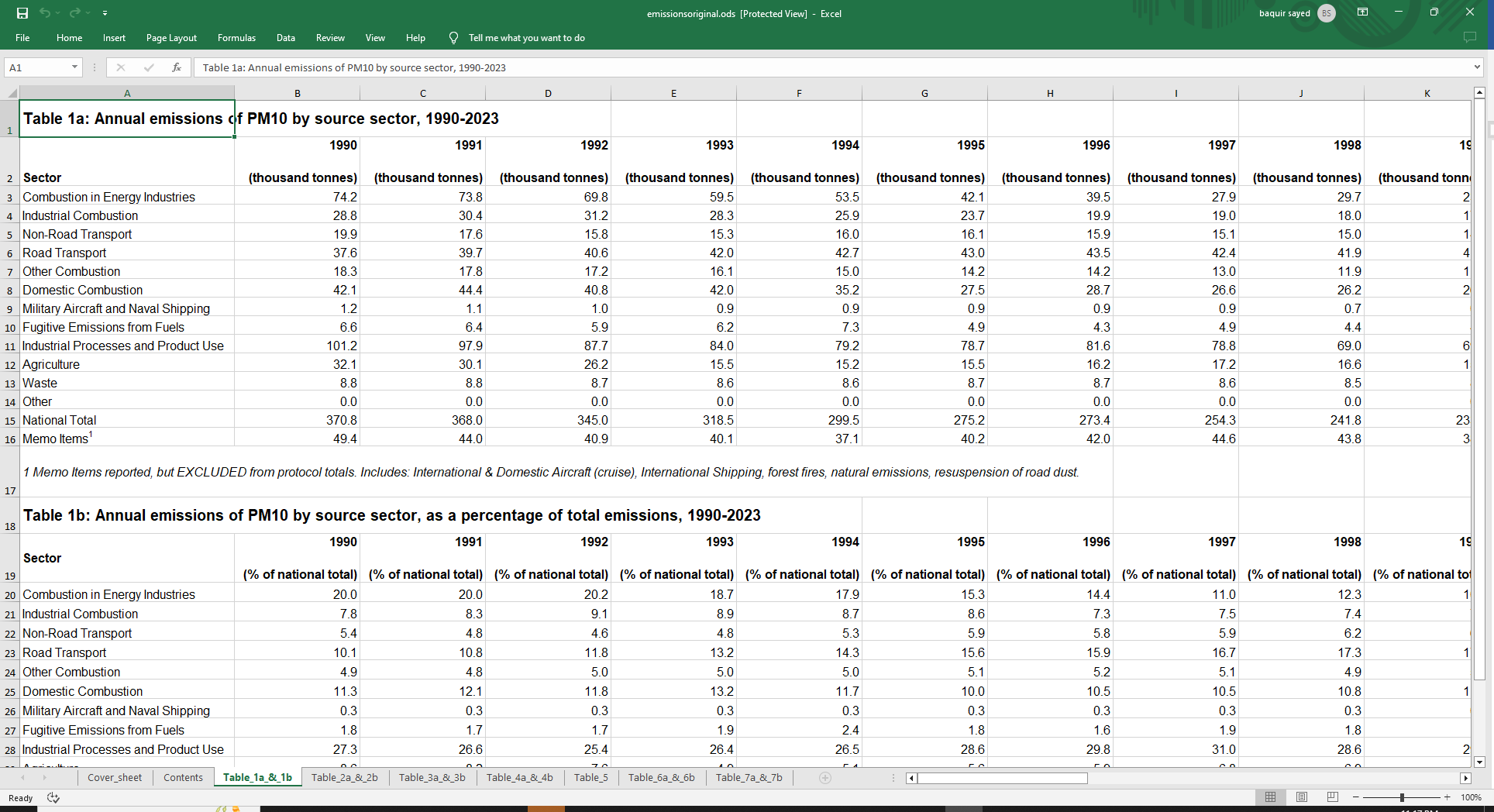
Disclaimer

I am pretty new to Power BI and data analysis in general, this was a personal project I did for practice and how to transform a set of data that is extremely convenient to work with visualizations. This has been a learning experience in determining what is a solid setup for power bi reports, keeping it as optimized as possible and easy to visualize and gain insight from. An important note, ignore some of the discrepancies in the images in case they are present, because some pictures might be older and changed/reworked, although I will do my absolute best to remove these. Last updated on 14/5/25.

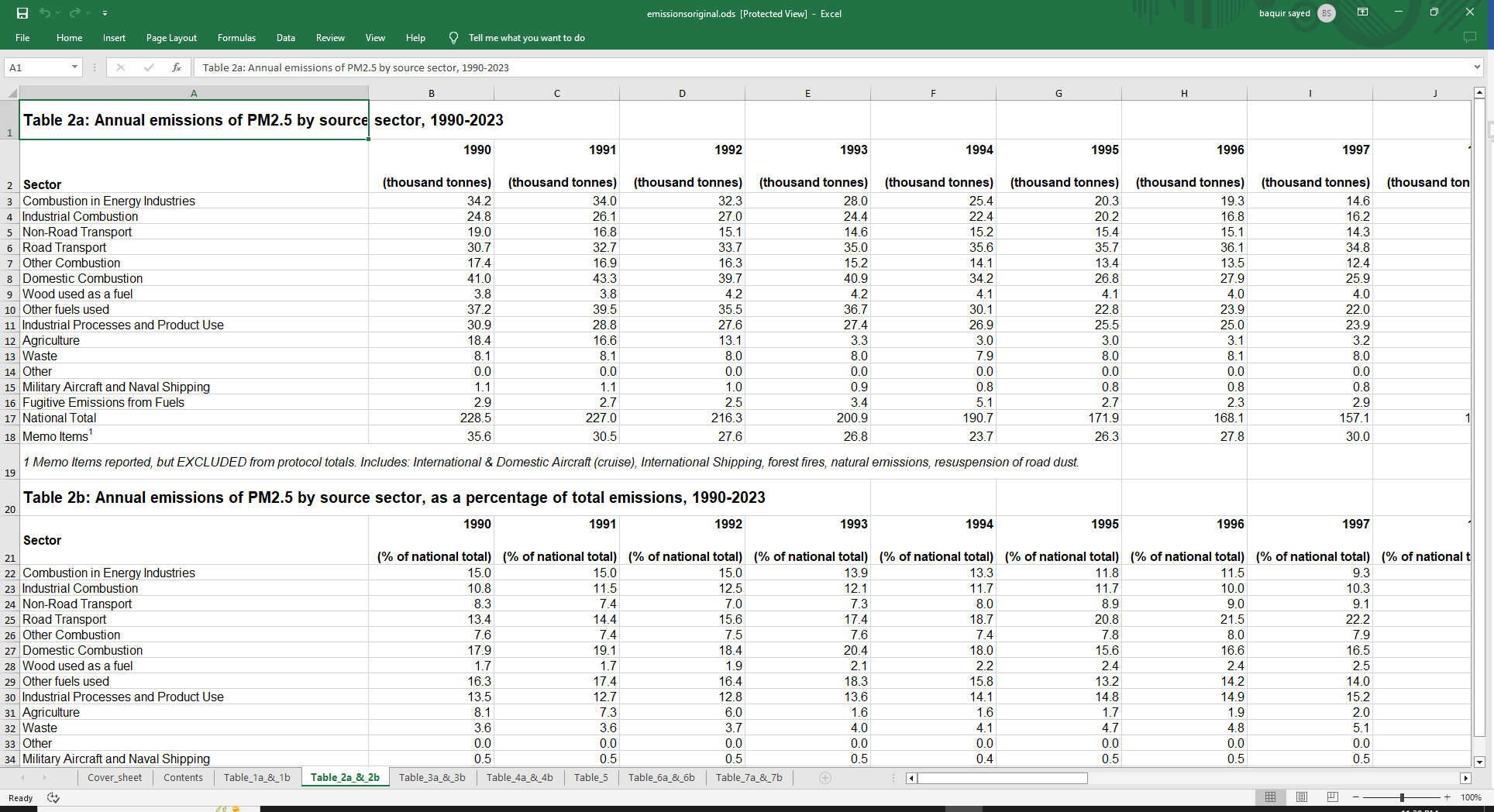
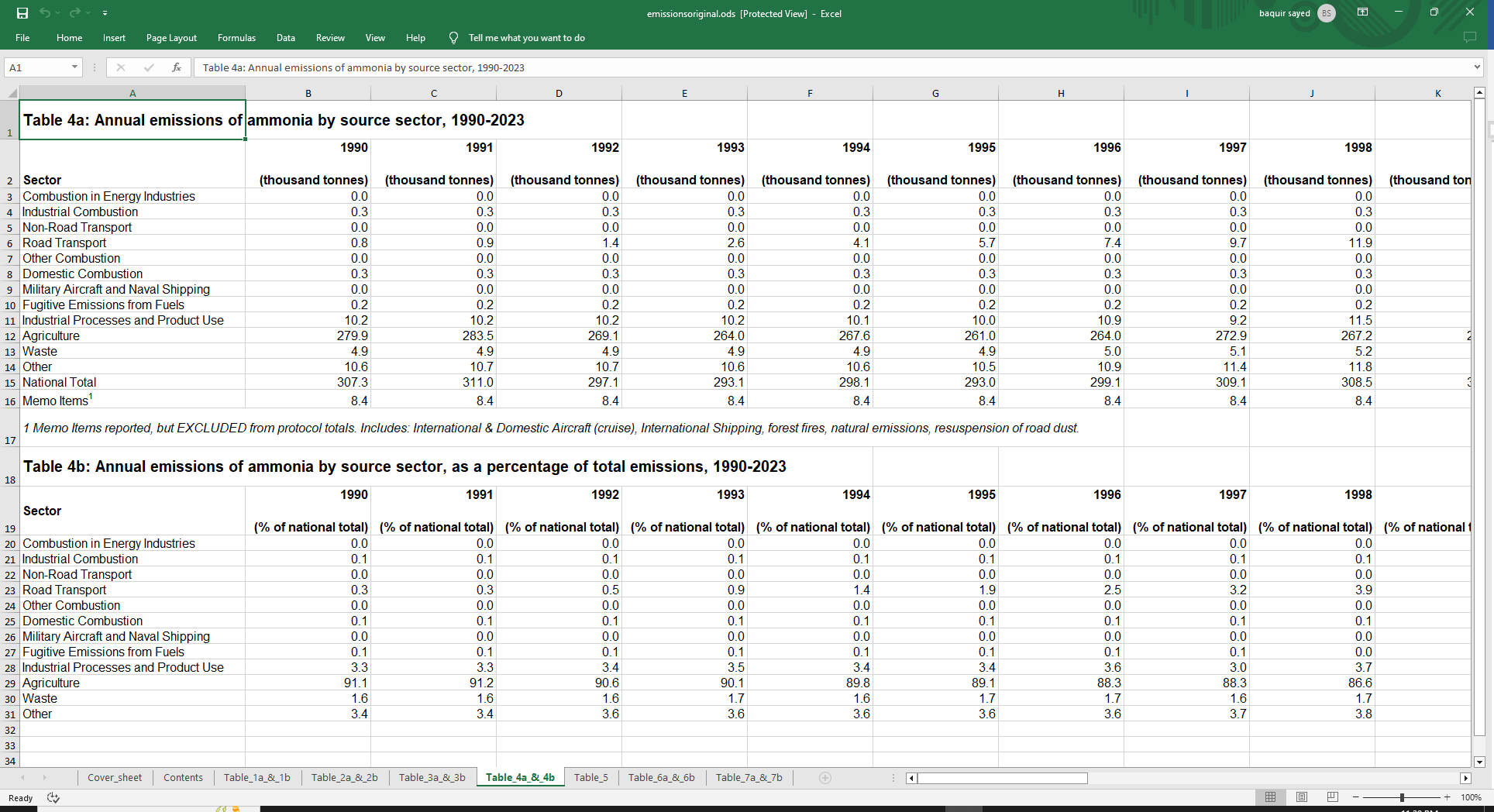
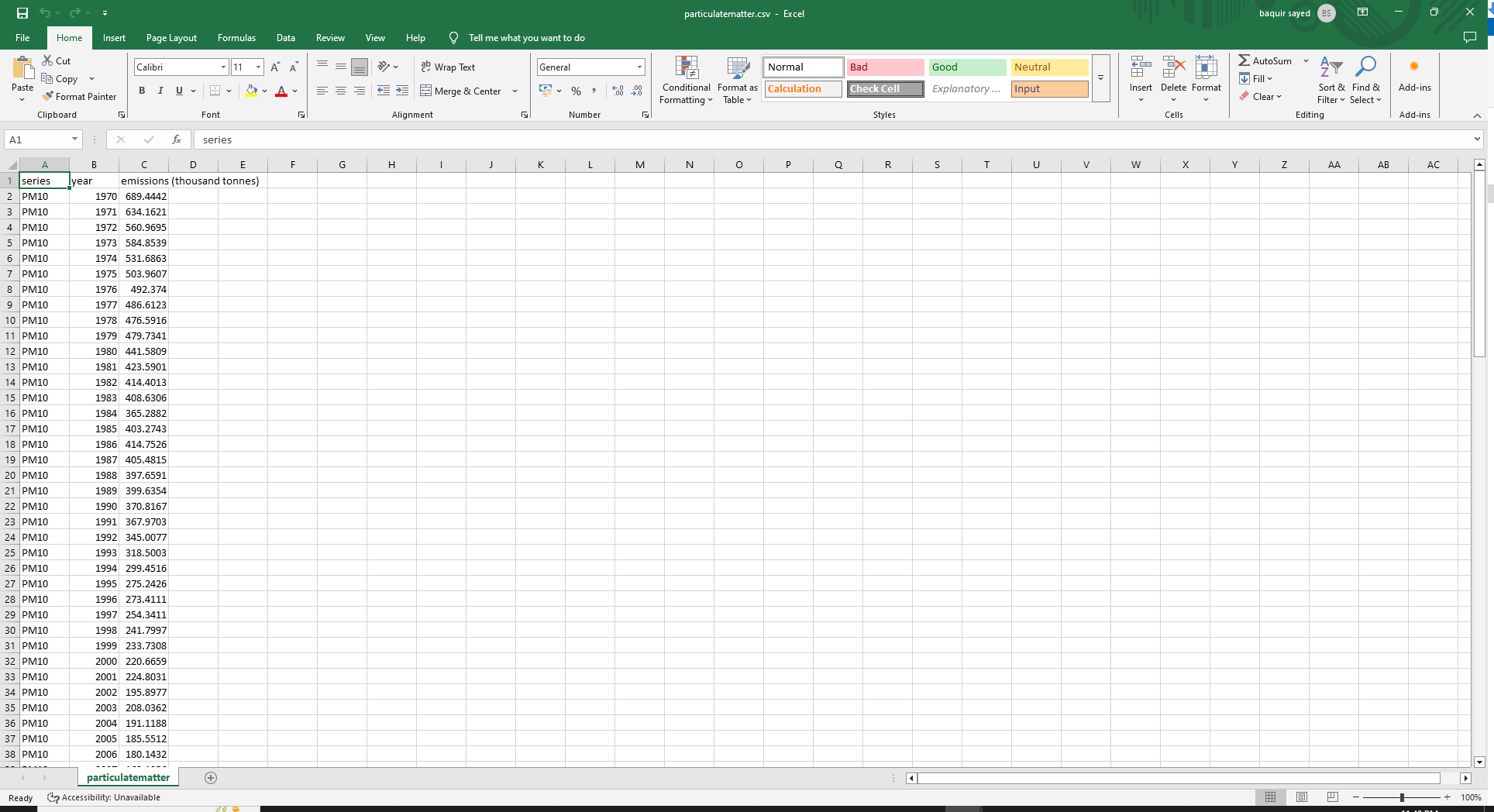
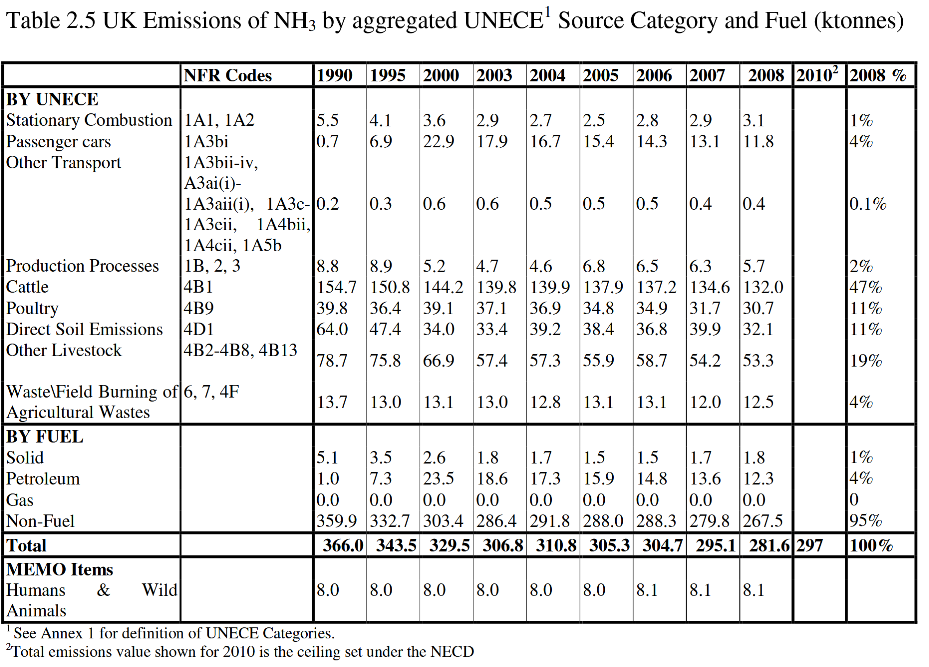
# Data sourcing, cleaning and transformation

Being directly sourced from the government, the data is very clean, so I didn’t have to do any cleaning on my end (although there is some discrepancy as mentioned later in this segment). As for transformation, I did a few transformations to make it easier to work on the data. Starting off, the data looked like this from the Excel and ODS files (links are found in reference and also uploaded to the repository). Here is an example.

**Emissions of PM10 in tonnes and as a %**

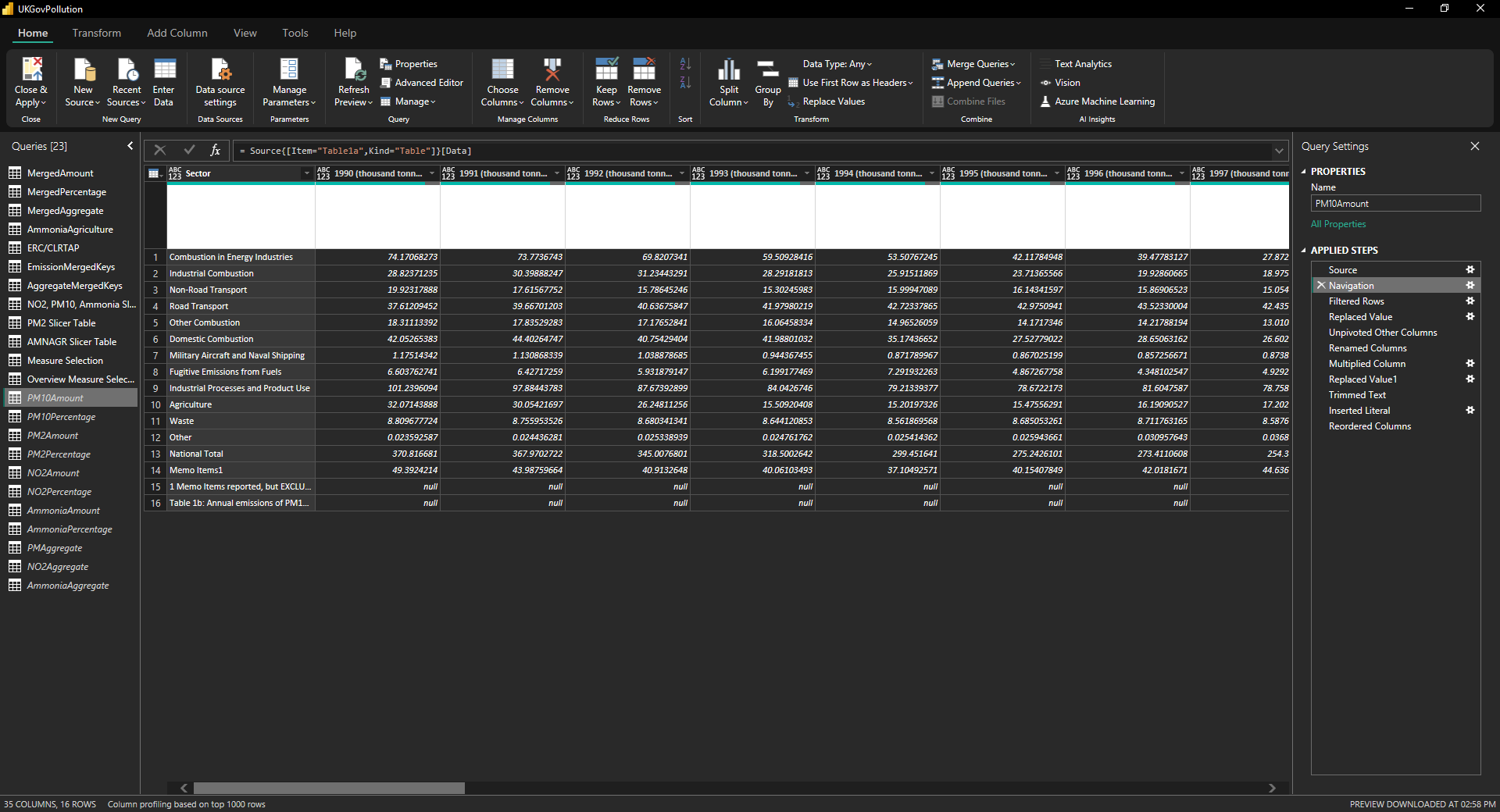


The data was split between different industries/sectors where the emissions were coming from, and were summed up yearly with one of the columns being a “National Total” for that year. There was also a list of “Memo items”, which were recorded but not included in the total, as it included data from emissions such as forest fires, international aircraft/shipping and natural emissions. From the three emissions I decided to visualize, there were a few key points that I learned while looking over the dataset.

* PM10, NO2 and Ammonia had the same sectors between the three of them, while PM2.5 and Ammonia from agriculture had different sectors. PM 2.5 had *most* of the same columns, but had a few differences, such as the inclusion of “Wood used as a fuel” and “Other fuels used”. 
* Ammonia was heavily skewed in its distribution, as most of its emissions come from agriculture. More specifically, animals and use of inorganic fertilizer produces an immense level of emissions. Between 1990 and 2023, the lowest amount of ammonia produced by agriculture was 84.3% of the national total, and thus it needed its own separate second table based only around agriculture and ammonia produced there. 
* Nitrogen Oxide was by far, the highest emission produced from the three I selected, and showed the steepest decline compared to the rest.
* The UK government had published a set of aggregate data as well, these were just the national totals in their own .csv files. The problem here is that these records stretch back all the way to 1970 (or 1980 for Ammonia). We do not have the data spread between sectors from that time period, just its aggregate. I still decided to import this data, mainly for static visuals if necessary because applying a filter context to it would be useless.   
  
* While trying to find pre-1990 sector data, I came across a report from DEFRA (Department for environment, food and rural affairs) and these reports came up with significantly higher ammonia levels and significantly lower NO2 levels from agriculture compared to the one on gov.uk. I’ve still decided to go with the data on gov.uk, mainly because it is a more complete dataset, but I’ve linked the entire report in references and a list of NFR codes aswell. The report has completely different numbers compared to everything.
* Nitrogen Oxide is measured as an aggregate in (million tonnes) instead of the standard (thousand tonnes) that is done by all other emissions, while the statistical report uses thousand tonnes.

These were the first important bits of information I gleamed over while performing some basic EDA on my data and trying to find additional data sources. I tried finding data based on the sector pre-1990, but didn’t find anything thorough enough to use in a report.

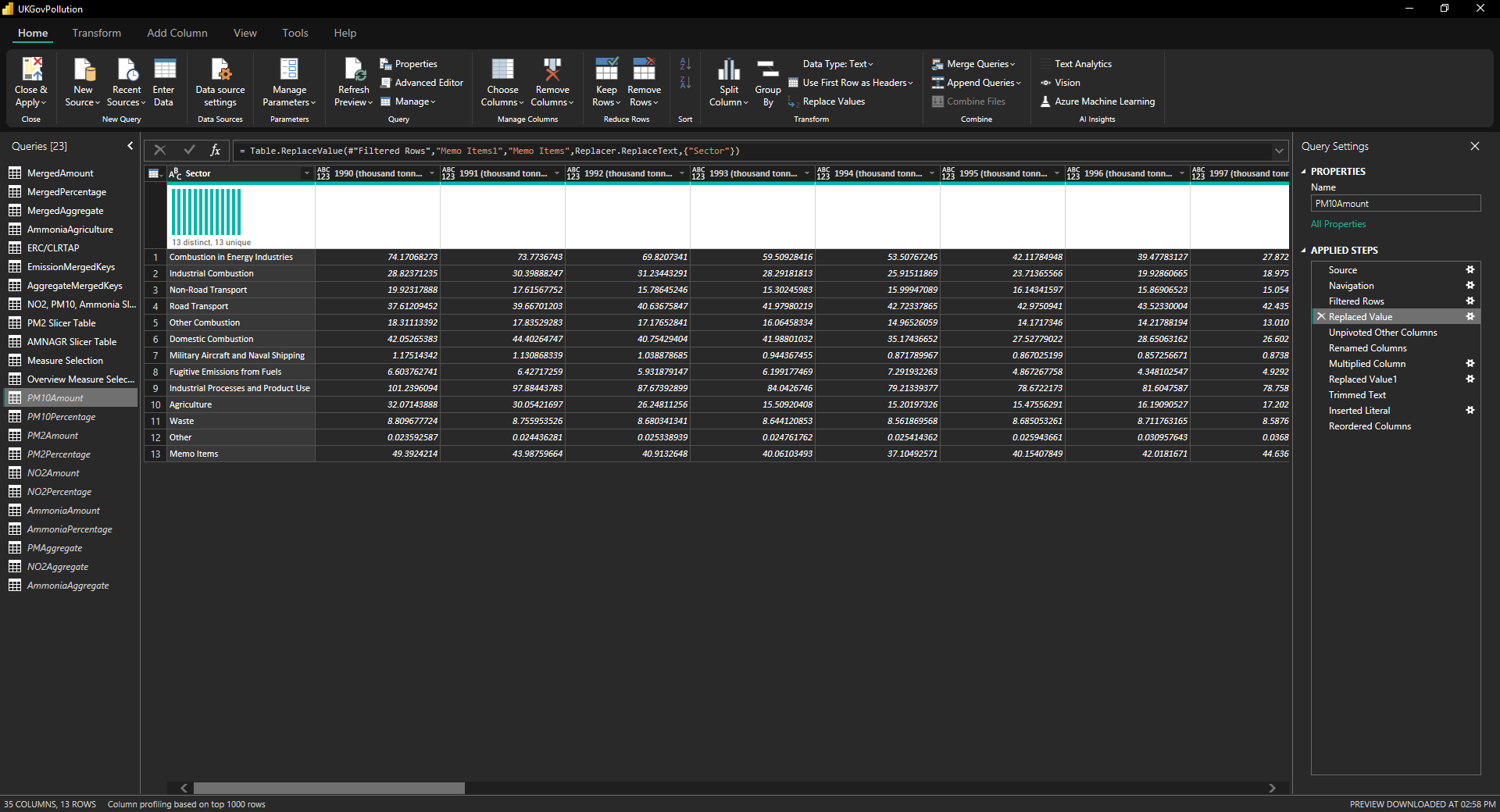
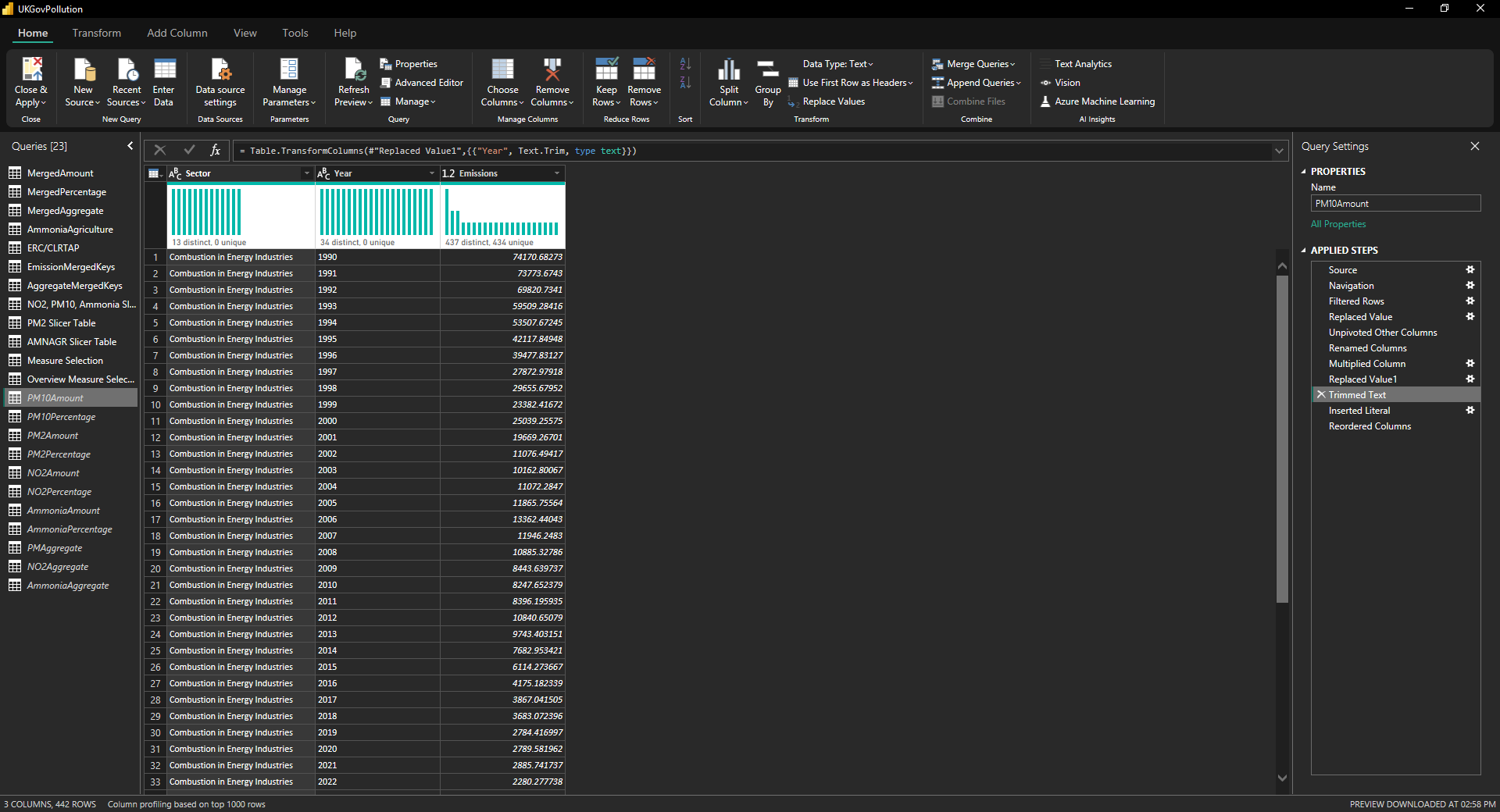
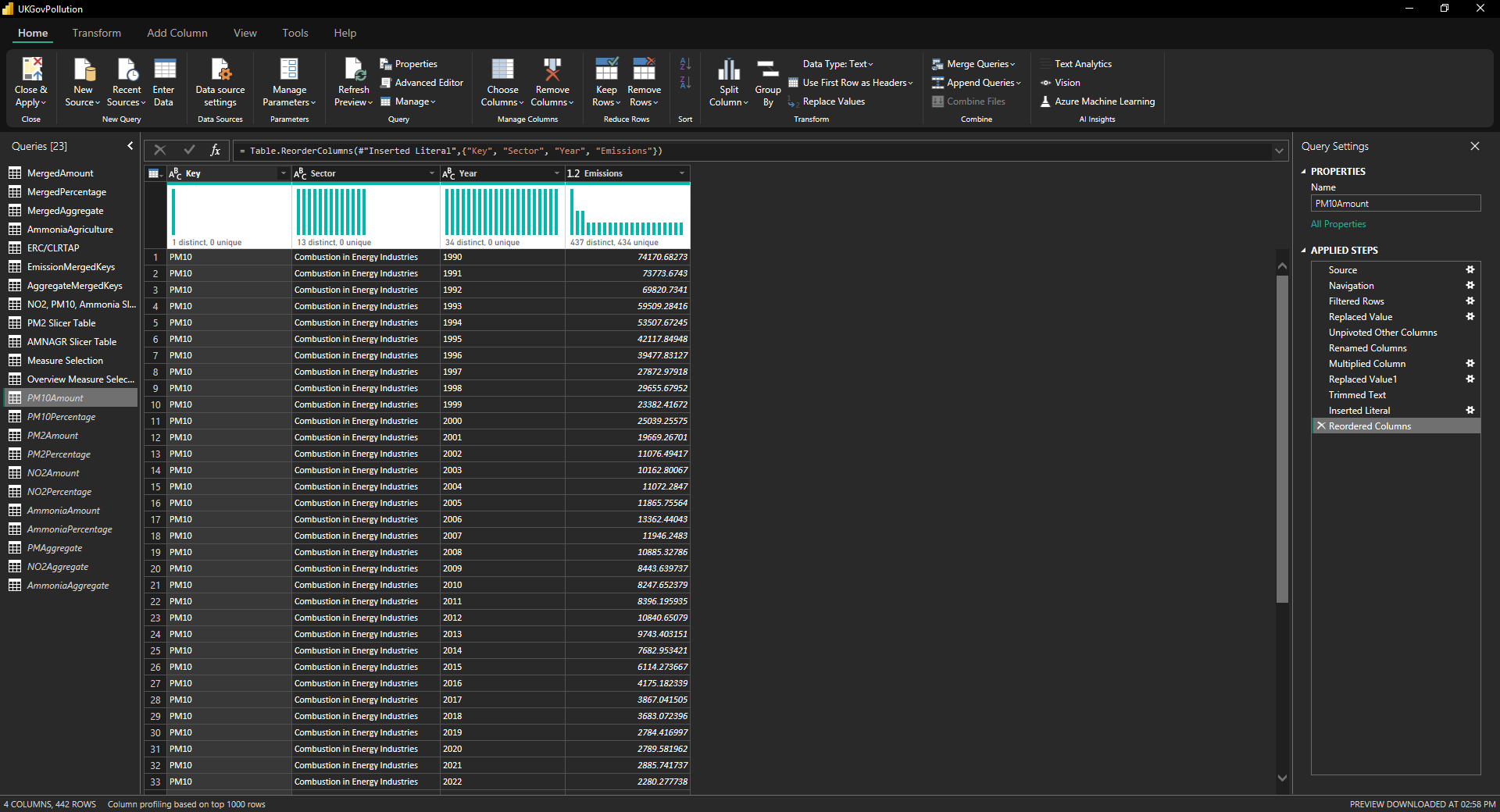
Importing data from the dataset, it conveniently split up between different tables inside the sheet itself. For example, Table 1a and 1b contain data about PM10 emissions, 1a is PM10 emissions in kilo tonnes, split based on their source sector, while 1b is percentage of the emissions split based on source sector. This allowed for me to split up the percentages and the of emissions in kilo tonnes for easier visualization and reporting. First import into power query looked like:

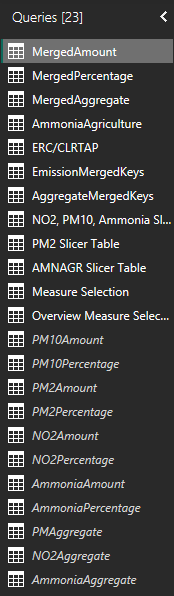


Theres a few problems here that are obvious from the data import, mainly:

* The imported data contains two lines that aren’t data points. These are swiftly removed by removing the bottom two rows of the table.
* ~~Memo tables are also present here, while these are good to have, they present a bit of a problem because they’re not calculated in the national totals. They should be in their own table, ideally~~ After giving it some thought, there is no reason to split these up and visualize the data accordingly. More of my thought process later in this document.
* National totals are present here as well, even though we already have aggregate tables in our model. This is duplication of data and is making the size of our model larger for no real reason.
* The years aren’t written as years, but as the year (thousand tonnes). This forces them to be text instead of date, but then I would create my own date/time table using the CALENDAR() DAX function instead.
* The values, while consistently in thousand tonnes in the statistical dataset, aren’t a good format to visualize data in. The problem here lies in the fact that because they are in thousand tonnes, and Power BI automatically compresses numbers to fit the right amount, the end-user would have to calculate twice in their head for the numbers to make sense. An example would be Ammonia, calculated in 300 thousand tonnes. The visual itself would make it look like a value between 250-350, but the user would need to know that it is in thousand tonnes. Another problem is the nitrogen aggregate calculation being in millions tonnes, compared to everything else being in thousand tonnes. The best solution here would be to get rid of both millions and thousand tonnes, and let Power BI visuals automatically compress them in the visual.

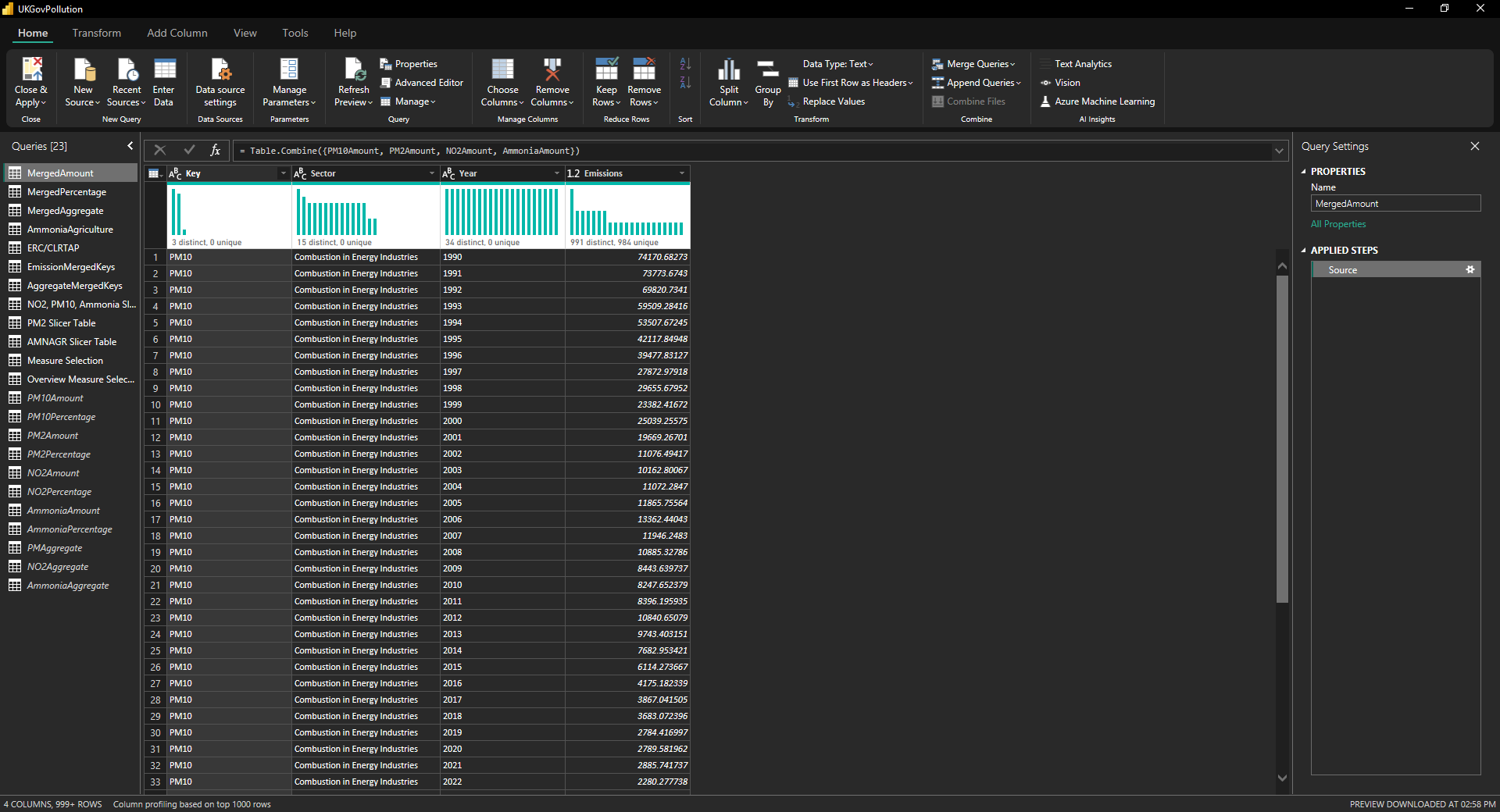
The first step I took in data transformation was to remove the thousand tonnes from all years, make them into a format that’s better for visuals. The first time around, I used a bit of M code to try to transform the data, as changing it one by one is extremely slow. Initially, I used Table.ColumnNames to produce a list of the name of every column in a table and change it that way, but after an entire redo of the data transformation stage, I found a better way of doing it for every table :

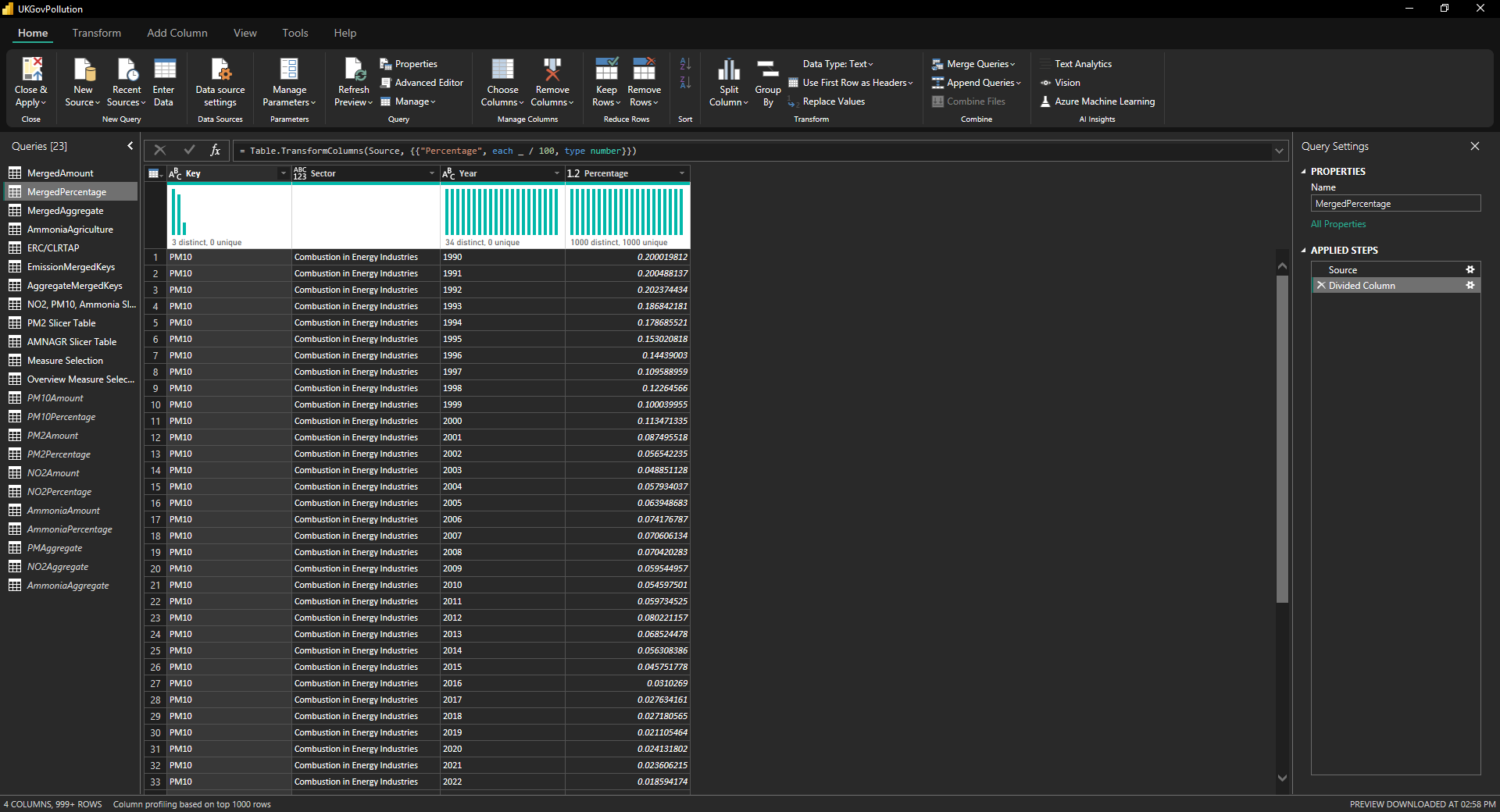
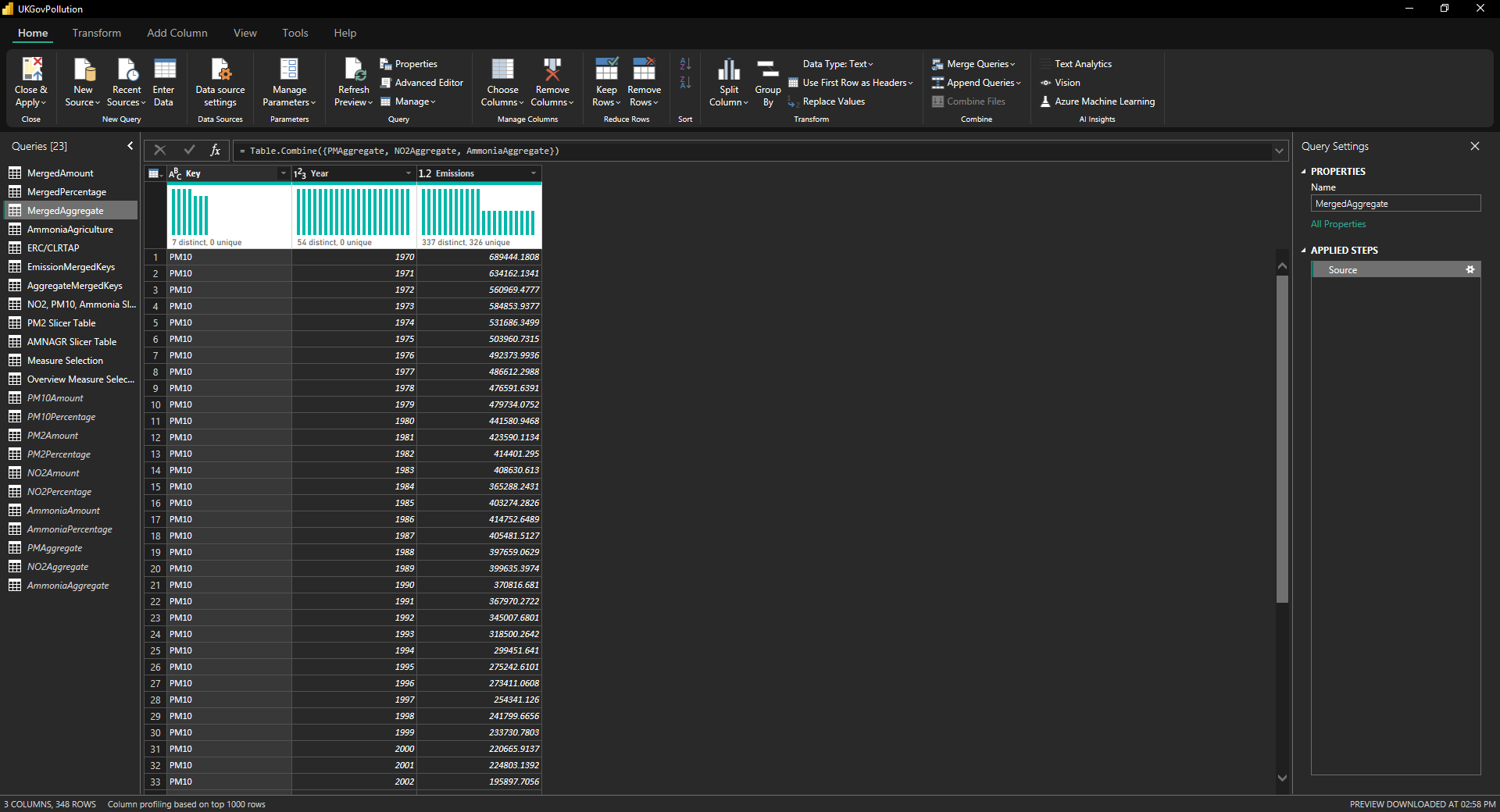
* Step 1: Filter the unnecessary rows out first, so I don’t have to deal with them later and replace “Memo items 1” with “Memo Items” on all tables.  
  
* Step 2: Unpivot other columns for the Sector column. This allows me to make a couple important changes I pointed out earlier, such as multiplying emissions by whatever desired value and removing the “thousand tonnes” from the year column. Pretty quick and easy compared to using Table.ColumnNames and using a whole bunch of unnecessary M code to achieve what I want, though I wouldn’t do it if the \*only\* reason was to rename the columns because that would be unnecessarily bad on performance. Because it was in the intended data format, it made doing these changes very easy.  
    
  
* Step 3: Add a “Key” column using “Column from Examples” and use it to give each table a unique key before I merged them later down the line.   
    
    
    
    
    
  After doing these changes to all tables and a few different ones to the aggregate tables, I came up with a set of tables I’d use for data visualization

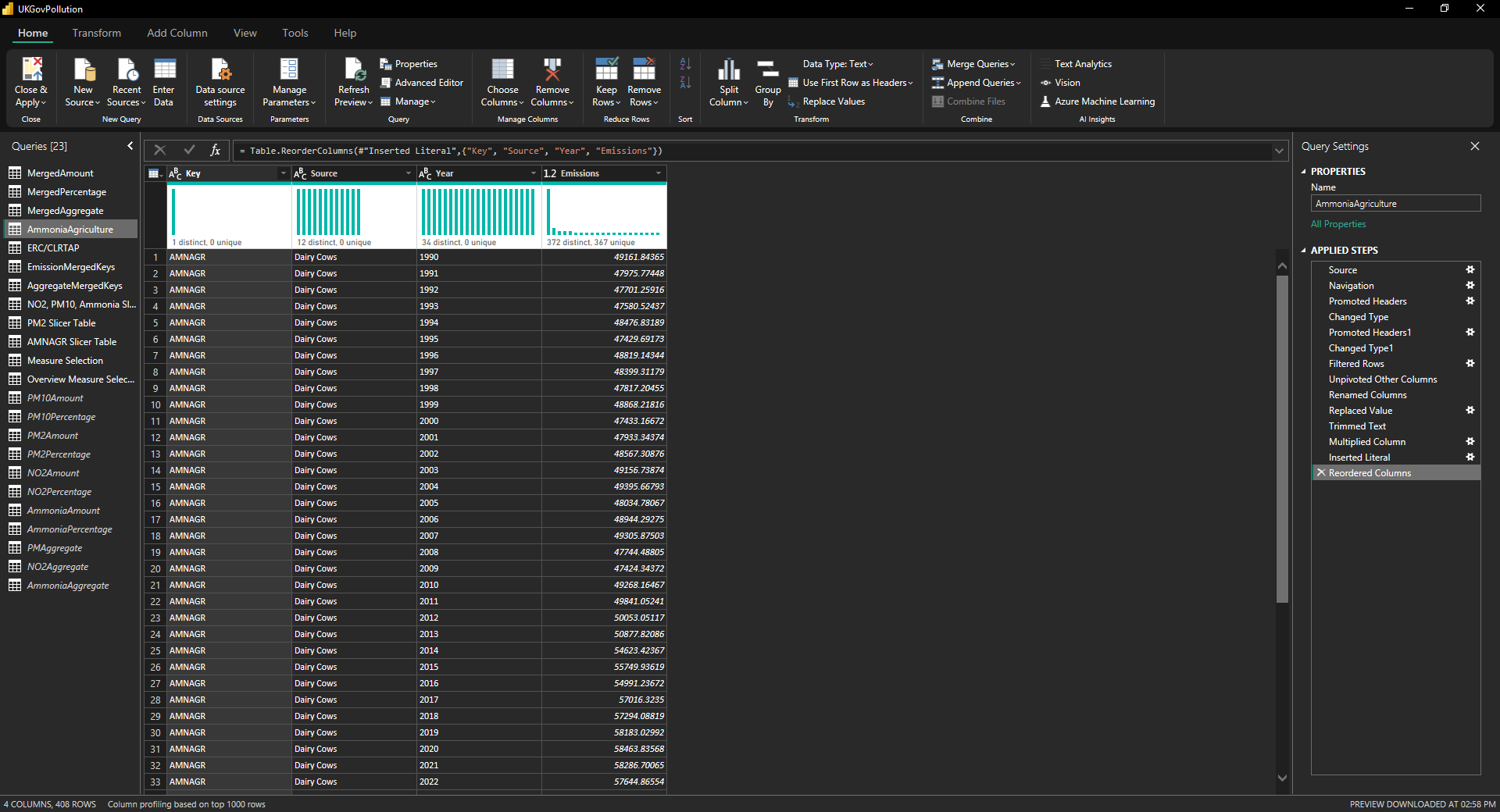


I merged all of the smaller tables into three merged tables:

* MergedAmount: This table contains values for PM10, PM2.5, NO2 and Total Ammonia per year per sector. Merging all the tables into one is best practice as it is the closest I can come to a star schema like design, though this data isn’t very suitable for it unless you merge everything together. The merged table also includes the Memo table, because after a bit of thinking, I realized I didn’t really need the memo table to be different than the Merged Tonnes table, especially if the user can sort by Sector. Ideally, the user uses the aggregate tables for the national total, instead of using the sector total for the same job. I will adjust the visualizations accordingly to better fit that model.



* MergedPercentage: This table contains percentage values for PM10, PM2.5, NO2 and Total Ammonia per year per sector. These emissions are in decimal numbers, but are already percentage values. This means we don’t have to do any calculations for it.  
  
* MergedAggregate: This table contains totals for all emissions. These totals are of many types. For example, in aggregate type there are two entries per year for NO2. These are “NO2 Total” and “NO2 Compliance Total”. The former is the total level of nitrogen oxides released in the atmosphere, while the compliance total is as required by the National Emission Ceilings Regulations, which exclude agricultural sources for their assessment of reductions. Both of these can be important for analysis, and thus are included. The ERC limits have been taken out of these aggregate tables and made into their own table called ERC/CLRTAP and have their own set of keys for easier filtering.  
  

I also have separated agricultural ammonia, mainly because I don’t want to have all of its different sources mixed in with the sources from the merged tables. PM2.5 got away with it because it only had 3 columns like that. Agricultural ammonia will get its own filter and visualization for easy analysis.  
  


The next few tables are fairly self-explanatory:

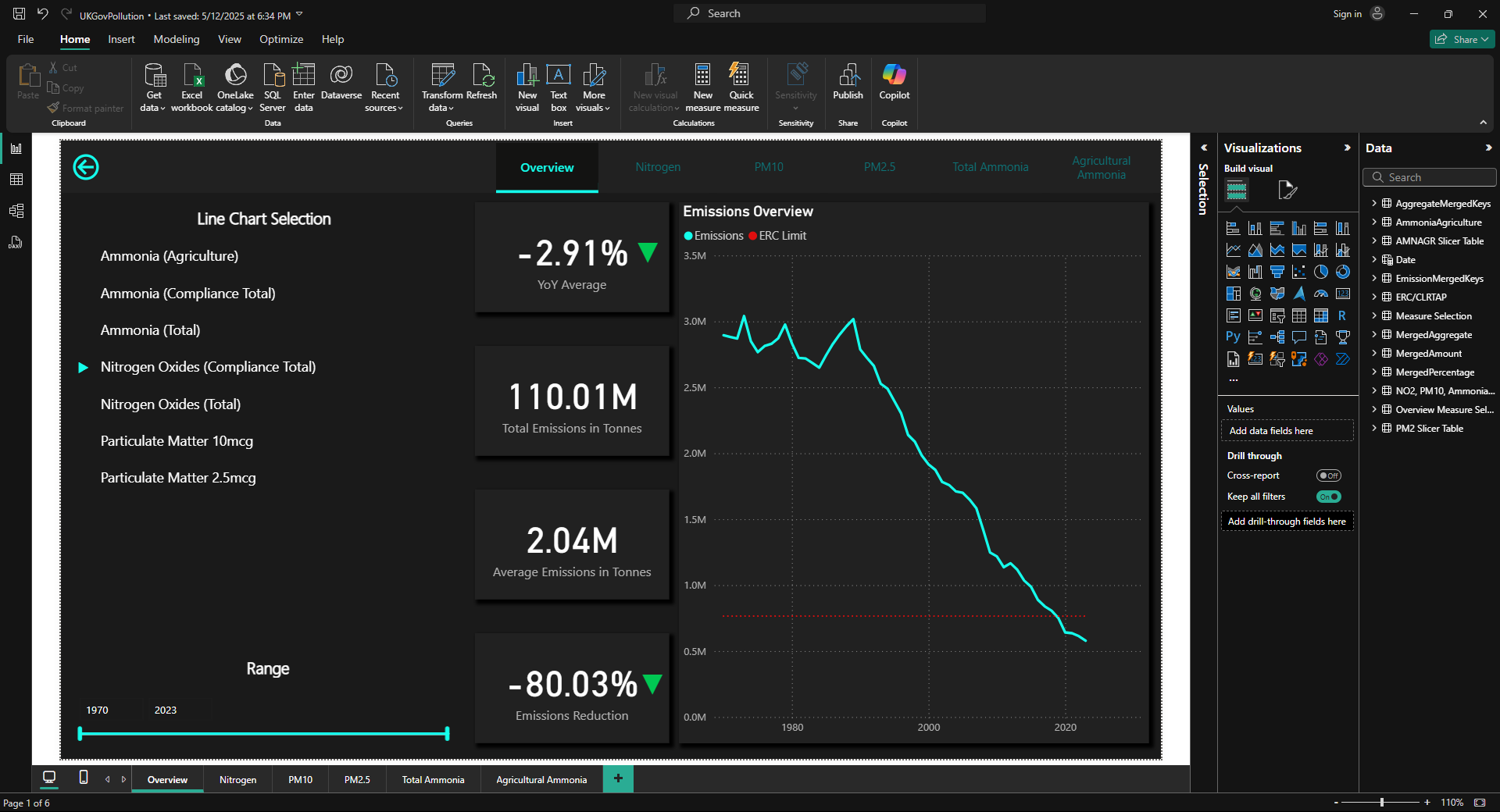
* ERC/CLRTAP: The limits that are given by the National Emissions Ceilings Regulations and the Convention on Long Range Transboundary Air Pollution. Both of these have different limits, albeit very close. For more information on what they include/exclude, use the reference links at the end of this document.
* EmissionMergedKeys : The keys that are used in the MergedAmount and MergedPercentage tables along with their Emission Types mentioned.
* AggregateMergedKeys : The keys that are used in the MergedAggregate tables. The reason they’re different from EmissionMergedKeys is because the aggregate tables have different values for NECR-compliant values and agricultural ammonia per year, so it is easier to have a set of keys for them instead.

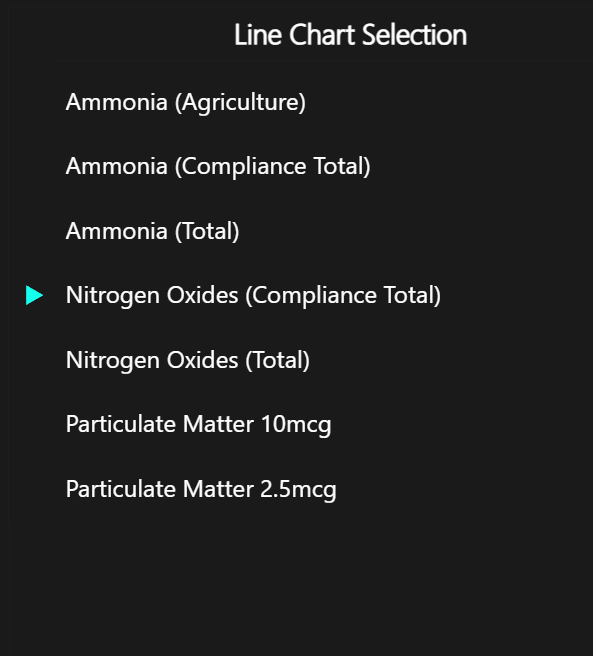
Now come the next set of tables, which consist of dummy tables or duplicated tables which don’t have any relationships specifically to make visualizations work.

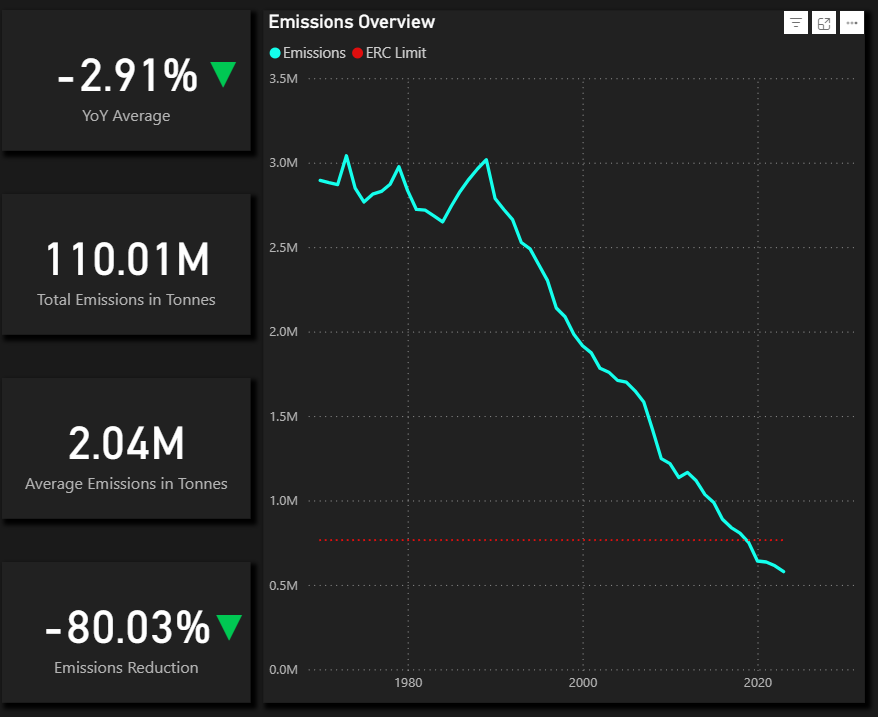
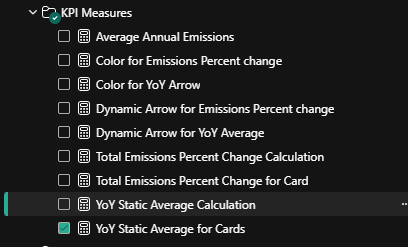
* NO2, PM10, Ammonia Slicer Table: Contains a dummy table containing all sectors for these three pollutant types. I couldn’t filter and use the keys table because it had to be completely unrelated to the other tables, while still being able to be filtered by the slicers.
* PM2 Slicer Table: Contains a dummy table containing all sectors for PM2.5. Power BI doesn’t allow dots (“.”) in table names, so it is PM2 instead of PM2.5. These are different from the first set because they have more sectors.
* AMNAGR Slicer Table: Contains a dummy table containing all sectors for agricultural ammonia. Same reason as PM2.5 and PM10, with unrelated tables and different sectors.
* Measure Selection: Contains a dummy table that has all different pollutant types for aggregate tables.
* Overview Measure Selection: Contains emission types and ERC limits, used as a dummy table as well for visualizations.

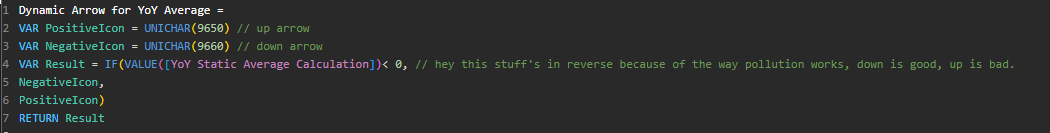
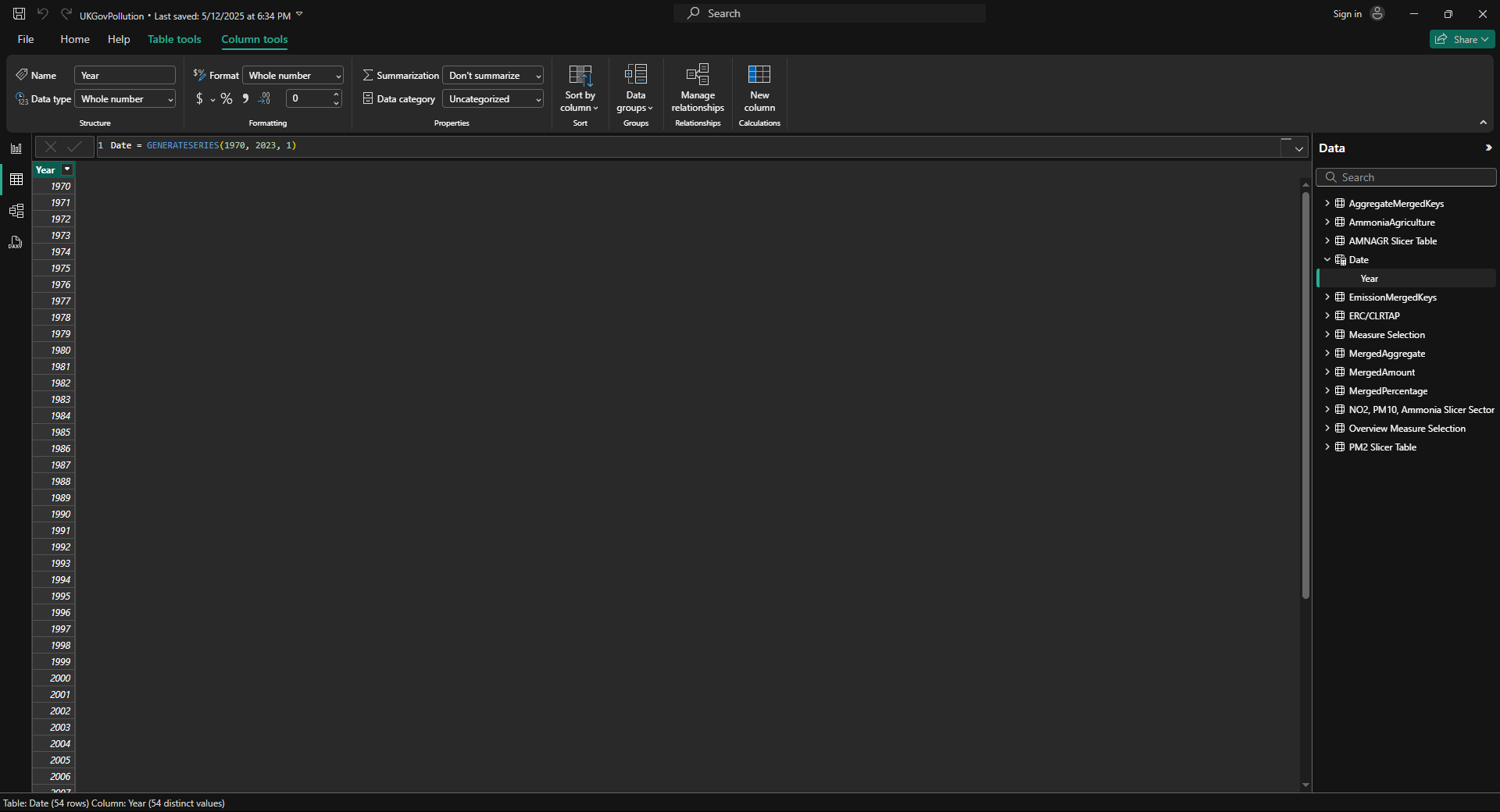
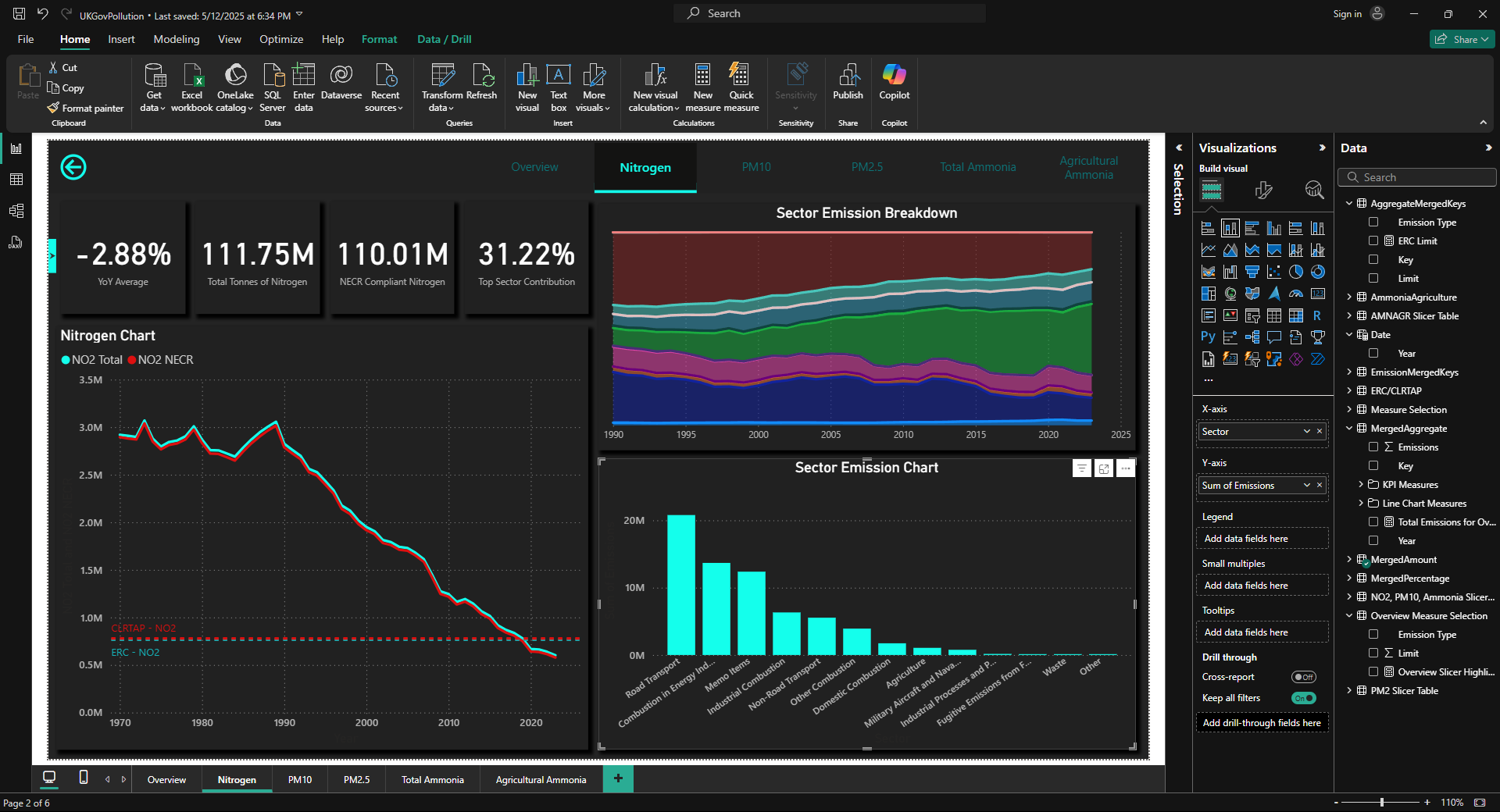
With that being done, we are ready for making a few measures and start visualizing data.

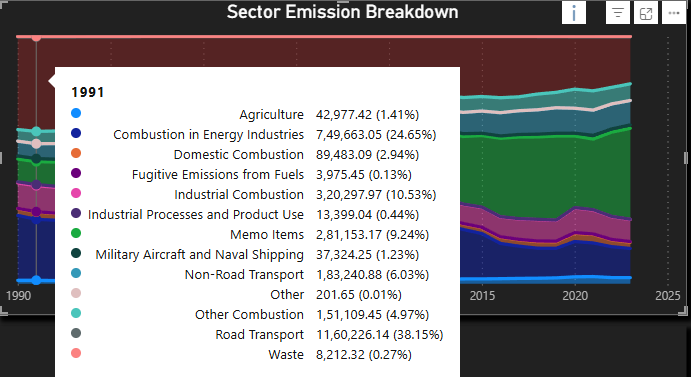
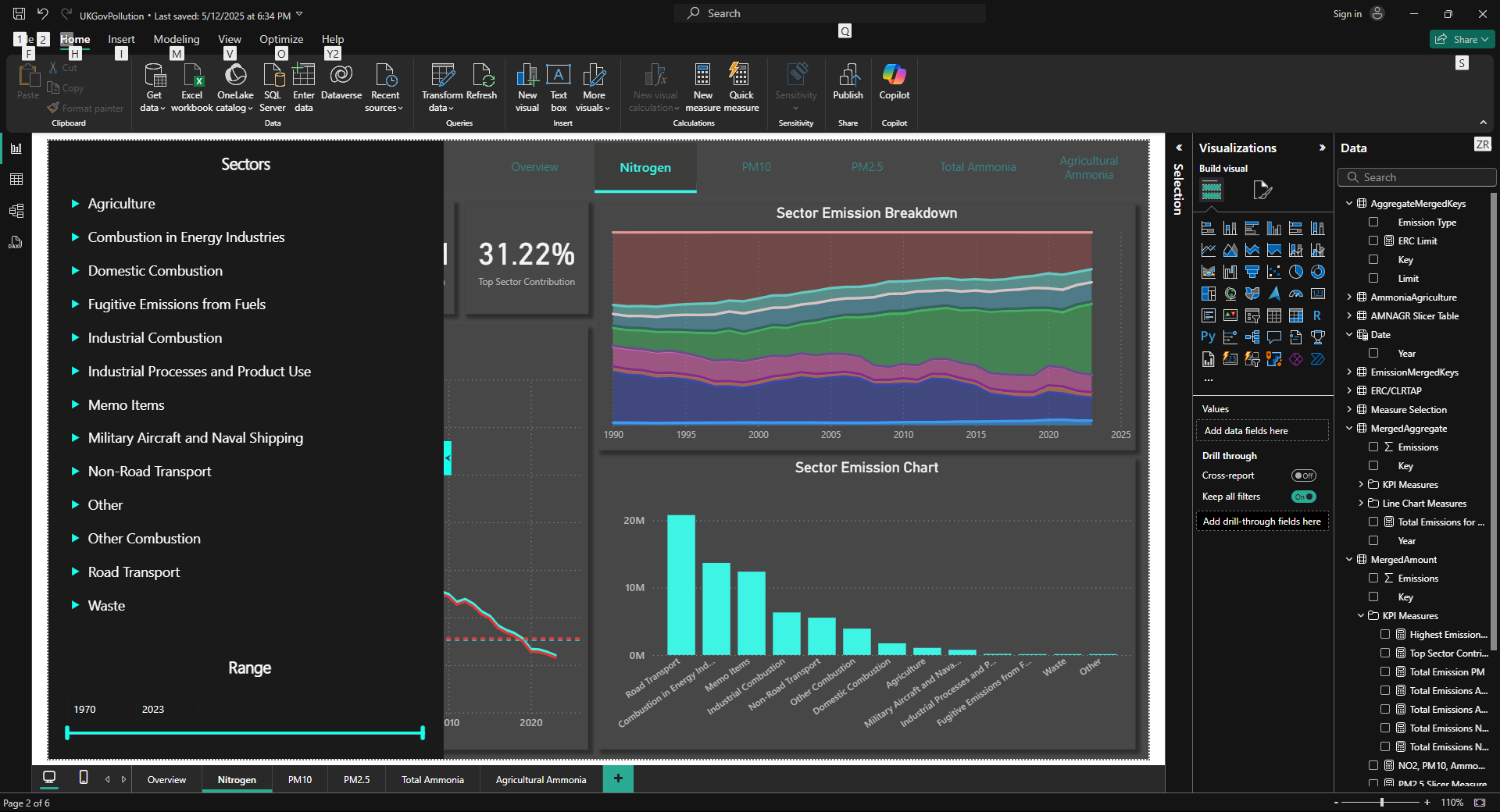
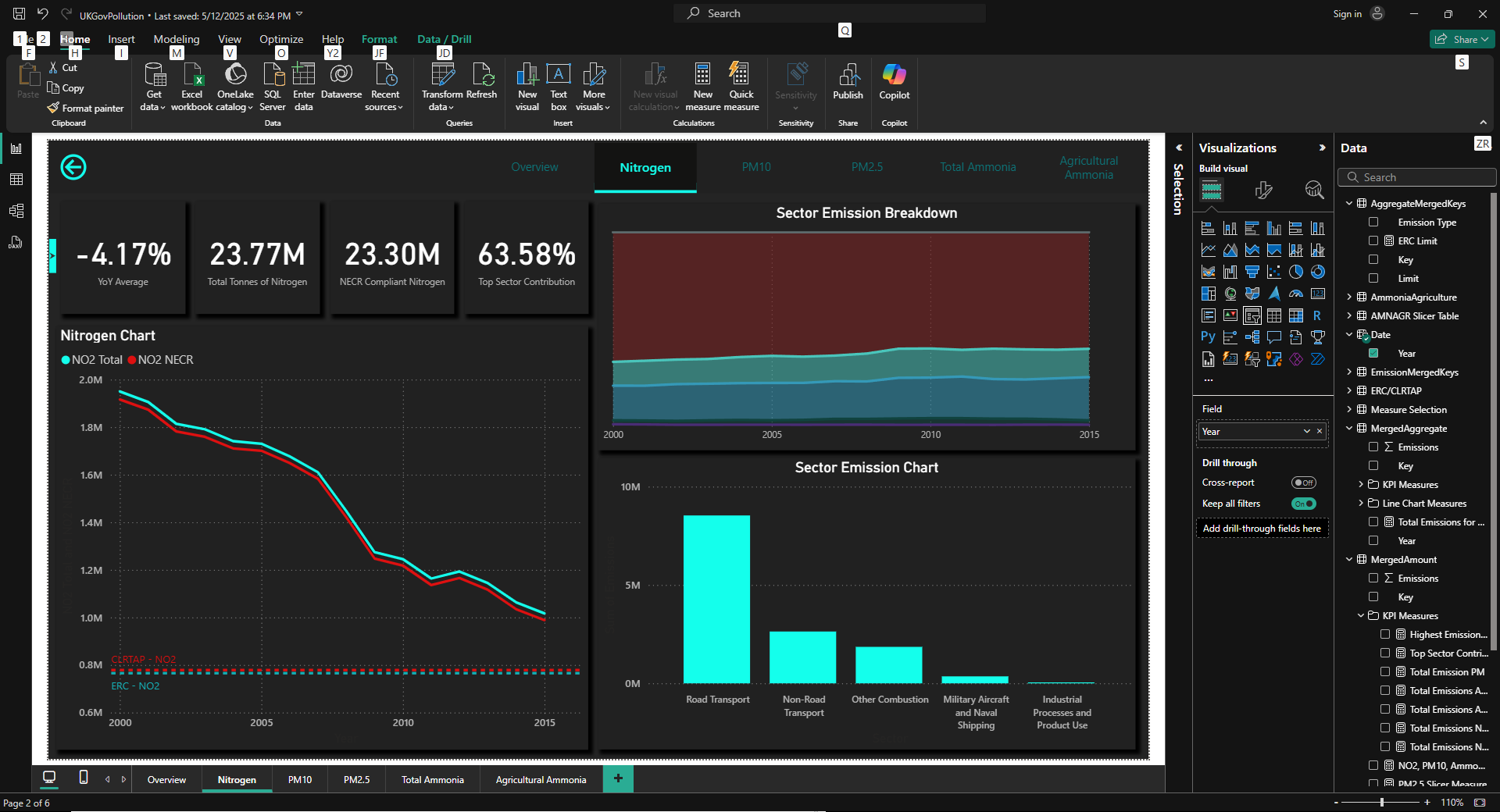
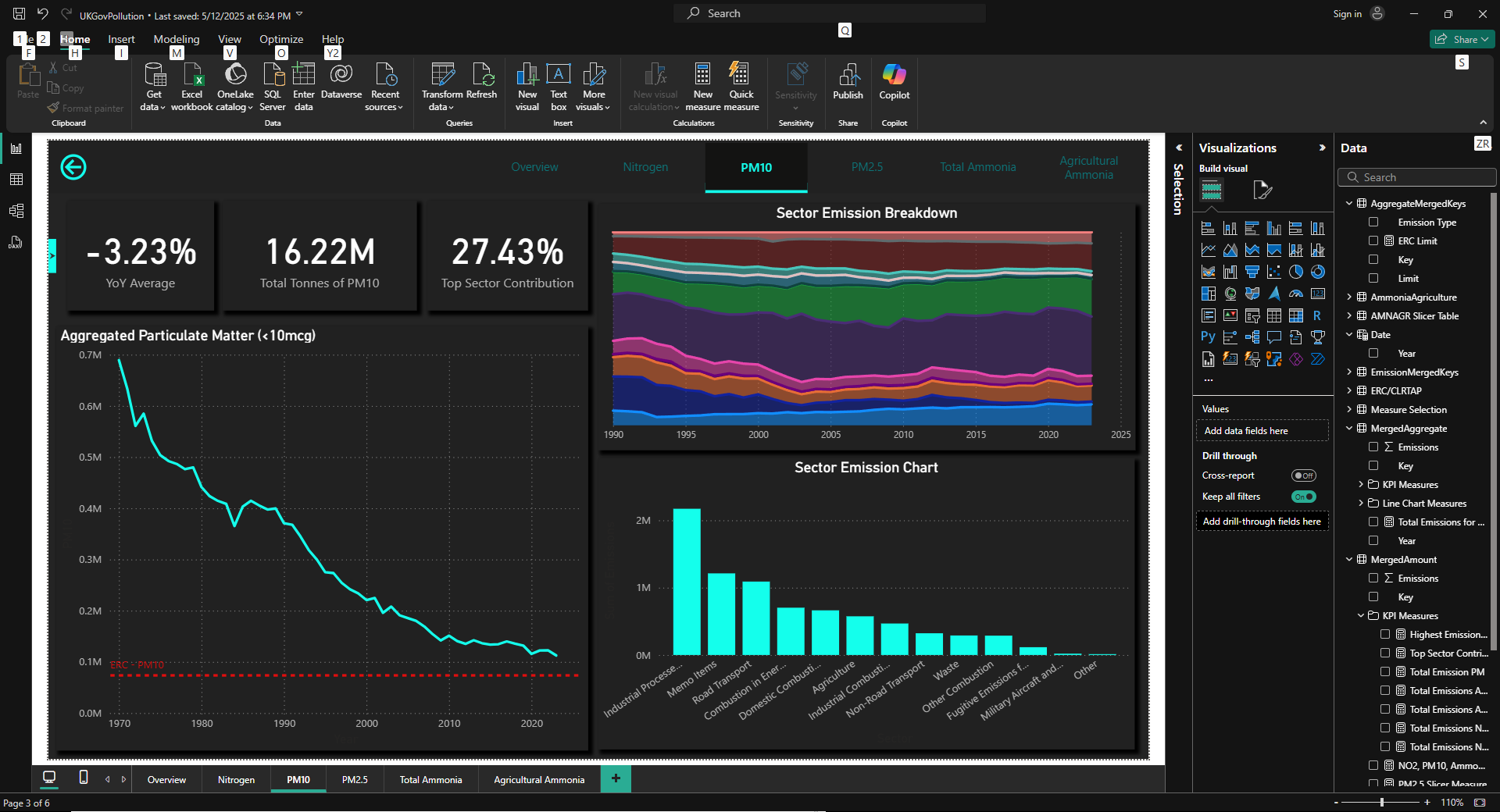
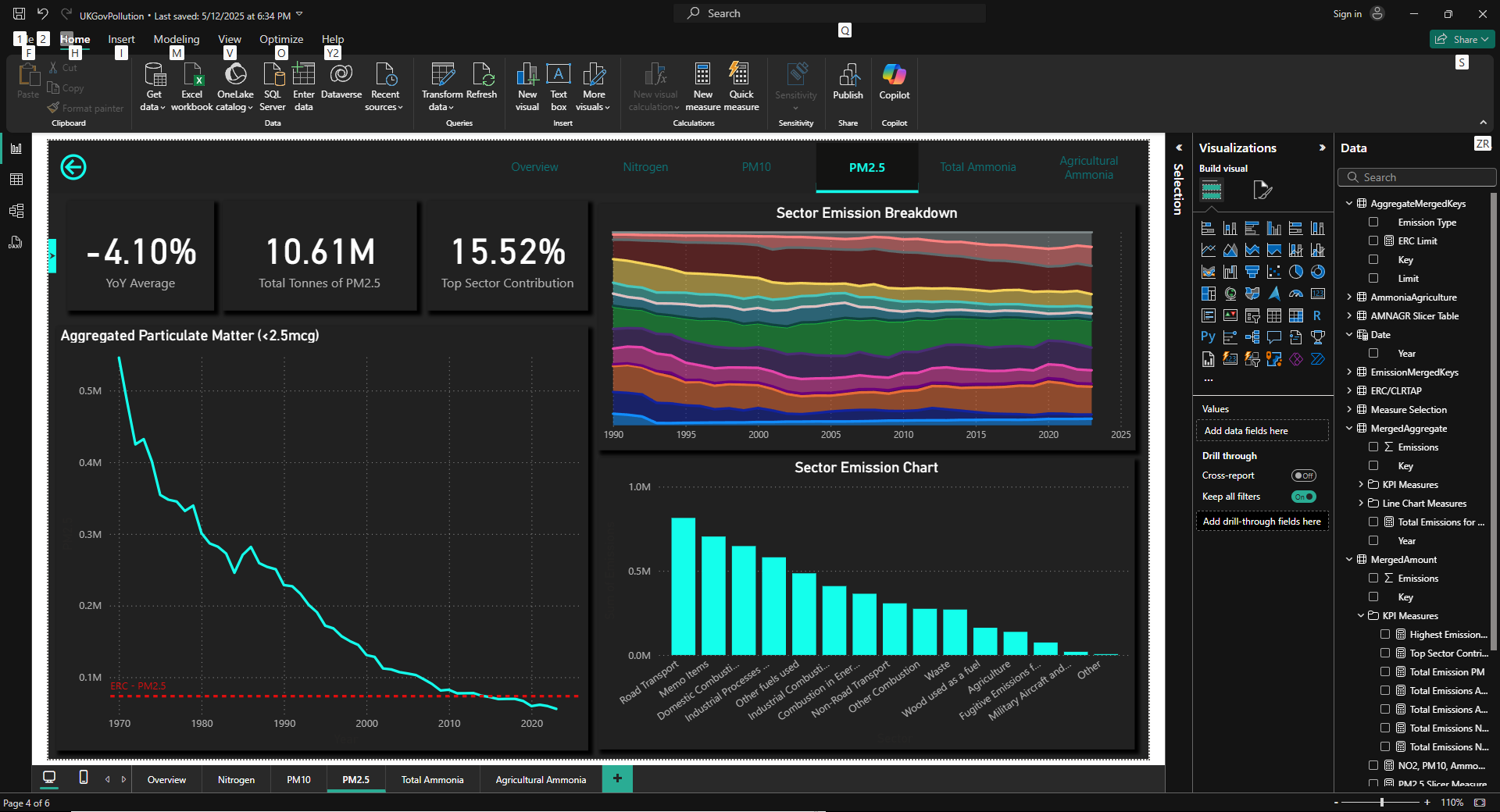
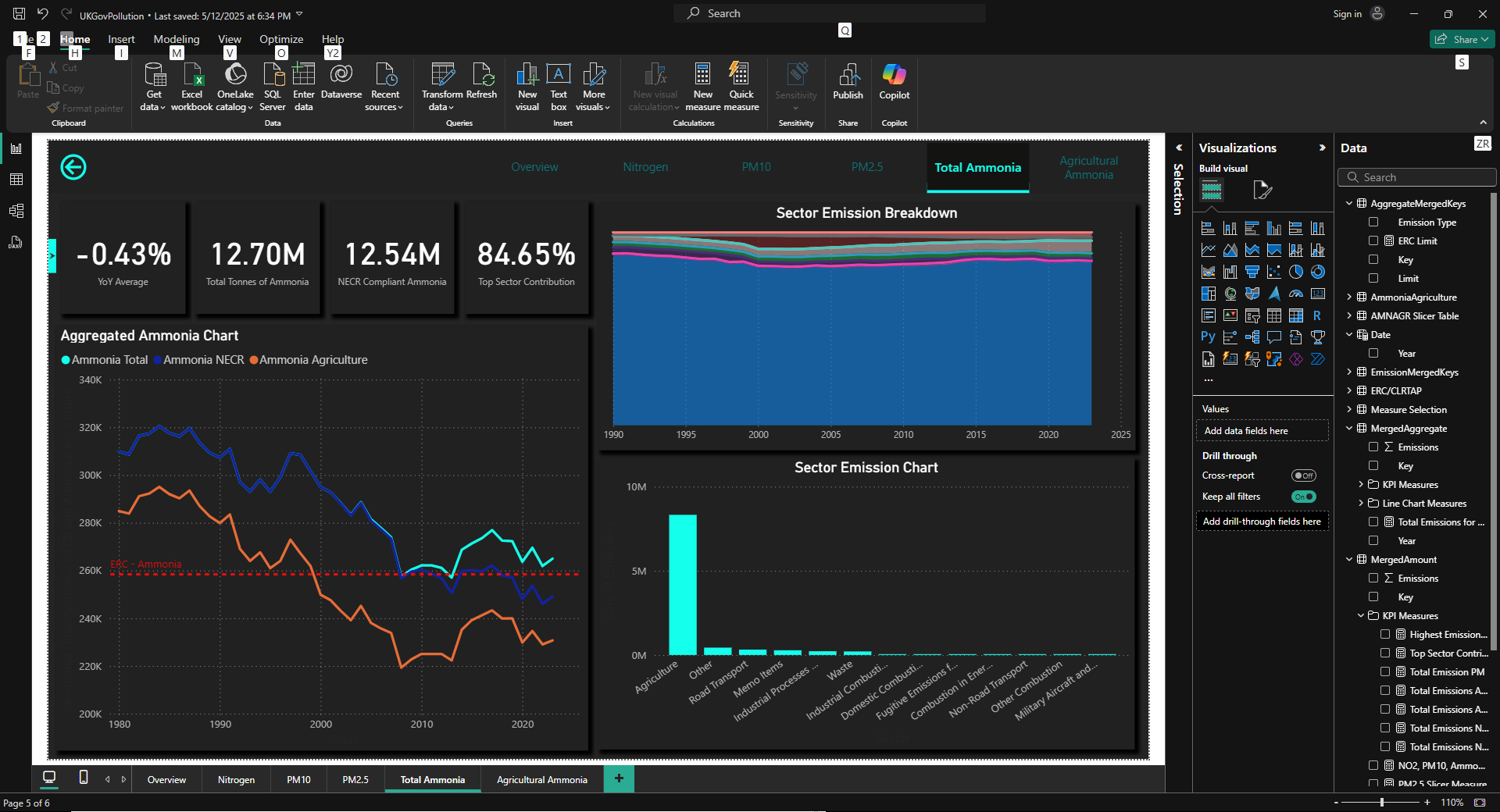
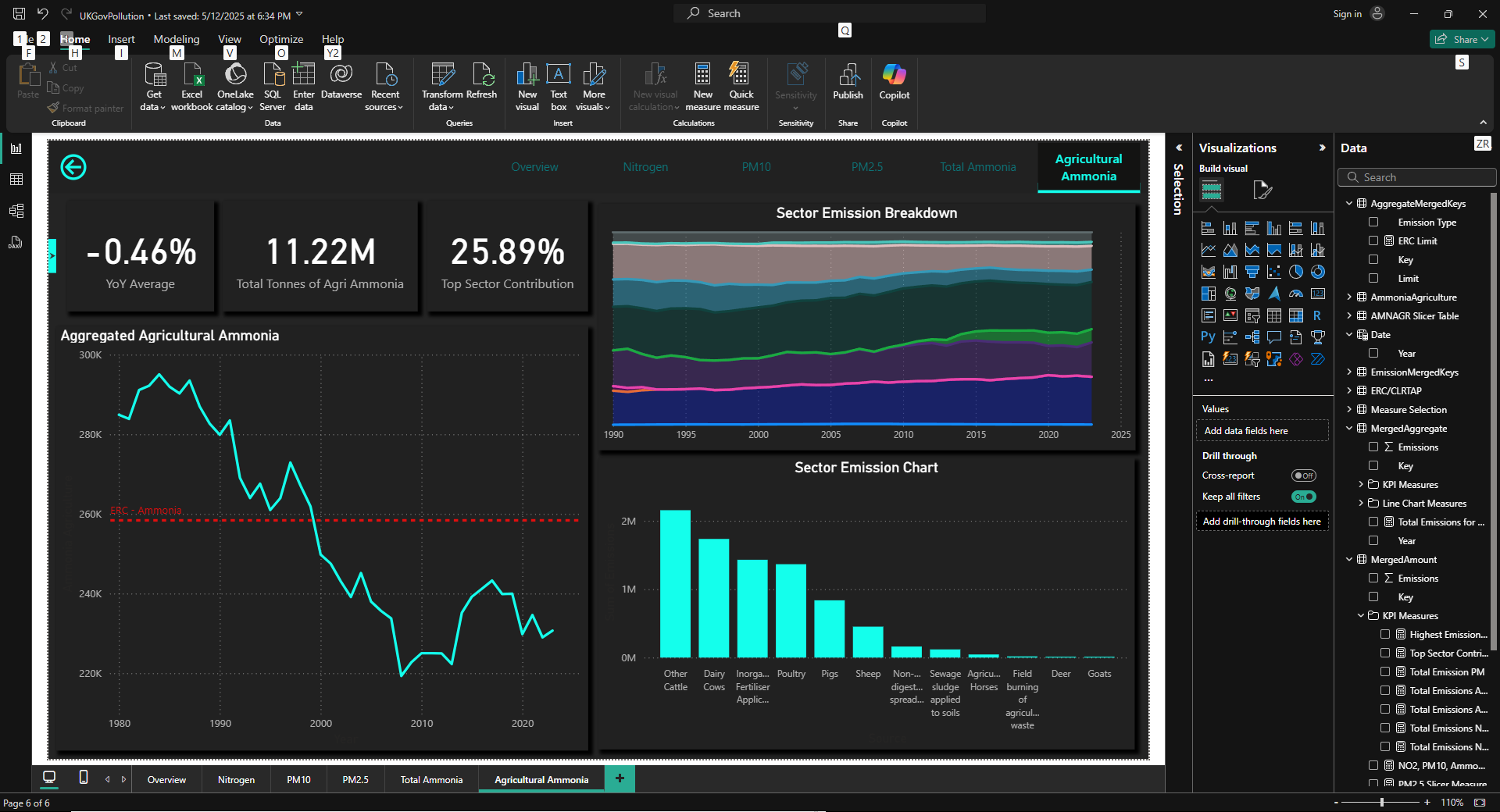
# Data Visualization

First step of visualizing this data was, deciding what pages I would actually want on the report itself. Originally I thought having four charts and KPI cards for everything would be good, but I changed my mind and overhauled the page entirely.   
  
  


This is the final overview page, it contains a slicer that allows adjusting between different emission types, a slicer that has a year range for convenient year adjustment and 4 KPI cards that allow for important information to be shown. There are arrows that dynamically switch between green and red. The main reason the green arrow is pointing down is because less pollution is better. This also shows why I needed those dummy tables to begin with, which I will explain now:  
  
  
  
These are the pollutant types in a slicer. Slicers themselves don’t allow changing it anything other than a checkbox, so the arrows are actually a bar chart using the dummy tables so we can show all values at once and checks if the value in the slicer is selected to light the sector up.  
  
The bar chart has its bars removed and only uses error bars, allowing it to show up as a nice looking visual. From there, we move onto the KPI cards and the line chart.

  
  
YoY Average here doubles up as a current YoY when only one year is selected, but it adjusts with the year range slicer and can allow someone to analyze data that shows larger spikes in either an increase or decrease in pollution. Total emissions is self-explanatory, shows the total amount of pollution in tonnes. Average emissions is another useful metric when working with year ranges and shows the average amount of emissions in a timeframe. Emissions reduction shows a total reduction in emissions as a percentage, also incredibly useful when working with year ranges. All of these are DAX measures, which I sorted in a folder.  
  
  
  
These are mostly self-explanatory, the only thing to explain here is the difference between “Calculation” and “for Cards”. The reason there are two of these measures is because the Calculation measure returns the value of the measure, while the second measure changes it into a different format. An example is:

   
  
This DAX measure creates a table that has the current year, previous year and calculates YoY average overtime, and returns a single value of the average of YoY reduction as a decimal number.  
  
  
  
This merely takes the number the measure above returns, and changes it into a percentage value to be displayed into a percentage. The reason this is important is because the dynamic arrows need a number to actually change color and direction. Their code is:  
  
  
Because FORMAT() changes it into a text string, it isn’t suitable for dynamic arrows. Those are all the KPI cards. Now, the line chart, it is the simplest part of the overview page. It is just aggregated emissions over time. I made a separate date table for best practice for convenience.  
  
  
  
Since the data is only in years, it made more sense to generate a series to act as a date table instead and connect it into the model to make it work. Anyway, the line chart also shows the ERC limit for the currently selected pollutant, which was an easy DAX measure to create.  
  
  
  
From here, we are done with the overview page and we move to the pollutant pages. I’ll only show one, because they have the same design. The first thing to note is the page navigator that allows us to switch pages.  
  
  
  
This has the names of all pollutant types present here, a back arrow that takes you to the overview page, the same as pressing on the Overview page would do. Each bit of text slightly increases in size on hover and reduces on click, to give it an animated button feel. The line at the bottom is actually another page navigator that is used to make a line on the selected page. It is a great way of making it, because it can very easily be expanded by just adding more pages onto the report. Moving onto the pollutant page, I’ll describe the nitrogen page as an example.  
  


These have the same YoY KPI cards, and can be easily filtered on the right between the total NO2 and the NECR complaint NO2, which are cards for the aggregate table. The reason for it is because the stacked area chart is incredibly potent at providing information, though I’ve made a “Top Sector Contribution” which works over a range of years. The reason it doesn’t show the top sector, is because sector names are massive and also, it is right there in the bar chart on the left.   
  
  
  
The aggregate table also has CLRTAP limit present on it, which is not present on the overview page. For convenience, there is a small button besides the YoY Average card with an arrow indicating it can be opened.  
  
  
  
This opens up the slicer pane, which allows the user to filter the data using sectors and perform thorough analysis as they like.  
  
  
  
The year slicer is also present here to adjust year ranges. Either the background or the button can be pressed again to exit back to the page. Here is an example of a filtered page.  
  
  
  
  
Here are how the rest of the pages look, though I won’t go into detail because by design, they are the exact same.  
  
     
  
This concludes the visualization of the report.

# Conclusion

In conclusion, this report allows someone to really dig deep into analyzing the pollution data provided by the UK government. I might do my own analysis of this data, though having seen it so many times during testing and visualizing it, I have a fair idea of what is going on here. But for anyone else, this could be a useful tool for reporting about pollution reduction/increase, checking out historical data and seeing which sector have been the worst offenders for each pollutant. Personally, I had a lot of fun making a pretty decent looking report out of this data and finding more information that isn’t present on their website by just working on the data given to me. Learning how to make DAX measures that worked and didn’t work, making tens of measures and scrapping them because I found a better way of doing it and almost two whole overhauls to the project gave me a lot of experience on how to tackle a problem like this again. All in all, oodles of fun making it. I might make smaller adjustments if I think it would be a good idea, although for now I’m going to start work on a new project.

# References

<https://www.gov.uk/government/statistics/emissions-of-air-pollutants/emissions-of-air-pollutants-in-the-uk-nitrogen-oxides-nox>

<https://www.gov.uk/government/statistics/emissions-of-air-pollutants/emissions-of-air-pollutants-in-the-uk-particulate-matter-pm10-and-pm25>

<https://www.gov.uk/government/statistics/emissions-of-air-pollutants/emissions-of-air-pollutants-in-the-uk-ammonia-nh3>

<https://www.gov.uk/government/statistical-data-sets/env01-emissions-of-air-pollutants>

DEFRA report: <https://uk-air.defra.gov.uk/reports/cat07/1009030925_2008_Report_final270805.pdf>  
NFR codes: <https://naei.energysecurity.gov.uk/sites/default/files/cat07/0910130851_DA_AQ_Inventory_Report_2007_Appendices_Issue_1.pdf>