I. Datasets and motivations

Energy has become a main concern for a lot of people today. Enedis, through its website, has open a huge quantity of data to public. But due to the large amount of data available, this data can become very fast unfriendly and it can become very frustrated to navigate through it.

A lot of graph can appear unappealing for a person not-initiated, so we wanted to create a tool that would give a very simple but understandable overview to every curious people. For this reason, our design was based on two types of graphs, understandable by most people: heatmaps, and node-link graphs

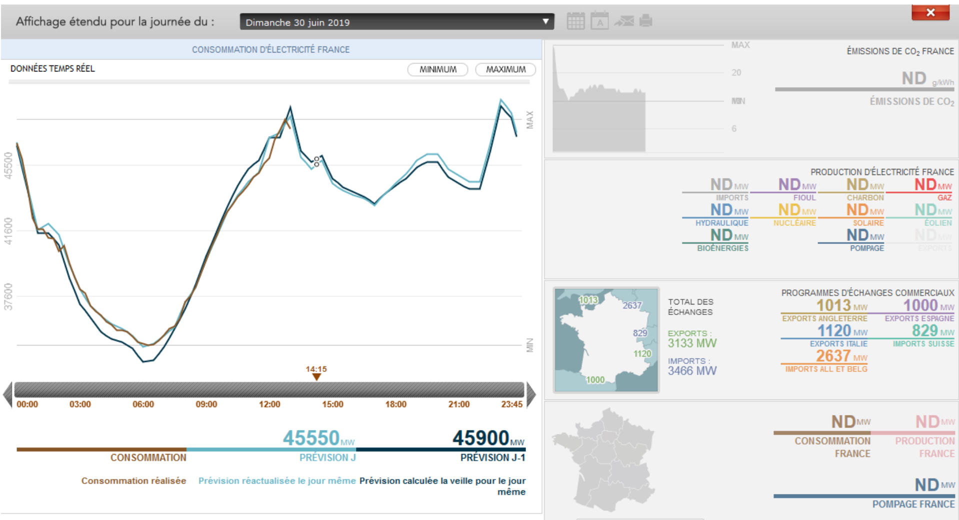


Figure : An exemple of visualisation board from Eco2mix by RTE

As explained, the data are directly available from Enedis dedicated website. Every user can find them here: https://data.enedis.fr/page/accueil/. There is a total of 60 different dataset spitted into 6 categories:

- Energy: giving insights on the production and consumption of energy in France, with different time splits and geography splits
- Environment: giving the electrical loading station for cars
- Exploitation: giving information about lines installed and blackouts
- Infrastructures: giving the architectures of low voltages and high voltages lines
- Electricity market: the datasets about Joule losses and market adaptation.

These 60 datasets gather together millions of lines of data. One of the first tasks was to explore these data and decide which ones we will use for the project. We knew that there were a lot of data available, but we underestimated at first the time it would take to explore all these datasets and chose only data we wanted to dig more.

As our target was to target most people, we went simply on the data related to Energy. The main questions we wanted to answer were as following:

- From where is coming the Energy we are using in France?
- How it is produced?

- What are the areas that produce more than what they consume?
- Who are the good examples to follow? Who are not?
- How all of this is evolving in time?

But even after limiting the field of possibility, we had to choose also the scale we wanted to present the data as a lot of format was available. What temporality? What geographical mesh? Again, it is simplicity that drived our choices and we went to the largest scales:

- Year for the temporality
- Region for the geographical mapping.

Let's now dive into the data! We selected mainly two datasets from the ones available in the Energy section from Enedis open data.

Consommation-electrique-par-secteur-dactivite-region.csv

Let's have a look at how the data is represented here. There is a total of 36 columns and 84 rows. Each row represents insights for a given region at a given year. The columns we are selecting are listed below:

AnnéeNom régionCode régionConso totale Résidentiel (MWh)Conso moyenne Résidentiel (MWh)Conso totale Professionnel (MWh)Conso moyenne Professionnel (MWh)Conso totale Agriculture (MWh)Conso totale Industrie (MWh)Conso totale Tertiaire (MWh)Conso totale Secteur non affecté (MWh)Geo ShapeGeo Point 2D

Geo Shape and Geo Point 2D have been used to build the heatmap.

The year has been mapped to our chronological selector, the region code to make a region selector. Years are going from 2013 to 2017.

Finally, total consumption and average consumption are used as insights that we will use as colour encoding for our heatmaps.

Données_economix.csv

This is the second dataset used. It summarised, again at a year scale, the different vector of production for each Region between 2013 and 2017. This dataset has 22 columns, all used and 60 lines.

Let's go quickly through the data here:

Année / Code / région : As before, data used to map the chronological selector and region selector

Production_totale : Used as a feature to encode the colours of the heatmaps.

Production_nucleaire / Production_thermique_totale / Production_hydraulique / Production_eolien / Production_solaire / Production_bioernergies : This data has been used in 3 different ways :

- They are mapped in the linked-node graph to the size of the nodes, in order to illustrate the global proportion of energy produced in France

- They can be also mapped to the different histograms available below the heatmaps, to see the evolution over time of the production of each type of energy.
- Finally, we used them to build a new feature, CO2 production. We will come back on this in the next section of this report.

Grande_industrie_PME_PMI / Energie_industrie_agriculture / Chime_parachimie / Construction_automobile / Metallurgie_mecanique / Mineraux_materiaux / Papier_carton / Siderurgie / Autres_industries / Excedent_Deficit : All this data are used to build another node link graph where we see insights on consumption instead of production.

Other features built

CO2 production

CO2 production is a popular topic. We wanted to give the intuition to people on how is evolving the CO2 production with the mix energetic for a given region.

In order to build that feature, we had to know how much CO2 is produced for each type of energy in our energy mix. To get this overview, once can visit [The Intergovernmental Panel on Climate Change website](), where experts gather all the required data. A table available in [the Annexe III : 1329IIIANNEXTechnology-specific Cost and Performance Parameters]() gives us all the information we wanted about energy source and CO2 emission. For our feature, we used Median values on lifecycle emissions.

Options	Direct emissions	Infrastructure & supply chain emissions	Biogenic CO_2 emissions and albedo effect	Methane emissions	Lifecycle emissions (incl. albedo effect)
	Min/Median/Max	Typical values			Min/Median/Max
Currently Commercially Available Technologies					
Coal—PC	670/760/870	*9.6*	*0*	*47*	**740/820/910**
Gas—Combined Cycle	350/370/490	*1.6*	*0*	*91*	**410/490/650**
Biomass—cofiring	n.a.[ii]	–	–	–	**620/740/890[iii]**
Biomass—dedicated	n.a.[ii]	*210*	*27*	*0*	**130/230/420[iv]**
Geothermal	0	*45*	*0*	*0*	**6.0/38/79**
Hydropower	0	*19*	*0*	*88*	**1.0/24/2200**
Nuclear	0	*18*	*0*	*0*	**3.7/12/110**
Concentrated Solar Power	0	*29*	*0*	*0*	**8.8/27/63**
Solar PV—rooftop	0	*42*	*0*	*0*	**26/41/60**
Solar PV—utility	0	*66*	*0*	*0*	**18/48/180**
Wind onshore	0	*15*	*0*	*0*	**7.0/11/56**
Wind offshore	0	*17*	*0*	*0*	**8.0/12/35**
Pre-commercial Technologies					
CCS—Coal—Oxyfuel	14/76/110	*17*	*0*	*67*	**100/160/200**
CCS—Coal—PC	95/120/140	*28*	*0*	*68*	**190/220/250**
CCS—Coal—IGCC	100/120/150	*9.9*	*0*	*62*	**170/200/230**
CCS—Gas—Combined Cycle	30/57/98	*8.9*	*0*	*110*	**94/170/340**
Ocean	0	*17*	*0*	*0*	**5.6/17/28**

Figure: Emissions of the selected electricity supply technologies (gCO2eq / kWh)

Interesting facts here: when we take into account all life cycle emissions, we can see that solar panel are not as low producer has expecting, with an average of ~40gCO2eq / kWh, almost 4 times more than a nuclear plant. Also, as expected, biomass and thermal energy mix are the bad elements of low CO2 energy mix, we will come back on that during our data exploration.

Excedent / Deficit production

Another very interesting feature we wanted to highlight is the excedent or the deficit of production for every Region. The main reason is simple: in order to make as few wastes as possible, each Region shall produce the energy it is consuming, in order to limit the energy vanishing through lines by Joule Effect.

To build such a feature, we are simply subtracting consumption from production, for each region and each year. This is quite straight-forward to make.

II. Technical choices

In this section, we will try to enumerate the different technologies used in our project, justify the choice of each of them and finally explain briefly how the different components and interactivities are implemented.

In our visualization project , we used python and many web technologies in order to implement the suggested design : We used python to clean and transform the data , Reactjs to handle and facilitate the interactivity and especially to implement reusable components that can be put together to form more complex components, and finally D3js to create the different graphs.

D3js

D3.js is a low-level JavaScript library for manipulating documents based on data. D3 helps to bring data to life using HTML, SVG and CSS. D3's emphasis on web standards gives you the full capabilities of modern browsers without tying yourself to a proprietary framework, combining powerful visualization components and a data-driven approach to DOM manipulation. We were eager to use the library in our project because we saw that it is a powerful tool for creating different types of visualization, manipulating each component of the graph and making graph interactive through D3js data-driven transformations and transitions.

ReactJS

React, which was built by Facebook, is currently one of the hottest JavaScript libraries for building user interfaces. Some of the greatest advantages for using React are its ability to create reusable code blocks called Components, the use of the Virtual DOM, and a debugging system that quickly tells you if you have any bugs and which component they are in. Moreover, React has a super-fast rendering capabilities.

In React class components, there are multiple lifecycle components that can be used to manipulate the app based on the current state of the user interface. Some reasons we generally might need lifecycle methods are for binding event listeners, asynchronous requests, and improving the application's performance. In our project , we need to make our dashboard highly interactive and sensitive to user behaviour (dynamic data filter , click on maps , barchart , date …) , it was a bit complicated to use only d3js to provide these functionalities in an efficient way , that's why we suggested to use a web framework like React to ensure a high interactive visualization tool

In our project, we used lifecycle methods of react to mount and update the data to the DOM. For example, we used the method, componentDidMount(), which called once and will immediately mount our json file after the component is rendered in the DOM. Besides, our data is highly dynamic and changing state after a certain event like user selection or click events, the lifecycle method, componentDidUpdate(), was very important to ensure that we re-render the new data onto the DOM. Also, in React , we can use the notion of State that help us to share many state variable (year , region , filter) across all the components and update it automatically when there is a change .

Components and interactions

All the react components implemented in our dashboard are created with D3JS , version 4 for the graph , and version 5 for the rest . We used also Boostrap for the html template .

We recall that in our project, we used 3 types of graphs:

- Maps with tooltip
- Barchart
- Graphchart

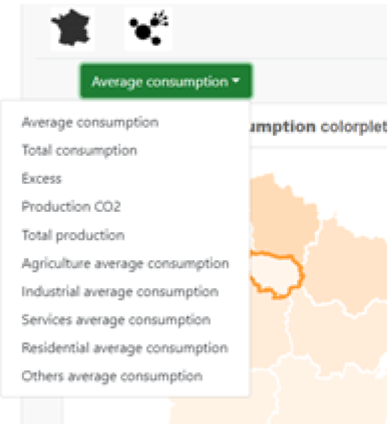
Below the implemented interaction:

- Comparaison between maps through compare/hide menu

Through this interaction, the user can analyse the difference between regions and compare the different statistics. For example, we can select «total production » in the first map, click on "compare", select « Total consumption » in the second map and compare the production/consumption per region.

This interaction was implemented by using state variable (shown / hidden) for each map. The update of this state is done when the button « compare » is activated.

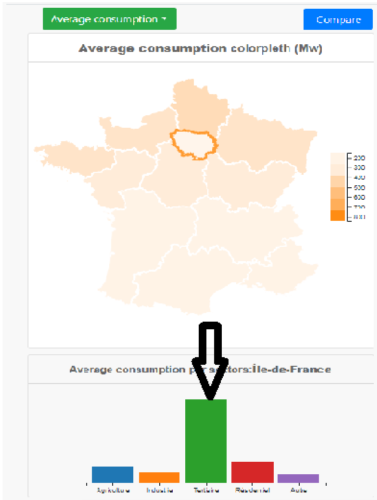
- Dropdown list for view selection:



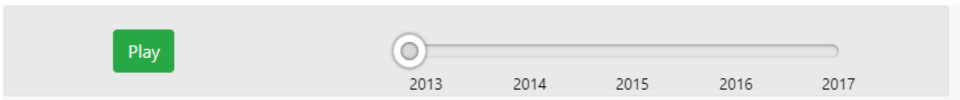
Through the dropdown selection menu, the user can filter the data and focus on a specific energy indicator like Total consumption , excess or CO2 production . The implementation of this functionality was done by filterning the data inside each d3js component .

- Interaction between barchart and map

We offered to the user the possibility to select a specific region and display a detailed view about the repartition of energy production/consumption , depending on his choice .



- Play Button



We added a play button through which the user can visualize sequentially the evolution of a specific energy indicator both in the map and in the barchart.We believe that this kind of visualization will help the user to detect quickly the evolution of an indicator through the color gradation/degradation

Data format

The .csv format of the data was not the more optimized format to use with chronological selector together with the region selectors. For this reason, we included in our pre-processing step a data convertor to build standardized, multi-keys unique json.

The primary key is the main feature to check out. For instance, “CO2 production” or “Average House Consumption”. This feature is interactively selected by the user.

The second key level is only used to encode the histograms. This feature is not always available, so one of the keys, “histogram” point to a boolean that indicate whether we shall show a histogram.

Next level of keys is linked to the year used to show the data on the map.

Finally, the last level of keys is connected to the different regions.

With this architecture, we respect the natural order of interactivity that a user can have with our application: First selecting a feature to show, then checking the chronology and finally the region, from more general to more focus.

As our application is divided in two parts (Node-link graphs and Heatmaps) we also built a second json in order to build the interactivity on the node-link graph, following, again, the natural interactivity sequences (Production -> Field -> Region or Consumption -> Field -> Region).

Heatmap colours

It might look insignificant at first, but the colour ranges have not been selected randomly. At first, all map was showing same green scaling. This has two major drawbacks.

- First one: For one map: Excess/Deficit. A basic green scale was not showing well enough the neutral, deficits and excess of production. For this case, a blue to red scale has been used.
- Second one: Once we come up with two maps for comparison, it was tough to quickly get what map was where (in left and right side). For this reason, we took different colour range for “Production”, “Consumption” and “CO2” topic. That makes a total of 4 colour ranges (if we include the excess/deficit map), that makes it much clearer from start.

III. The application

Let’s see now how the application is behaving, and how it answers the question we were initially asking ourselves.

The application is available online at this address: https://radpc.hopto.org/igr204/

Let’s first have a look at the Node-link graph. It gives very high-level information, but sometimes it is good to get a refreshment here. For instance, if once wonder how is made our energy mix, or where are located the main nuclear plant, he can do it directly from here, has highlighted below:

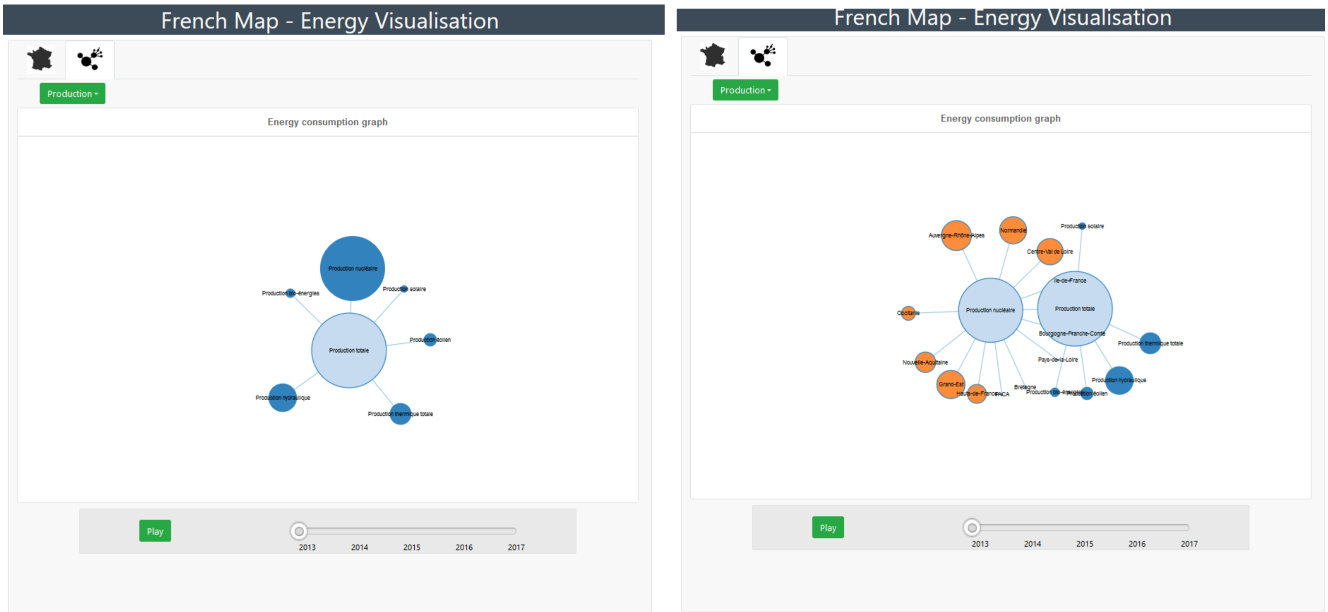


Figure: the node-link graph

We see from this view that the energy produced in France is massively coming from the Nuclear plants, followed by Hydraulic sources and thermal plants. This is not a surprise for us, but it might be a new information for some people. Where are these Nuclear plants? Well... A bit everywhere... But there are 3 regions without their own nuclear plants: PACA, Bretagne and Pays de la Loire.

Let’s now have a look in our heatmaps. As stated, it has a very simple form so everybody can jump into it.

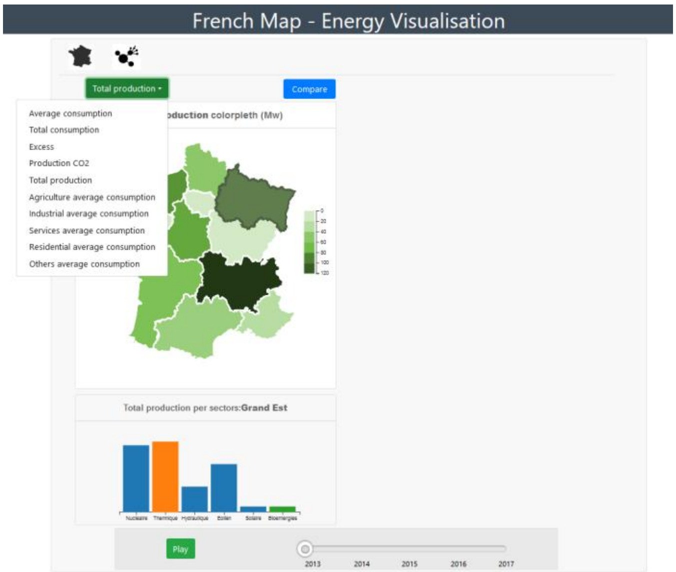


Figure: Basic Heatmap presentation

User has the choice between a list of features to look at, these features have been described previously. When available, a histogram below the map shows the relative variation for each type of energy. The chronological cursor allows to switch the year, or to go on automatically, and finally, a “Compare” button make it possible to add a second screen in the map. Let’s see what we can find here in practice.

As stated before, a standard call by user would be to screen the CO2 emissions. Once we look at it, we might be curious to know why it is evolving in one or the other direction.

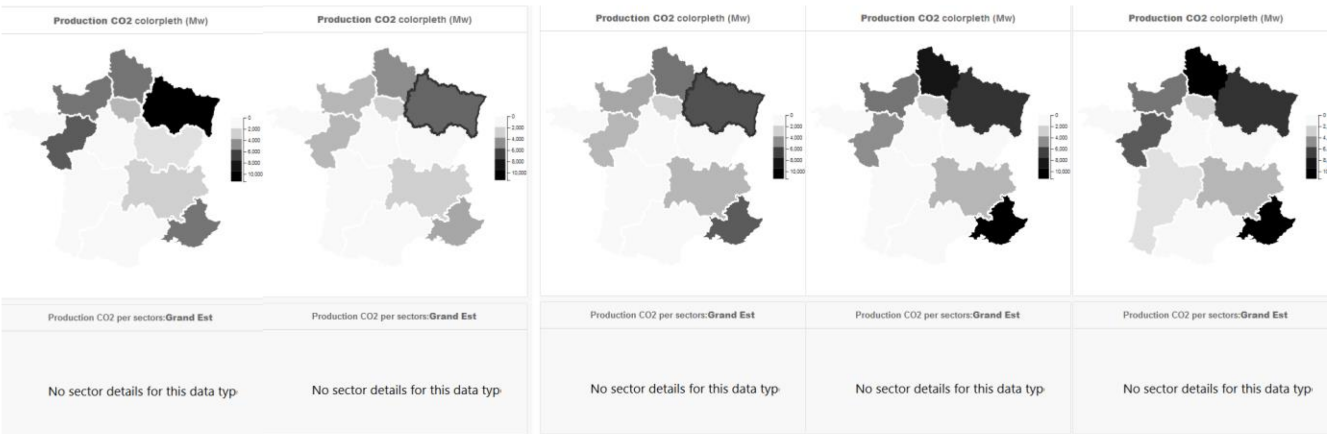


Figure: Evolution over time of the CO2 emissions by regions

We can see clearly from here who are the bad and the good Regions: for instance, Region Grand Est and PACA are the biggest producer of CO2 emissions. Let’s focus on these two regions and try to understand these figures.

For this, we will plot in paralleled the “Total Production” figures, and especially, check how the histograms are evolving in the time.

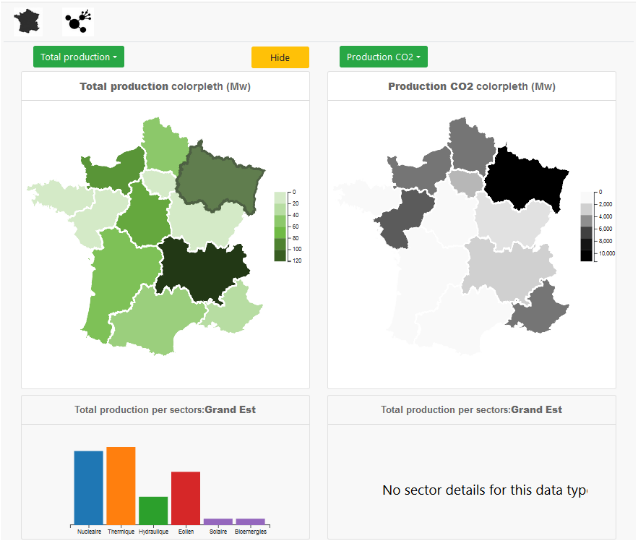


Figure: Example of map comparison for Region Grand Est

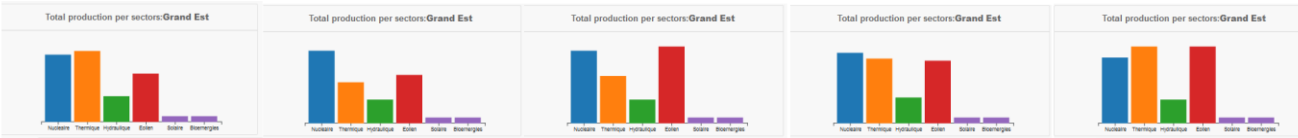


Figure: Absolute evolution for the different sources of energy for the Region Est

The histogram of region Grand Est show that between 2013 and 2014, the production of thermic energy decreased. From 2014, it seems that the same production of thermic energy starts to increase again, while the nuclear part is decreasing slightly. Or, it is the thermic energy production that is the main source of CO2 emission (with a factor of 70 compare to nuclear!), so small variation of the production here might have huge impact in term of CO2. If we go back to history, we can also understand these evolutions: in fact, the plant of Fessenheim suffered in that period of a decrease of its production and this has been compensated by an increase in the thermic production.

Same insights can be seen when “zooming” on PACA. It is mainly the increase of thermic production that is responsible for the increase of CO2 emissions.

Another interesting case is to check the average residence consumption per m², to assess who are the good and the bad examples to follow.

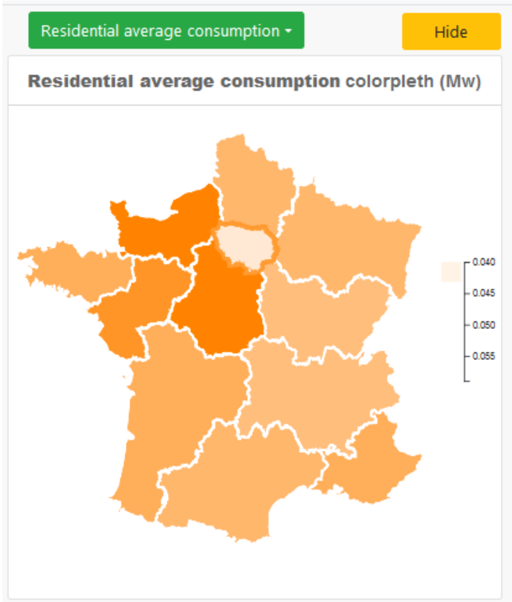


Figure: Residential average consumption

We can see from this map that it is in Ile de France that the residence is in average the more sober. This can be very well explained by the density of the area: in fact, a lot of the residential surface there are apartments, which implies less heat loss, so less need of energy.

What is interesting is to notice that for Val de Loire and Normandie, the average consumption for residency is almost 30% higher than in the rest of France. It could be interesting to dig this further by correlate this to the average age of the buildings, that would imply a less efficient isolation.

Finally, there is one last example of feature we would like to go through with you: the mapping of exceeding and deficit production per region, that we can correlate to the CO2 emissions, both for 2013 and 2017:

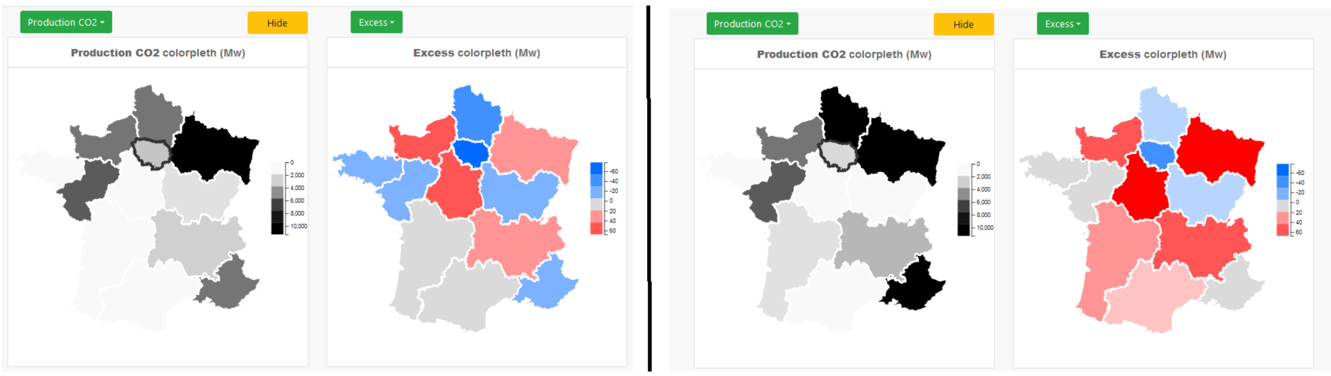


Figure: CO2 and Excess of Production, for 2013 (left) and 2017 (right)

Let's have a first look on the "Excess" map, showing in blue the area that are in energy deficit and in red the area in energy excess.

First observation is that region Ile de France is the region with the biggest deficit. This looks obvious has it is also the densest area in term of residence, and that there are no places for production plants. One part of the energy going there is coming from the border zones.

Also, we can notice that the number of regions in excess of energy increased between 2013 and 2017. There was 6 Regions in energy deficit in 2013, this number goes down to 3 in 2017, and in the same time, the number of areas in excess of production has increased too. This is good as in underline that more Regions of France gained their energetic independence other the past years.

Nevertheless, this good point must be counterbalanced by the trade-off in term of CO_2 emissions, where we can see a global increase for the regions that improve their power balance. Thus, we can question ourselves if it is relevant to become “energetically independent" if it is to use more polluting sources like biomasses or thermic energy (coil, gaz), and later export it the other countries.

Conclusion

In conclusion, we saw that despite a precise idea in mind, it can be tough to make a user-friendly application that can truly help people understand a subject. We think that all the interactions we included are nice for a user because it allows a ludic way of exploring the data.

We regret not having the time to push the subject forward to include more insights as we wanted originally. For instance, adding correlations with weather or building age, or even dig more in the news to understand one or another pattern. Also, even if this project is very good to have a first descriptive overview, it also cuts a lot of details. We could have refined to a tighter temporal or geographical mesh, as well as including energy transfers with border countries.

Also, the technical choices made for the histograms are a bit tricky to understand and a common user could be easily tricked as each column varies on its own scale. Without this design choice, we could not see the trend over time as nuclear would have been the major energy for every department.

To summarize, we think that our application can be a very good initiation for a person newly interested by the energy problems, and we hope that it will bring them the flash of curiosity to dig even further by themselves.