



***Spritify***

**What?**

**Why?**

**How?**

# Data Sources



[About us](#)[Merger control](#)[Ban on cartels](#)[Abuse control](#)[Economic sectors](#)[Public procurement](#)[Consumer Protection](#)

## Content

[Digital Economy](#)[Energy](#)[Food retail trade](#)[Health sector](#)[Media industry](#)[Mineral oil](#)[Market Transparency Unit for Fuels](#)[Waste management](#)[Telecommunications](#)[Homepage](#) ▶ [Economic sectors](#) ▶ [Mineral oil](#) ▶ [Market Transparency Unit for Fuels](#)

## Market Transparency Unit for Fuels

**The Market Transparency Unit for Fuels is intended to enable consumers to gain information on current fuel prices in Germany.**



(© Gina Sanders-Fotolia.com)

The price data collected by the unit are also meant to improve the Bundeskartellamt's possibilities to intervene in the case of illegal predatory strategies and other forms of market power abuse.

Since 31 August 2013 companies which operate public petrol stations or have the power to set their prices are obliged to report price changes for the most commonly used types of fuel, Super E5,

Super E10 and Diesel "in real time" to the Market Transparency Unit for Fuels. This then passes on the incoming price data to consumer information service providers, which in turn pass it on to the consumer. Via the internet, a smartphone or navigation system, motorists will be able to gain information on the current fuel prices and find the cheapest petrol station in their vicinity or along a specific route. This will allow for

## Contact

Market Transparency  
Unit for Fuels

Phone: +49 228

9499 525

Fax: +49 228 9499  
403

[write e-mail](#)

Azure DevOps

Search

Sign in

tankerkoenig-data

Overview

Boards

Repos

Files

Commits

Pushes

Branches

Tags

Pull requests

Pipelines

> dev.azure.com/tankerkoenig/\_git/tankerkoenig-data

<<

tankerkoenig / tankerkoenig-data / Repos / Files / tankerkoenig-data

master tankerkoenig-data / prices / 2019 / 05

Clone

tankerkoenig-data

prices

2014

2015

2016

2017

2018

2019

01

02

03

04

05

stations

README.md

Contents

History

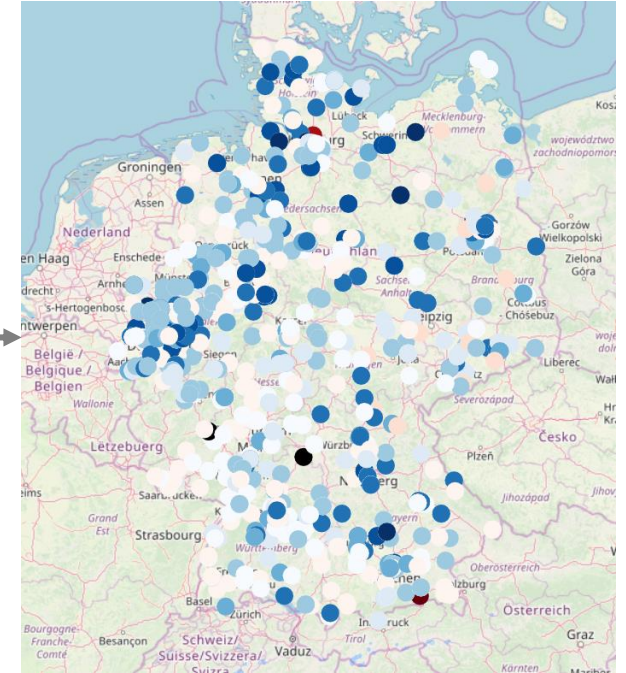
Download as Zip

Name ↑	Last change	Commits	
2019-05-01-prices.csv	02.05.2019	4d9af5db	add files for 2019-05-01 Hermann
2019-05-02-prices.csv	03.05.2019	b223b048	add files for 2019-05-02 Hermann
2019-05-03-prices.csv	04.05.2019	68dcbe46	add files for 2019-05-03 Hermann
2019-05-04-prices.csv	05.05.2019	f1a14d02	add files for 2019-05-04 Hermann
2019-05-05-prices.csv	06.05.2019	35afbe48	add files for 2019-05-05 Hermann
2019-05-06-prices.csv	07.05.2019	c4ae6b02	add files for 2019-05-06 Hermann
2019-05-07-prices.csv	08.05.2019	8d6c1dcf	add files for 2019-05-07 Hermann
2019-05-08-prices.csv	09.05.2019	08e06165	add files for 2019-05-08 Hermann
2019-05-09-prices.csv	10.05.2019	dbc8b5a2	add files for 2019-05-09 Hermann
2019-05-10-prices.csv	11.05.2019	2dafc0b3	add files for 2019-05-10 Hermann
2019-05-11-prices.csv	12.05.2019	261a1db3	add files for 2019-05-11 Hermann
2019-05-12-prices.csv	13.05.2019	3c8cd844	add files for 2019-05-12 Hermann
2019-05-13-prices.csv	14.05.2019	9045f14a	add files for 2019-05-13 Hermann
2019-05-14-prices.csv	15.05.2019	0a9a10b7	add files for 2019-05-14 Hermann
2019-05-15-prices.csv	16.05.2019	2b69a7a2	add files for 2019-05-15 Hermann
2019-05-16-prices.csv	17.05.2019	46d18a42	add files for 2019-05-16 Hermann
2019-05-17-prices.csv	18.05.2019	e458f684	add files for 2019-05-17 Hermann
2019-05-18-prices.csv	19.05.2019	66c05e8d	add files for 2019-05-18 Hermann
2019-05-19-prices.csv	20.05.2019	00000000	add files for 2019-05-19 Hermann

# Data Wrangling

```
pd.read_csv('2019-05-06-stations.csv')
```

	uid	name	street	house number
228	u1a128-231-070-640-000a0a96d63d	Bad Driburg, Pymmer Str. 48	Pymmer Str.	48
230	u1a128-232-070-640-000a0a96d63d	Witten, Bodenborn Str. 17	Bodenborn	50
232	u1a128-232-070-640-000a0a96d63d	Uckeritz, Tie-Hagemann Str. 17	Tie-Hagemann-Str.	17
234	u1a128-232-070-640-000a0a96d63d	Sand, Vette-Linden 11	Vette-Linden	11
236	u1a128-232-070-640-000a0a96d63d	Witten, Vette-Linden 11	Vette-Linden	11
238	u1a128-232-070-640-000a0a96d63d	Ottensmünde, Möller-Land 4	Möller-Land	4
240	u1a128-232-070-640-000a0a96d63d	Nordenbunde, Tengelinge Lander 18	Tengelinge Lander	18
248	u1a1281-2521-070-640-000a0a96d63d	Neuesbüdingen, Deymer Str. 80	Deymer Str.	80
241	u1a1281-2523-070-640-000a0a96d63d	Niemeln, Schauer Str. 5	Schauer Str.	5
242	u1a1281-2523-070-640-000a0a96d63d	Leutzing, Thelwe Str. 32a	Thelwe Str.	32a
243	u1a1281-251c-070-640-000a0a96d63d	Kötthen, Prossner Kaserne / B13	Prossner Kaserne / B13	Nah
244	u1a1281-251c-070-640-000a0a96d63d	Kell, Ratscherstr. 1	Ratscherstr. 1	1
247	u1a1281-251c-070-640-000a0a96d63d	Goslar, Chemn 12	Chemn 12	12
248	u1a1281-251c-070-640-000a0a96d63d	Freiburg, Frankfurter Str. 1	Frankfurter Str.	1
250	u1a1281-251c-070-640-000a0a96d63d	Witten, Kaserne 18	Kaserne 18	18
251	u1a1281-251c-070-640-000a0a96d63d	Dresden, Hermann Str. 7	Hermann Str.	7
252	u1a1281-252a-070-640-000a0a96d63d	Dortmund, Lindenhorst 165	Lindenhorst-Str.	165
254	u1a1281-252b-070-640-000a0a96d63d	Darmstadt, Fleisch-Baer Str. 73	Fleisch-Baer-Str.	73
255	u1a1281-252b-070-640-000a0a96d63d	Cottbus, Cösterlin-Str. 61	Cösterlin-Str.	61
256	u1a1281-252b-070-640-000a0a96d63d	Chemnitz, Dresden Str. 119	Dresden Str.	119
257	u1a1281-252b-070-640-000a0a96d63d	Burg, Grodenwald 68	Grodenwald	68
258	u1a1281-2527-070-640-000a0a96d63d	Bremenhaven, Haltern 23-20	Haltern	23-20
260	u1a1281-252a-070-640-000a0a96d63d	Bielefeld, Berliner Str. 27a	Berliner Str.	27a
262	u1a1281-252c-070-640-000a0a96d63d	Querenhorst, Helmreichs Str. 8	Helmreichs Str.	8
263	u1a1281-252c-070-640-000a0a96d63d	Witten, Kaserne 18	Kaserne 18	18
264	u1a1281-252c-070-640-000a0a96d63d	Langerhagen, Am Englischen 11	Am Englischen 11	11
265	u1a1281-251b-070-640-000a0a96d63d	Grodenwald, Magdeburger Str. 21	Magdeburger Str.	21
268	u1a1281-251b-070-640-000a0a96d63d	Burgscheid, Caler Str. 13	Caler Str.	13
269	u1a1281-251b-070-640-000a0a96d63d	AMN, an der Bundesstr. 19	An der Bundesstr.	19
270	u1a1281-251b-070-640-000a0a96d63d	Wittenberg, Herberstr. 18b	Herberstr.	18b

[illegible]

```
pd.merge()  
def cleanup(...):
```

## TimestampedGeoJson()

```
folium.Map()
```

```
pd.read_csv(r'{0}-{1}-{2}-prices.csv'.format(year, month, day))
```

**Demo time!**

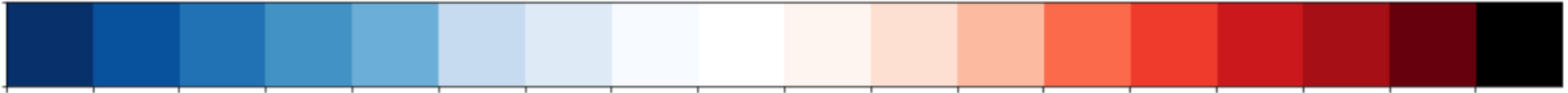


**libraries / tools used**

```
pd.cut(pumps['e5'], bins = 15, labels = ['#08306b', ...])
```



```
sns.palplot(sns.color_palette(color_scale))
```



```
from folium.plugins import TimestampedGeoJson
```

```
def create_geojson_features(df):
```

## Visualizing Air Pollution with Folium Maps

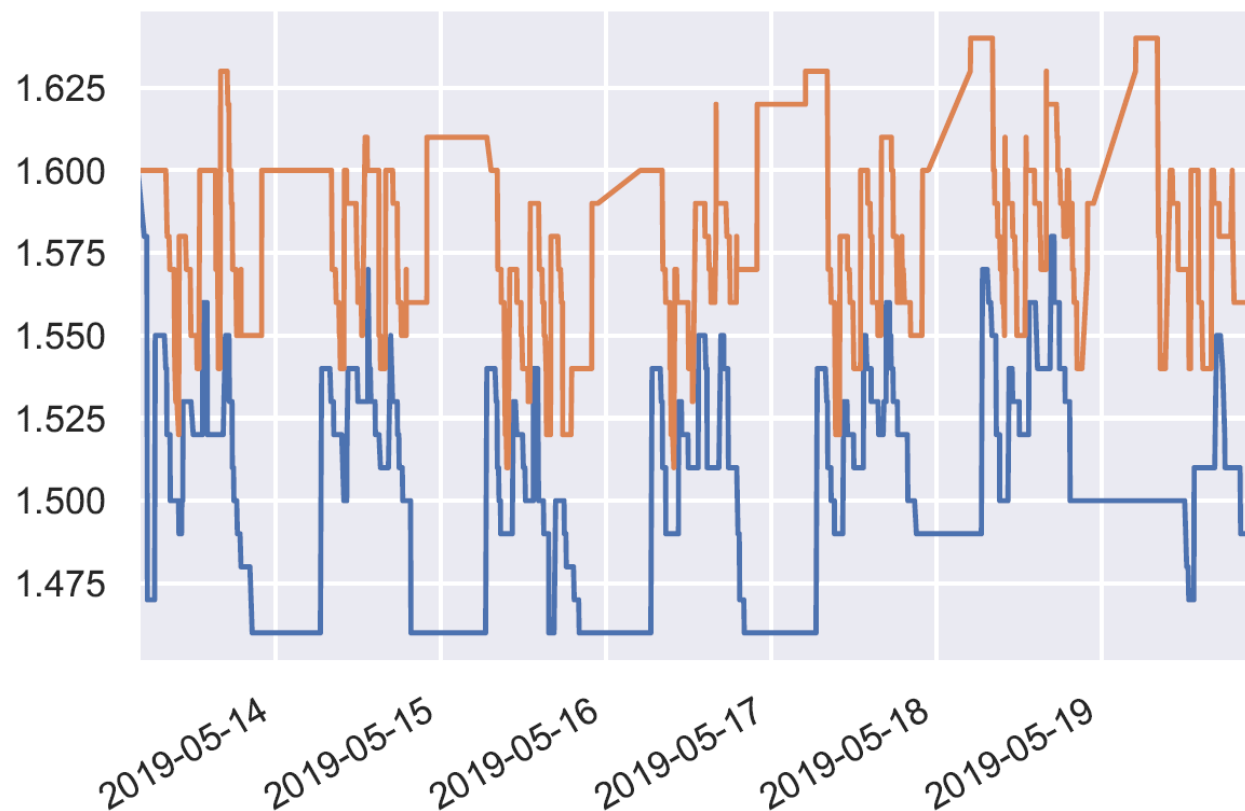


Bert Carremans

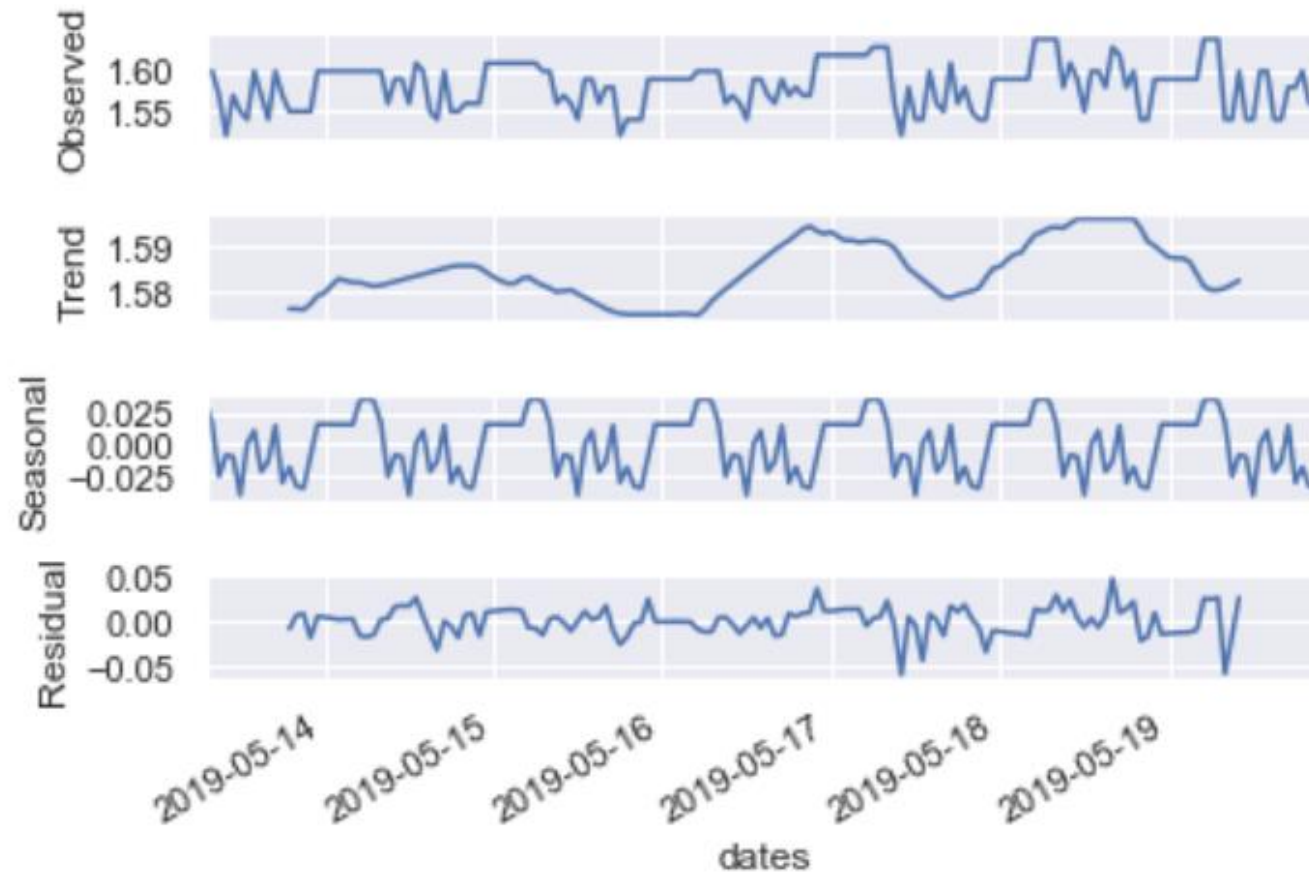
Follow

Nov 26, 2018 · 6 min read

# insights from project



# Next steps predict gas prices (RNNs, SARIMA?)



# insights from project

- By converting data to the GeoJSON format, timeseries data, with spatial information can be easily displayed with folium for exploratory data analysis
- Don't try to reinvent the wheel – someone else on the internet probably had a similar question
- Don't start to optimize too early!



# Thanks!

Now: 10 min break

# GeoJSON features

```
[340]: features = create_geojson_features(pumps_short)
       features[1]
```

```
> Creating GeoJSON features...
> Done!
```

```
[340]: {'type': 'Feature',
       'geometry': {'type': 'Point', 'coordinates': [13.3135, 52.4804]},
       'properties': {'time': '2019-05-09 08:22:06',
                      'style': {'color': '#6baed6'},
                      'icon': 'circle',
                      'iconstyle': {'fillColor': '#6baed6',
                                     'fillOpacity': 0.8,
                                     'stroke': 'true',
                                     'radius': 7}}}
```

# Top brands (> 400 stations)

[369]: <matplotlib.axes.\_subplots.AxesSubplot at 0x1d8c4641a58>

