

same pdftitle=Electricity Prices & Renewable Generation, pdfauthor=Ebba
Mark, hidelinks, pdfcreator=LaTeX via pandoc

Electricity Prices & Renewable Generation

Ebba Mark

2025-05-29

Overview

Motivation: Recall our motivation originally stemming from an interest in the “relative” volatility of fossil fuel versus renewable prices. After repeated discussion regarding the comparability of the two types of energy sources and scoping of the data available we pivoted to questions more directly related to “explaining”/investigating the recent spike in electricity prices (both in level and volatility) following a surge in natural gas prices and the Russian invasion of Ukraine and how it might have been mediated (or not) by increased renewable energy generation.

RQs: Though still to be workshopped, potential guiding questions along these lines (most follow from our past conversations):

1. To what extent, if any, does increased reliance on renewable sources of electricity insulate consumers or industry from the effects of spikes in electricity price inputs (ie. natural gas spike post-Ukraine)?
2. What are the costs (to electricity consumers - industrial and/or household) of recent increased price volatility and to what extent is greater renewable energy as a share of generating capacity a mediating or aggravating force?
3. Setting the price is an economic question - how efficiently does the market translate the true cost into price signals (ie. shifting share to lower cost renewables not yet filtering into price signals due to marginal price-setting in the market)?
4. What is the potential efficiency gain of switching to renewables even if it is not yet translated into prices?
5. Exogenous: use the connection of large wind or solar connections to the grid as plausible exogenous variation. See if there have been changes in capacity and whether they had any detectable effect on prices, all else equal.

Data: The below documents the data currently pulled to attempt a preliminary mean model of hourly wholesale electricity prices in Germany: * *Wholesale Electricity Prices* (Hourly - EMBER)

- * *Electricity Demand/Load* (Hourly - ENTSOE)
- * *Electricity Generated by Energy Source* (Hourly - ENTSOE)
- * *Various Commodity Price Indices* (Monthly - IMF)
- * *Total and Country-Specific Net imports* (Hourly - ENTSOE)
- * *Day-ahead Solar and Wind Forecasts* (Hourly - ENTSOE)
- * *Redispatching Costs* (“Hourly” - netztransparenz.de - they are reported at a particular hour but are contracted for “implementation” over a certain multi-hour period, I think...as mentioned, to be investigate more)

Data that remains to be integrated:

- * *Auction Bid Data* (Data on electricity auction bids - this would likely 1. help us determine if renewables set the market price (if ever) and 2. the variability in generation costs of fossil fuel providers)
- * *Economic Activity* (GDP or VAT)

Data Issues: Several data issues cropped up in the process. I note the most glaring issues in **bold** throughout this notebook. Notably, the main issues are:

- * the lower-frequency of certain candidate explanatory variables (ie. fuel and non-fuel commodity prices and

redispatching costs)

* missing data on net imports from France and Denmark in specific years (data exists but is not complete)

* how to aggregate the data on redispatching costs which are reported with hourly specificity but with different summarising metrics and “implementation horizons” (ie a load-balancing request made with a 4-hour implementation horizon). Fortunately, it does not seem to be an issue of data availability but rather deeper thinking about what measure (and harmonisation across energy types) is most relevant for our ultimate research question.

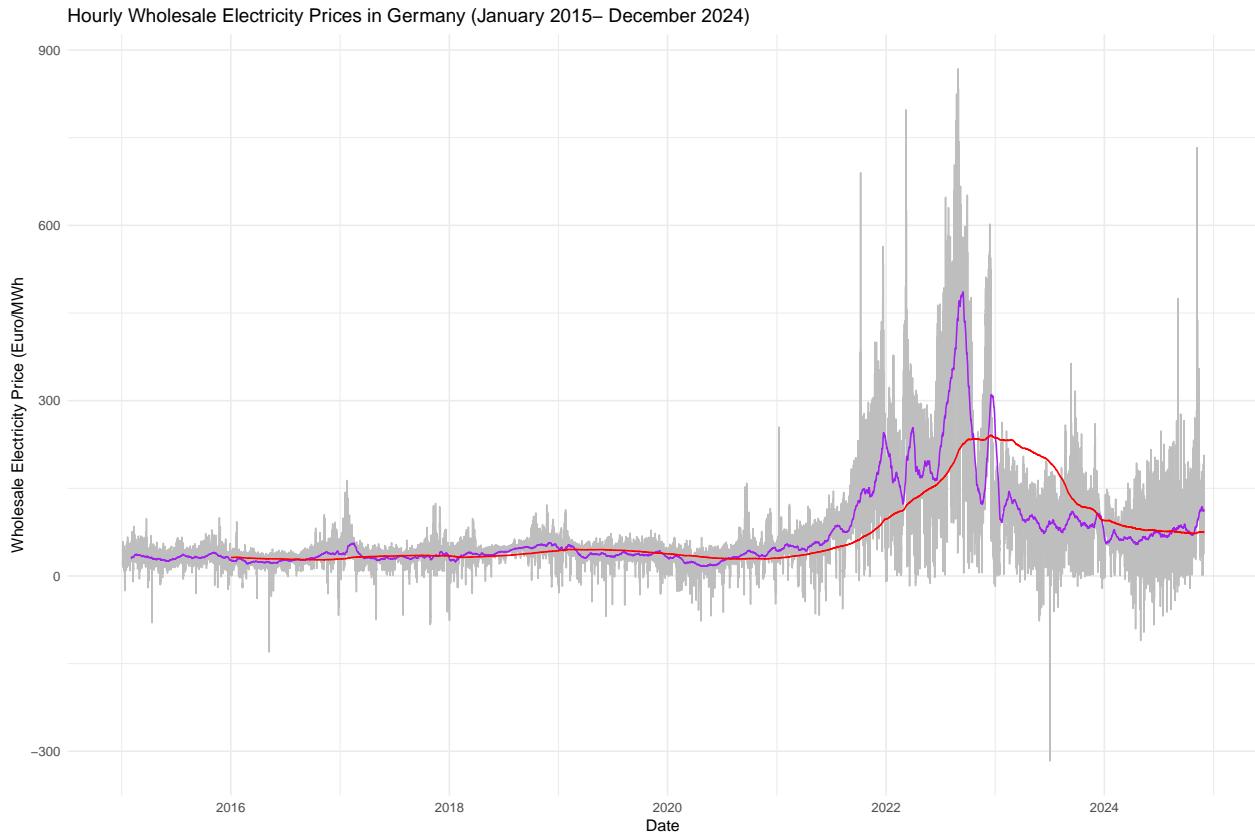
Model Attempt: Preliminary model attempt bringing this data together using gets under the “approaching a model...” tab.

Additional thoughts: Though, ultimately, it would be very interesting to broaden to a multi-country scope (especially if this might allow for some comparative analysis on variable country-specific shares of renewable electricity generation), we decided to focus on Germany as a first case to see how we go. If/when we take the step to further countries, any data that I pull from ENTSO-E has an accompanying scraping script that could (reasonably) easily be repurposed to pull the identical data on other countries.

Start simple: Germany Only

Data

Using data from <https://ember-energy.org/data/european-wholesale-electricity-price-data/EMBER> on European wholesale electricity prices (hourly - daily and monthly also exist). The hourly prices are the day-ahead wholesale prices in Germany.

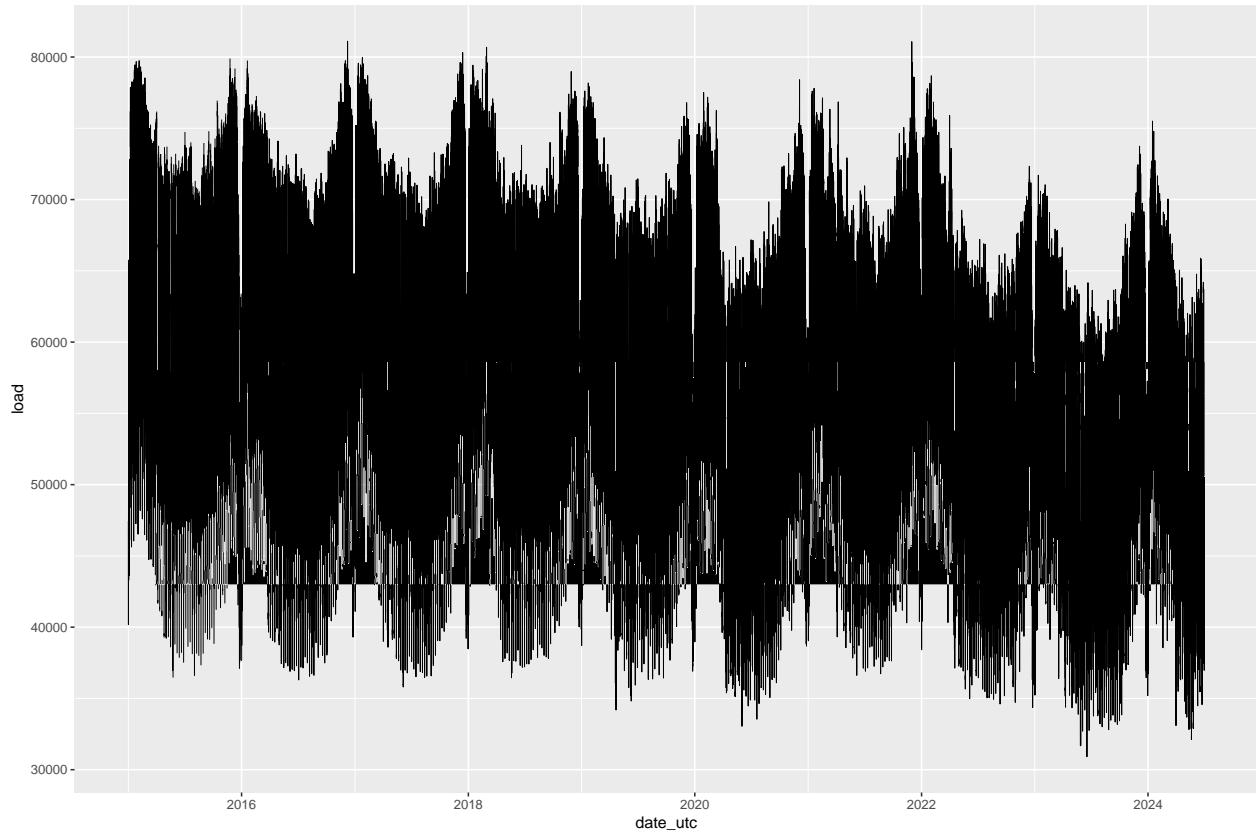


Electricity Demand/Load Total load per bidding zone per market time unit - in our case, total load, per country, per hour in MW.

https://transparency.entsoe.eu/content/static_content/Static%20content/knowledge%20base/data-views/load-domain/Data-view%20Total%20Load%20-%20Day%20Ahead%20-%20Actual.html

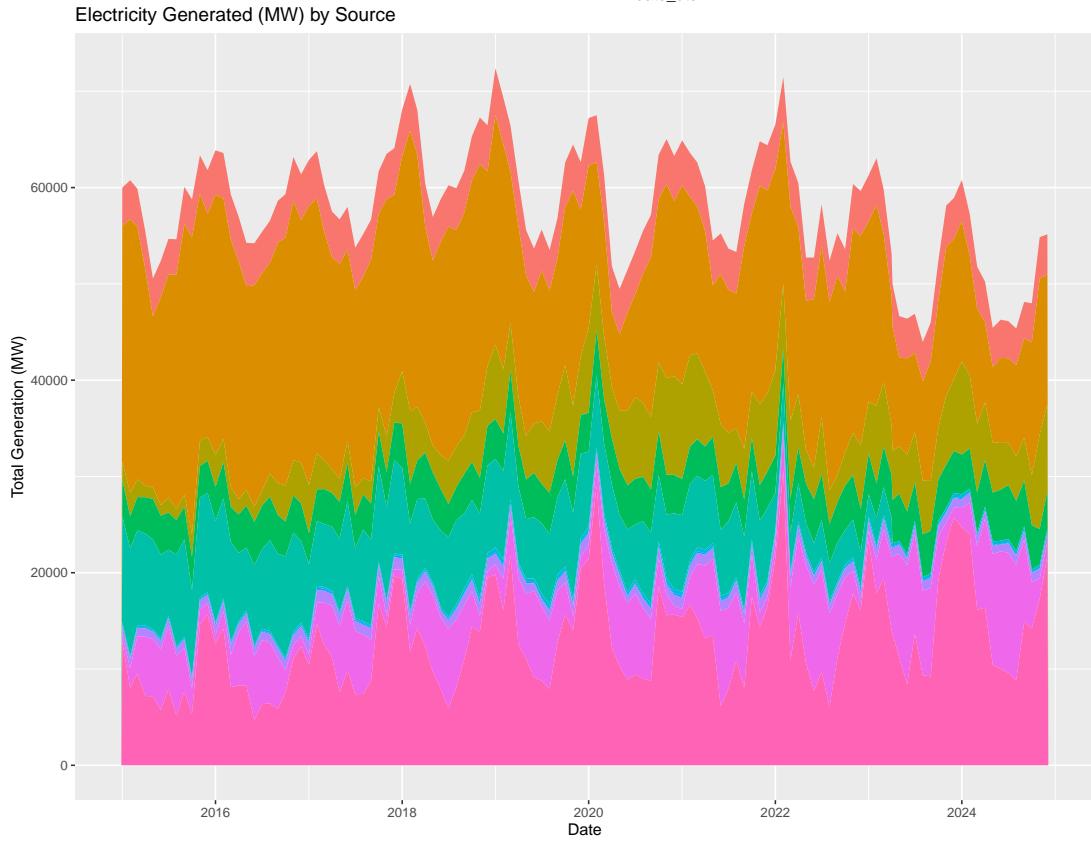
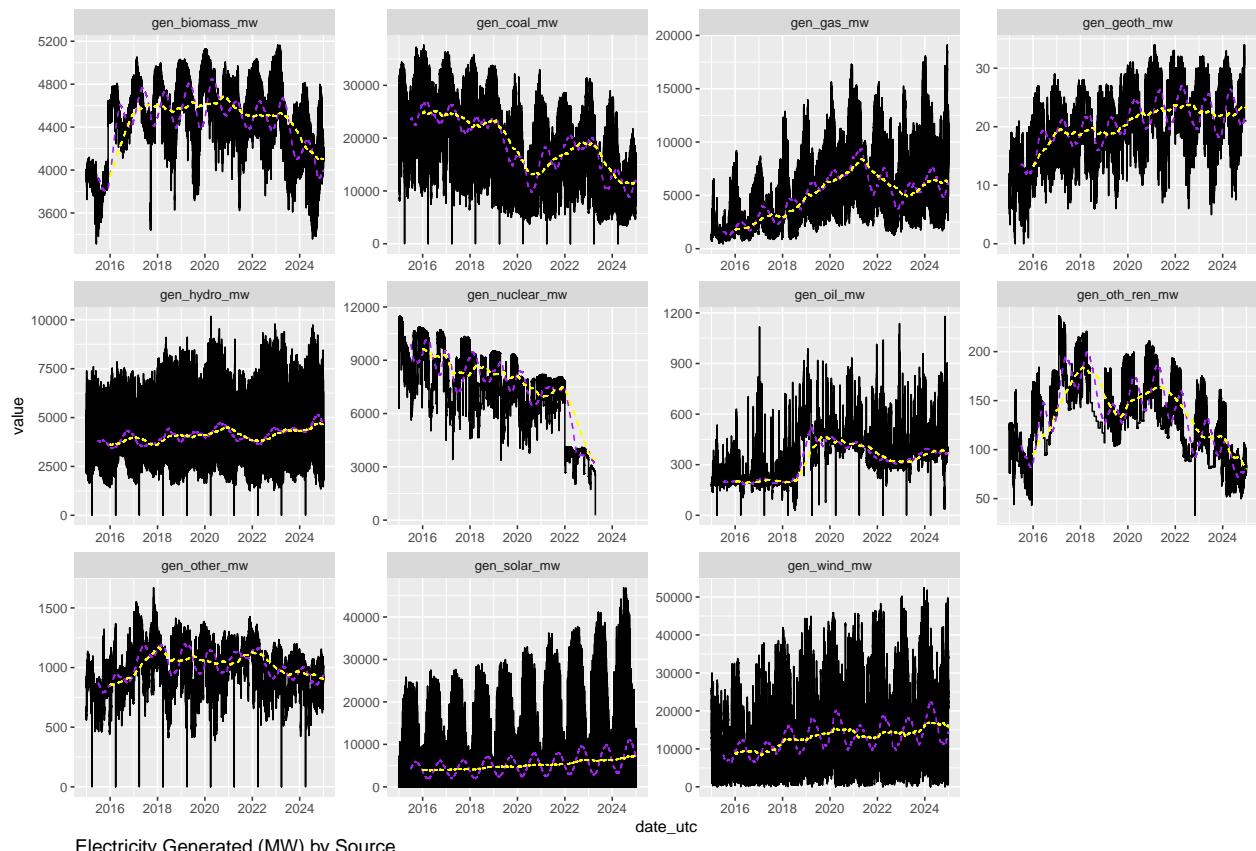
The following is data from <https://www.entsoe.eu/data/power-stats/ENTSO-E> on hourly electricity loads.

Components: Actual total load (including losses without stored energy) = net generation – exports + imports – absorbed energy



Generation by Resource Actual Generation per Production Type
from https://transparency.entsoe.eu/content/static_content/Static%20content/knowledge%20base/data-views/generation/Data-view%20Actual%20Generation%20per%20Production%20Unit.html ENTSOETHis value represents

Note: There is 1 observation missing per year (either 0 or NA) on a late day in march at 2 am...a cleaning issue? Something silly?



Fuel and Commodity Prices IMF provides *monthly* commodity prices and price indices.

Source: <https://www.imf.org/en/Research/commodity-prices>

The commodity prices listed below are:

Note: These are MONTHLY values. I fill these values to hourly when merging the data later on.

I include the following three commodity prices in the model selection for now (final plot in this section):

* Natural Gas, Netherlands TTF Natural Gas Forward Day Ahead, US\$ per Million Metric British Thermal Unit

* Fuel & Non-Fuel Commodities: All Commodity Price Index, 2016 = 100, includes both Fuel and Non-Fuel Price Indices

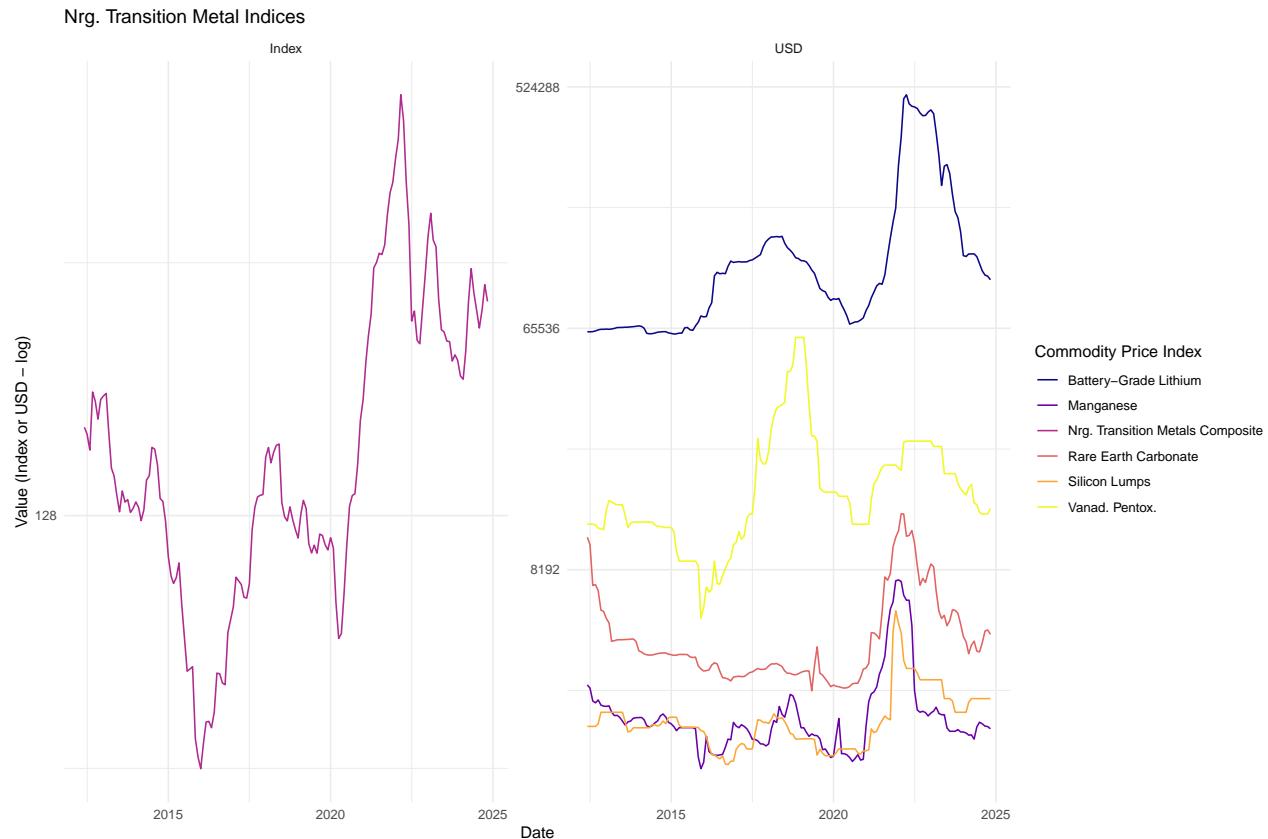
* Indust. Input Price: Industrial Inputs Price Index, 2016 = 100, includes Agricultural Raw Materials and Base Metals Price Indices

Note: We should probably rethink the non-natural gas prices (the latter two indices). The two latter indices are global indices and given the industrial structure of Germany we might want to consider a Producer Price Index for Germany specifically, assuming that exists? Additionally, as you can see in the graph, the latter two essentially trace each other/collinearity issues.

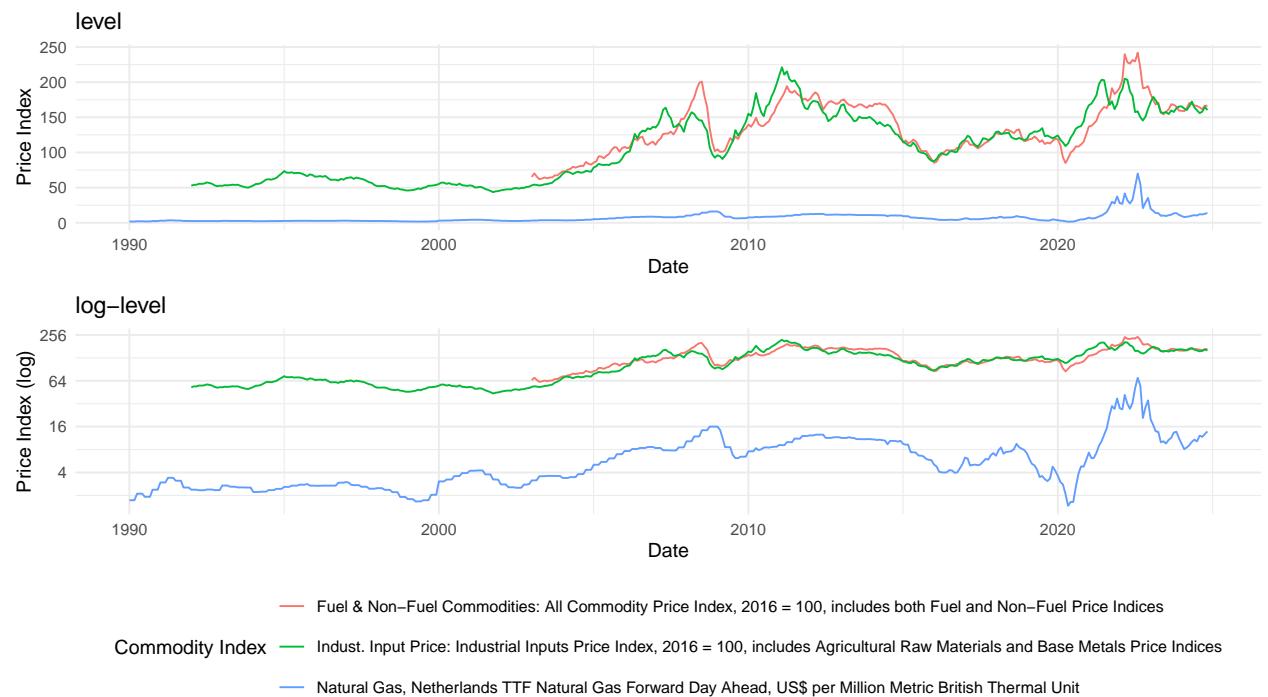
Commodity Type	Index Label (in plots)	Index Definition	Index Unit (USD or Index)
Coal	Coal	Coal Price Index, 2016 = 100, includes Australian and South African Coal	Index
Gas	Nat. Gas (NDL TTF)	Natural Gas, Netherlands TTF Natural Gas Forward Day Ahead, US\$ per Million Metric British Thermal Unit	USD
Oil	Oil - Avg Brent, West TX, Dubai (USD)	Crude Oil (petroleum), Price index, 2016 = 100, simple average of three spot prices; Dated Brent, West Texas Intermediate, and the Dubai Fateh	USD
Oil	Oil - Brent	Crude Oil (petroleum), Dated Brent, light blend 38 API, fob U.K., US\$ per barrel	USD
Oil	Oil - Dubai	Crude Oil (petroleum), Dubai Fateh Fateh 32 API, US\$ per barrel	USD
Oil	Oil - West TX	Crude Oil (petroleum), West Texas Intermediate 40 API, Midland Texas, US\$ per barrel	USD
Oil	Oil - Avg Brent, West TX, Dubai	Crude Oil (petroleum), Price index, 2016 = 100, simple average of three spot prices; Dated Brent, West Texas Intermediate, and the Dubai Fateh	Index
Gas	Nat. Gas Henry Hub	Natural Gas, Natural Gas spot price at the Henry Hub terminal in Louisiana, US\$ per Million Metric British Thermal Unit	USD
Composite Commodity Indices	Indust. Input Price	Industrial Inputs Price Index, 2016 = 100, includes Agricultural Raw Materials and Base Metals Price Indices	Index
Composite Commodity Indices	Fuel (Energy - oil, ng, coal, propane)	Fuel (Energy) Index, 2016 = 100, includes Crude oil (petroleum), Natural Gas, Coal Price and Propane Indices	Index
Gas	Nat. Gas (Euro, Jap, American)	Natural Gas Price Index, 2016 = 100, includes European, Japanese, and American Natural Gas Price Indices	Index

Commodity Type	Index Label (in plots)	Index Definition	Index Unit (USD or Index)
Gas	Nat. Gas (Indonesian LNG in Japan)	Natural Gas, Indonesian Liquefied Natural Gas in Japan, US\$ per Million Metric British Thermal Unit	USD
Composite Commodity Indices	Fuel & Non-Fuel Commodities	All Commodity Price Index, 2016 = 100, includes both Fuel and Non-Fuel Price Indices	Index
Nrg. Transition Input Material Prices	Nrg. Transition Metals Composite	Energy Transition Metal Index	Index
Nrg. Transition Input Material Prices	Battery-Grade Lithium	Lithium Metal =99%, Battery Grade	USD
Nrg. Transition Input Material Prices	Manganese	Manganese Electro CIF NWE (US\$ perMT)	USD
Nrg. Transition Input Material Prices	Rare Earth Carbonate	Rare Earth Carbonate REO 42-45 Dom.	USD
Nrg. Transition Input Material Prices	Silicon Lumps	Silicon Lumps CIF NWE (US\$ perMT)	USD
Nrg. Transition Input Material Prices	Vanad. Pentox.	Vanadium Pentoxide CIF NWE	USD





Preliminary Selected Commodity Prices for Model Selection

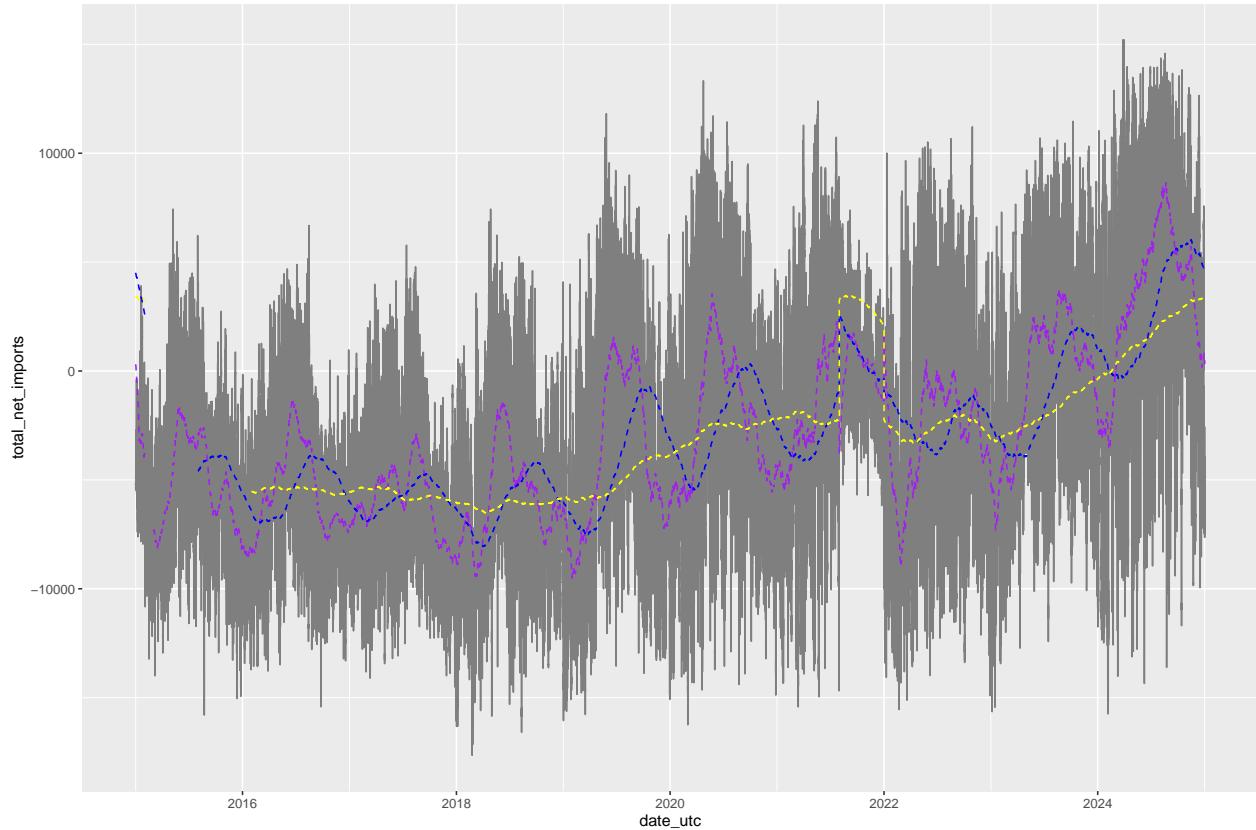


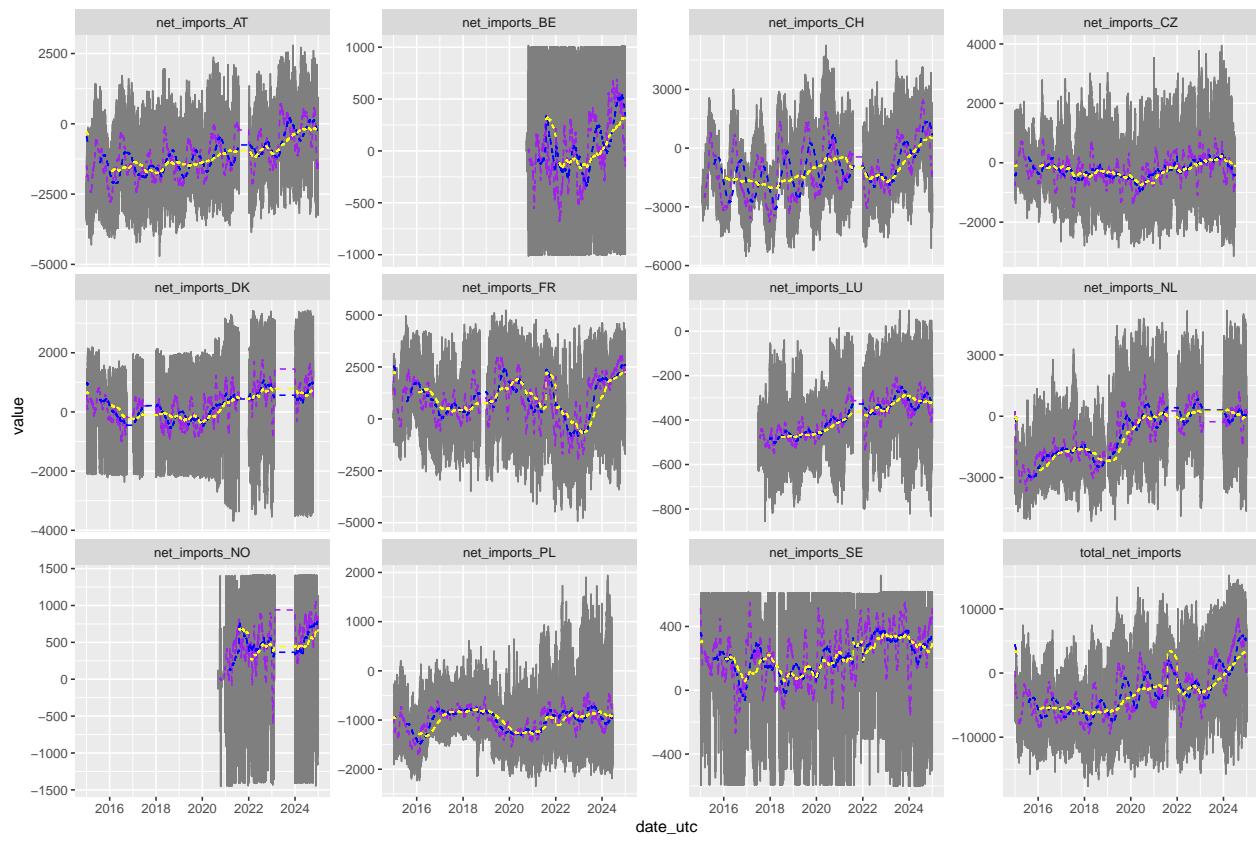
Net Imports Imports and export flows of electricity from https://transparency.entsoe.eu/content/static_content/views/transmission-domain/Data-view%20Cross%20Border%20Physical%20Flows.html ENTSO-E Transparency Platform. Physical flows between bidding zones per market time unit.

Physical flow is defined as the measured real flow of energy between neighbouring bidding zones on the cross borders.

Note 1: Net imports still has some missing data and I'm not quite sure where it comes from (see Austria, Denmark, France, Switzerland).

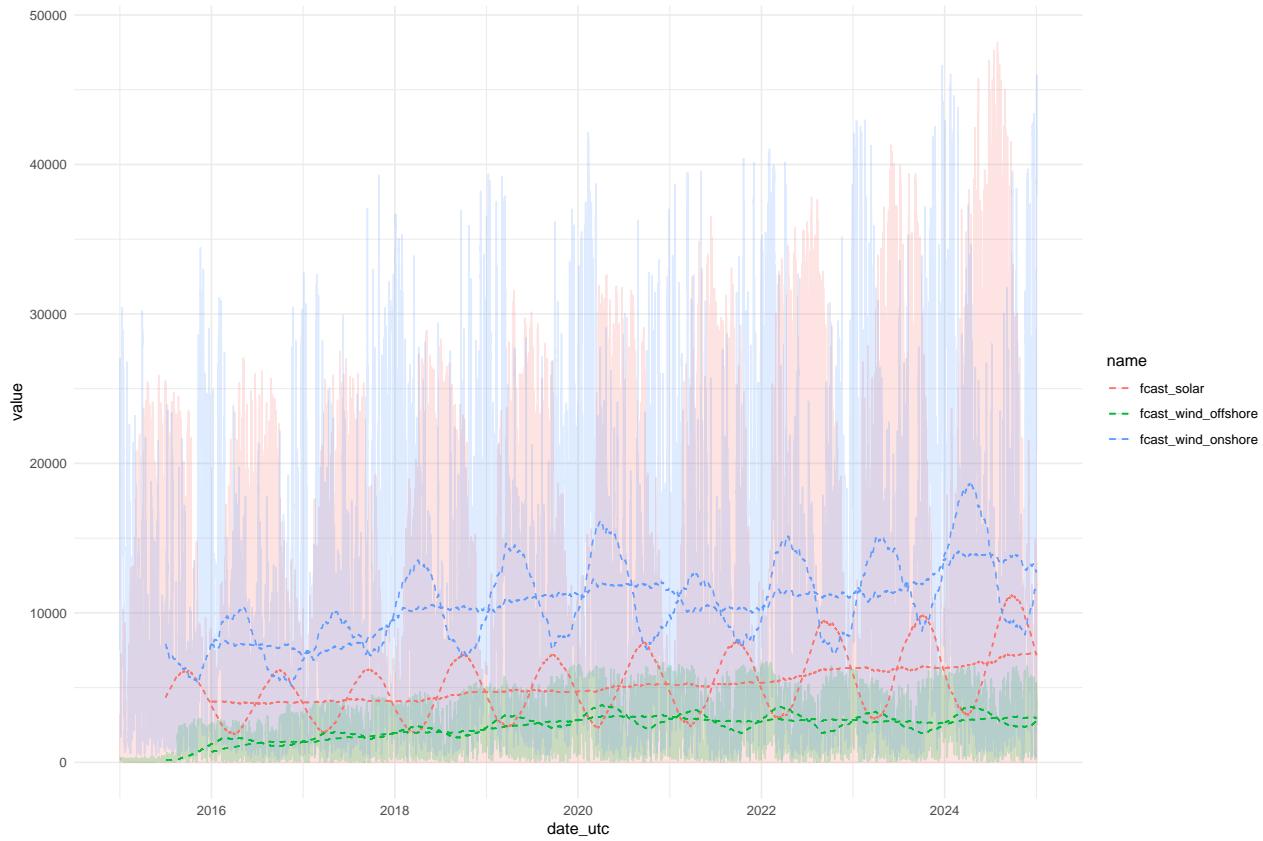
Note 2: Also, the seeming upper and lower limits must have to do with capacity constraints on imports and exports (see Belgium, Norway, Sweden).





Day-ahead Solar and Wind Forecasts A forecast of wind and solar power generation (MW) per country per hour of the following day data from https://transparency.entsoe.eu/content/static_content/Static%20content/knowledge%20base/data-views/generation/Data-view%20Generation%20Forecasts%20-%20Day%20Ahead%20for%20Wind%20and%20Solar.html

Note: we need to consider the logical link between these forecasts and price-setting/auction behaviour.



Redispatching Costs

Four transmission system operators (TSOs) in Germany actively monitor the contracted transmission of electricity in the grid to ensure load-balancing “in the event of imminent overloads in the electricity grid.”

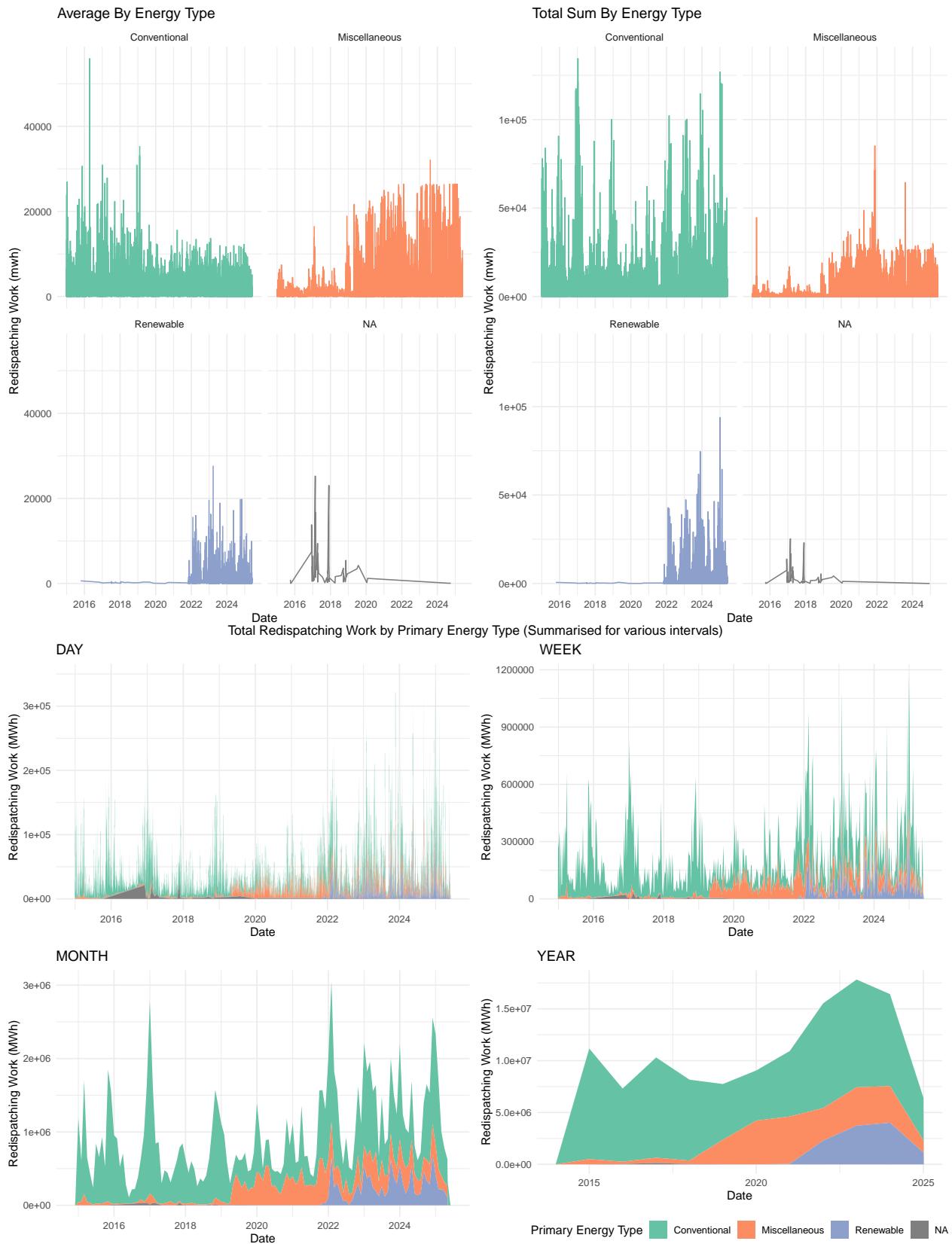
“Redispatch is a request to adjust the active power feed-in of plants (conventional power plants, CHP plants, renewable energy plants, storage facilities) by the grid operator with the aim of avoiding or eliminating any bottlenecks that occur. This measure can be applied within and across control areas. By reducing the active power feed-in of one or more systems while simultaneously increasing the active power feed-in of one or more other systems, the total active power feed-in remains almost unchanged while at the same time relieving a bottleneck” (<https://www.netztransparenz.de/en/Ancillary-Services/System-operations/Redispatch%5DNETZTRANSPARENZ.DE>).

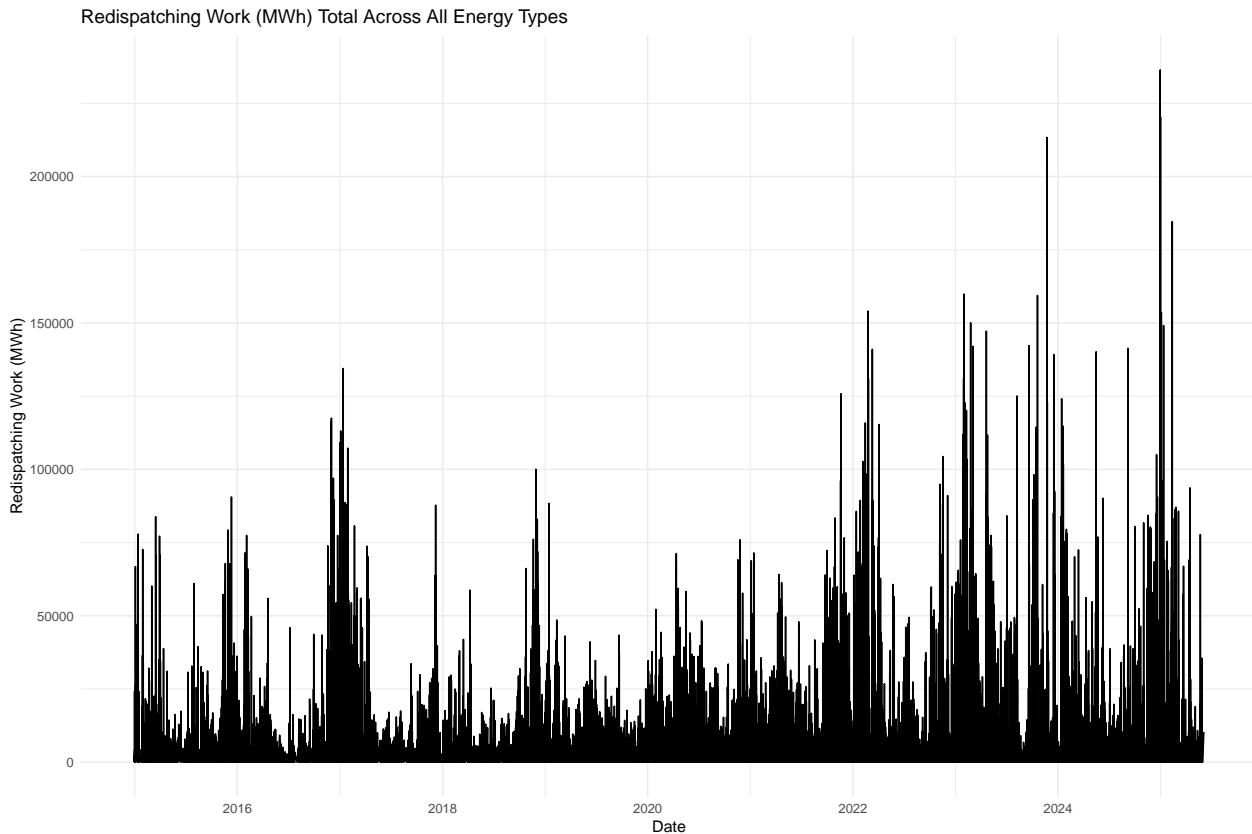
(<https://www.netztransparenz.de/en/Ancillary-Services/System-operations/Redispatch%5DNETZTRANSPARENZ.DE>) provide data on redispatch measures for the average, maximum, and per-quarter-hour(??) MW reduction. I think the latter is most relevant. See example at the link above.

The plots below show the associated “redispatching work” by primary energy type (conventional, renewable, miscellaneous, and unspecified).

Note: This information is also available from <https://transparency.entsoe.eu/congestion-management/r2/redispatching-internal/showENTSO-E>. We might consider using this data source instead for consistency.

Note: to summarise by time period, I use either the mean or sum total of the measure of “redispatching work in mwh.” I am still not 100% sure what this means and whether this measure is most appropriate (or best summarised per hour as a sum or mean). We MUST revisit this data before implementation - NON-NEGOTIABLE.





Combining Data For transparency's sake, the code is below:

1. merge price, generation data by date-time and country-code
2. upper limit of TS to end of 2024
3. log transform with $x + 1$ to avoid zero value issues
4. create "share" values by dividing any generation variable by total load
5. for monthly commodity price values I fill these variables *forward*
6. Create hour and month dummies

Note: Potential transformation issues to discuss:

1. Forward-fill of monthly commodity price values
2. To deal with 0 values for certain variables I need to log_transform the value + 1. I don't think this is a major issue given the magnitude of these variables but worth reconsidering.
3. (!!) For net imports there are several observations for country-specific flows that are NA. Most result from NA values reported for both imports and exports at those hours. However, for Denmark (1,860 obs) and France (2,234 obs) this issue results from non-NA imports and NA exports. **THIS WILL ABSOLUTELY NEED TO BE REINVESTIGATED - WE CANNOT FORMALISE ANY RESULTS WITHOUT FINDING A WAY TO DEAL WITH THIS.**

```
de_all <- de_hourly_prices %>%
  rename(date_utc = datetime_utc) %>%
  left_join(., entsoe_de, by = c("date_utc", "country")) %>%
  left_join(., de_generation, by = c("date_utc", "country", "country_code")) %>%
  arrange(date_utc) %>%
  select(-starts_with("sum_")) %>%
  filter(date_utc < "2024-03-31 02:00:00" & !is.na(gen_biomass_mw)) %>%
  mutate(across(contains("gen"), ~log(. + 1), .names = "log_{.col}"),
        across(starts_with("gen"), ~./load, .names = "shr_{.col}"),
```

```

    log_load = log(load+1) %>%
left_join(., imp_exp, by = "date_utc") %>%
mutate(across(contains("net_imports"), ~ifelse(is.na(.), 0, .)),
       shr_net_imports = total_net_imports/load) %>%
left_join(., fcast_sw, by = "date_utc") %>%
left_join(., redisp_df, by = "date_utc")

de_full <- de_all %>%
left_join(., ng_price, by = "date_utc") %>%
# We fill the monthly values down!
fill(PNGASEU, .direction = "down") %>%
fill(PALLFNF, .direction = "down") %>%
fill(PINDU, .direction = "down") %>%
# Given our redispatching data covers the 2015-2024 time period, it is safe to assume that NA values
mutate(redisp_work_mwh = ifelse(is.na(redisp_work_mwh), 0, redisp_work_mwh),
      # Create hourly and monthly dummies
      hour = as.factor(hour(datetime_local)),
      month = as.factor(month(datetime_local)),
      log_redisp_mwh = log(redisp_work_mwh + 1),
      log_PNGASEU = log(PNGASEU + 1),
      log_PALLFNF = log(PALLFNF + 1),
      log_PINDU = log(PINDU + 1)) %>%
dummy_cols(select_columns = c("hour", "month"), remove_first_dummy = TRUE, remove_selected_columns =

```

(TBA) Auction Bid Data

(TBA) Economic Activity

Approaching a model... Approaching a model...

Below, I incorporate most explanatory variables (except solar and wind forecasts) above into models of the mean (not yet variance). The first model tab incorporates the variables simply as (log) levels whereas the second incorporates them as shares of total “load.” In each tab, I run the model with the same set of regressors to be selected over but with or without a log.ewma argument. In line with Rintamaki et al, we incorporate moving average terms for half-day, daily, and weekly price volatility (levels?) via log.ewma argument (correct?). I need to think more closely about how that functions together with the hourly and monthly dummies.

Note: log.ewma() for half-day, daily, and weekly price volatilities? Correct idea?

Note: Hourly and daily dummies the best way to incorporate “seasonality”? How might this interfere with the ewma terms?

Note: I do not currently incorporate any ARCH terms.

Note: day-ahead versus current prices in all variables? - are we not interested in deviations from forecasts for price-setting?

Note: Finally, a basic question, but I do not currently have the output variable (hourly wholesale electricity prices) in logs as it frequently takes negative values)

```
# Check to make sure data is ordered
de_full %>% arrange(date_utc) %>% all.equal(de_full)
```

```
## [1] TRUE
```

```

reg_mat <- as.matrix(select(de_full,
  'log_load',
  #'log_gen_biomass_mw',
  'log_gen_gas_mw',
  'log_gen_coal_mw',
  'log_gen_hydro_mw',
  'log_gen_geoth_mw',
  'log_gen_nuclear_mw',
  'log_gen_oth_ren_mw',
  'log_gen_solar_mw',
  'log_gen_wind_mw',
  'log_PNGASEU',
  'log_PALLFNF',
  'log_PINDU',
  contains('hour'),
  contains('month'),
  "net_imports_SE",
  "net_imports_PL",
  "net_imports_CZ",
  "net_imports_BE",
  "net_imports_CH",
  "net_imports_NO",
  "net_imports_NL",
  "net_imports_LU",
  "net_imports_FR",
  "net_imports_DK",
  "net_imports_AT",
  "log_redisp_mwh"
  ##"total_net_imports" # Excluded as its sum components are each o
)) # 'gen_wind_mw' 'gen_coal_mw', 'gen_hydro_mw',

de_mod_levs <- arx(de_full$price_eur_m_whe,
  # Question: In theory, the ARCH(24) would proxy the seasonal (daily) AR term?
  #arch = 1:24,
  mxreg = reg_mat)

de_mod_levs_logewma <- arx(de_full$price_eur_m_whe,
  # As in Rintamaki et al. the below line incorporates moving average terms for the half-day
  log.ewma = c(12, 24, 168),
  # Question: In theory, the ARCH(24) would proxy the seasonal (daily) AR term?
  #arch = 1:24,
  mxreg = reg_mat)

# model selection and plotting of residuals
de_levs_sel <- getsm(de_mod_levs, ar.LjungB = NULL, arch.LjungB = NULL)

```

Levels as Exp. Vars.

```

##
## GUM mean equation:
##
##           reg.no.  keep      coef    std.error     t-stat   p-value

```

## mconst	1	0	3.2813e+02	2.0264e+01	16.1926	< 2.2e-16	***
## log_load	2	0	1.4264e+01	1.7982e+00	7.9327	2.174e-15	***
## log_gen_gas_mw	3	0	2.9921e+01	4.1248e-01	72.5390	< 2.2e-16	***
## log_gen_coal_mw	4	0	-1.0527e+01	6.9903e-01	-15.0591	< 2.2e-16	***
## log_gen_hydro_mw	5	0	-4.2343e+00	4.9264e-01	-8.5953	< 2.2e-16	***
## log_gen_geoth_mw	6	0	1.4893e+01	7.3486e-01	20.2670	< 2.2e-16	***
## log_gen_nuclear_mw	7	0	-2.3938e+01	6.8937e-01	-34.7246	< 2.2e-16	***
## log_gen_oth_ren_mw	8	0	1.4739e+01	7.9203e-01	18.6093	< 2.2e-16	***
## log_gen_solar_mw	9	0	4.9121e-01	8.9646e-02	5.4795	4.280e-08	***
## log_gen_wind_mw	10	0	6.2983e-01	2.5612e-01	2.4591	0.0139312	*
## log_PNGASEU	11	0	8.5731e+01	9.4949e-01	90.2918	< 2.2e-16	***
## log_PALLFNF	12	0	9.4051e+01	3.5942e+00	26.1672	< 2.2e-16	***
## log_PINDU	13	0	-2.2499e+02	2.1116e+00	-106.5507	< 2.2e-16	***
## hour_1	14	0	-1.7632e+00	1.0281e+00	-1.7149	0.0863622	.
## hour_2	15	0	-5.8969e-01	1.0359e+00	-0.5693	0.5691754	
## hour_3	16	0	-9.6671e-01	1.0444e+00	-0.9256	0.3546674	
## hour_4	17	0	3.1438e-01	1.0517e+00	0.2989	0.7649997	
## hour_5	18	0	3.1168e+00	1.0512e+00	2.9651	0.0030273	**
## hour_6	19	0	9.6354e+00	1.0577e+00	9.1100	< 2.2e-16	***
## hour_7	20	0	1.4310e+01	1.0842e+00	13.1983	< 2.2e-16	***
## hour_8	21	0	1.1163e+01	1.1569e+00	9.6494	< 2.2e-16	***
## hour_9	22	0	4.8684e+00	1.2666e+00	3.8435	0.0001214	***
## hour_10	23	0	9.3291e-02	1.3436e+00	0.0694	0.9446430	
## hour_11	24	0	-2.4434e+00	1.3952e+00	-1.7513	0.0798985	.
## hour_12	25	0	-6.0279e+00	1.4129e+00	-4.2664	1.989e-05	***
## hour_13	26	0	-8.7513e+00	1.4188e+00	-6.1680	6.952e-10	***
## hour_14	27	0	-6.4458e+00	1.4069e+00	-4.5815	4.624e-06	***
## hour_15	28	0	-1.9415e+00	1.3882e+00	-1.3986	0.1619377	
## hour_16	29	0	3.6883e+00	1.3592e+00	2.7136	0.0066581	**
## hour_17	30	0	1.3920e+01	1.3180e+00	10.5614	< 2.2e-16	***
## hour_18	31	0	1.8904e+01	1.2515e+00	15.1049	< 2.2e-16	***
## hour_19	32	0	1.5360e+01	1.2098e+00	12.6967	< 2.2e-16	***
## hour_20	33	0	8.0379e+00	1.1584e+00	6.9391	3.979e-12	***
## hour_21	34	0	3.5557e+00	1.1126e+00	3.1957	0.0013954	**
## hour_22	35	0	3.4225e+00	1.0698e+00	3.1992	0.0013785	**
## hour_23	36	0	-2.3987e+00	1.0416e+00	-2.3029	0.0212866	*
## month_2	37	0	5.2213e+00	6.9419e-01	7.5214	5.480e-14	***
## month_3	38	0	1.2214e+01	6.9551e-01	17.5606	< 2.2e-16	***
## month_4	39	0	1.4991e+01	7.9216e-01	18.9247	< 2.2e-16	***
## month_5	40	0	1.6877e+01	8.9936e-01	18.7651	< 2.2e-16	***
## month_6	41	0	2.8414e+01	9.4585e-01	30.0410	< 2.2e-16	***
## month_7	42	0	2.8659e+01	9.1942e-01	31.1707	< 2.2e-16	***
## month_8	43	0	4.6855e+01	9.0613e-01	51.7093	< 2.2e-16	***
## month_9	44	0	3.0546e+01	8.2174e-01	37.1717	< 2.2e-16	***
## month_10	45	0	6.8512e+00	7.6636e-01	8.9399	< 2.2e-16	***
## month_11	46	0	1.2208e-01	7.2122e-01	0.1693	0.8655874	
## month_12	47	0	1.3598e+01	7.1350e-01	19.0575	< 2.2e-16	***
## net_imports_SE	48	0	3.6400e-03	5.7645e-04	6.3145	2.725e-10	***
## net_imports_PL	49	0	-8.7696e-03	4.1434e-04	-21.1650	< 2.2e-16	***
## net_imports_CZ	50	0	7.6347e-03	2.4742e-04	30.8570	< 2.2e-16	***
## net_imports_BE	51	0	2.0309e-02	4.7000e-04	43.2104	< 2.2e-16	***
## net_imports_CH	52	0	4.8365e-03	1.6588e-04	29.1573	< 2.2e-16	***
## net_imports_NO	53	0	4.3056e-02	4.1138e-04	104.6608	< 2.2e-16	***
## net_imports_NL	54	0	6.7013e-03	1.3600e-04	49.2753	< 2.2e-16	***

```

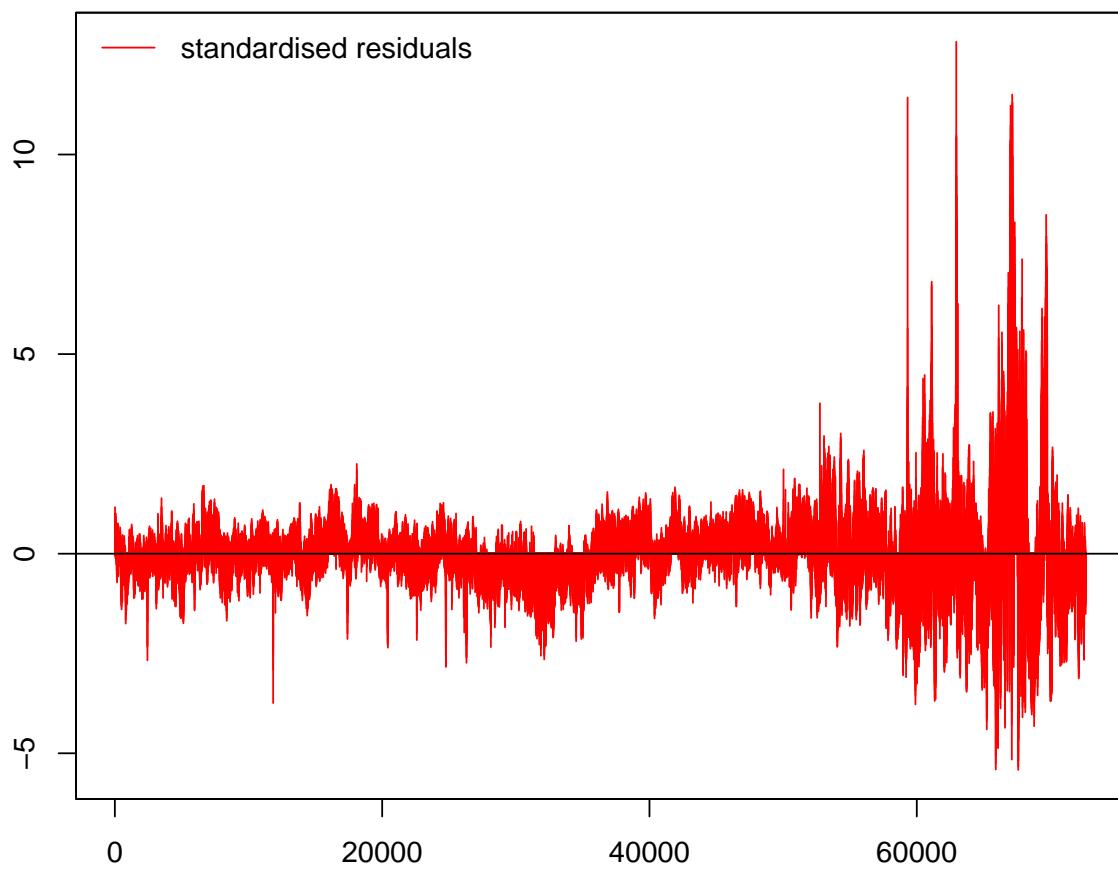
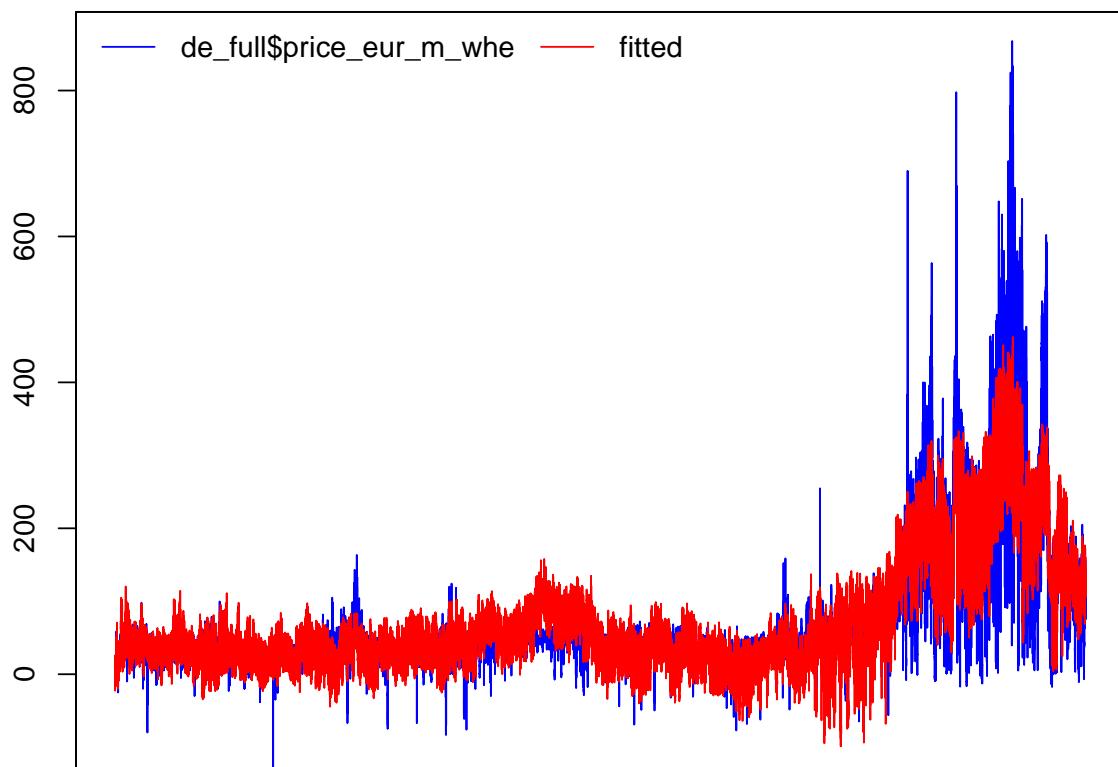
## net_imports_LU      55    0 -8.9564e-03  8.8479e-04 -10.1226 < 2.2e-16 ***
## net_imports_FR      56    0 -6.0007e-03  1.2437e-04 -48.2486 < 2.2e-16 ***
## net_imports_DK      57    0  1.9052e-03  1.7998e-04  10.5857 < 2.2e-16 ***
## net_imports_AT      58    0 -5.1762e-05  2.7211e-04 -0.1902  0.8491369
## log_redisp_mwh     59    0 -4.7652e-01  5.1344e-02 -9.2809 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Diagnostics:
##
##                  Chi-sq df p-value
## Ljung-Box AR(1)   66776  1      0
## Ljung-Box ARCH(1) 63312  1      0

## 9 path(s) to search

## Searching: 1 2 3 4 5 6 7 8 9

##
## Path 1: 14 16 58 46 15 23 17 28 24
## Path 2: 15 46 58 23 17 16 14 28 24
## Path 3: 16 15 58 46 23 17 14 28 24
## Path 4: 17 23 46 58 15 16 14 28 24
## Path 5: 23 46 58 17 15 16 14 28 24
## Path 6: 24 15 58 46 28 16 14 36 17
## Path 7: 28 15 58 46 16 24 14 36 17
## Path 8: 46 23 58 17 15 16 14 28 24
## Path 9: 58 23 46 17 15 16 14 28 24
##
## Terminal models:
##
##          info(sc)      logl      n  k
## spec 1 (1-cut): 10.14578 -368209.8 72639 50
## spec 2:         10.14576 -368209.0 72639 50
##
## Retained regressors (final model):
##
## mconst  log_load  log_gen_gas_mw  log_gen_coal_mw  log_gen_hydro_mw  log_gen_geoth_mw  log_gen_oil_mw
## de_levs_sel %>% plot

```



```
de_levs_sel_logewma <- getsm(de_mod_levs_logewma, ar.LjungB = NULL, arch.LjungB = NULL)
```

```
##  
## GUM mean equation:  
##  
##  
## reg.no. keep coef std.error t-stat p-value  
## mconst 1 0 3.2813e+02 2.0264e+01 16.1926 < 2.2e-16 ***  
## log_load 2 0 1.4264e+01 1.7982e+00 7.9327 2.174e-15 ***  
## log_gen_gas_mw 3 0 2.9921e+01 4.1248e-01 72.5390 < 2.2e-16 ***  
## log_gen_coal_mw 4 0 -1.0527e+01 6.9903e-01 -15.0591 < 2.2e-16 ***  
## log_gen_hydro_mw 5 0 -4.2343e+00 4.9264e-01 -8.5953 < 2.2e-16 ***  
## log_gen_geoth_mw 6 0 1.4893e+01 7.3486e-01 20.2670 < 2.2e-16 ***  
## log_gen_nuclear_mw 7 0 -2.3938e+01 6.8937e-01 -34.7246 < 2.2e-16 ***  
## log_gen_oth_ren_mw 8 0 1.4739e+01 7.9203e-01 18.6093 < 2.2e-16 ***  
## log_gen_solar_mw 9 0 4.9121e-01 8.9646e-02 5.4795 4.280e-08 ***  
## log_gen_wind_mw 10 0 6.2983e-01 2.5612e-01 2.4591 0.0139312 *  
## log_PNGASEU 11 0 8.5731e+01 9.4949e-01 90.2918 < 2.2e-16 ***  
## log_PALLFNF 12 0 9.4051e+01 3.5942e+00 26.1672 < 2.2e-16 ***  
## log_PINDU 13 0 -2.2499e+02 2.1116e+00 -106.5507 < 2.2e-16 ***  
## hour_1 14 0 -1.7632e+00 1.0281e+00 -1.7149 0.0863622 .  
## hour_2 15 0 -5.8969e-01 1.0359e+00 -0.5693 0.5691754  
## hour_3 16 0 -9.6671e-01 1.0444e+00 -0.9256 0.3546674  
## hour_4 17 0 3.1438e-01 1.0517e+00 0.2989 0.7649997  
## hour_5 18 0 3.1168e+00 1.0512e+00 2.9651 0.0030273 **  
## hour_6 19 0 9.6354e+00 1.0577e+00 9.1100 < 2.2e-16 ***  
## hour_7 20 0 1.4310e+01 1.0842e+00 13.1983 < 2.2e-16 ***  
## hour_8 21 0 1.1163e+01 1.1569e+00 9.6494 < 2.2e-16 ***  
## hour_9 22 0 4.8684e+00 1.2666e+00 3.8435 0.0001214 ***  
## hour_10 23 0 9.3291e-02 1.3436e+00 0.0694 0.9446430  
## hour_11 24 0 -2.4434e+00 1.3952e+00 -1.7513 0.0798985 .  
## hour_12 25 0 -6.0279e+00 1.4129e+00 -4.2664 1.989e-05 ***  
## hour_13 26 0 -8.7513e+00 1.4188e+00 -6.1680 6.952e-10 ***  
## hour_14 27 0 -6.4458e+00 1.4069e+00 -4.5815 4.624e-06 ***  
## hour_15 28 0 -1.9415e+00 1.3882e+00 -1.3986 0.1619377  
## hour_16 29 0 3.6883e+00 1.3592e+00 2.7136 0.0066581 **  
## hour_17 30 0 1.3920e+01 1.3180e+00 10.5614 < 2.2e-16 ***  
## hour_18 31 0 1.8904e+01 1.2515e+00 15.1049 < 2.2e-16 ***  
## hour_19 32 0 1.5360e+01 1.2098e+00 12.6967 < 2.2e-16 ***  
## hour_20 33 0 8.0379e+00 1.1584e+00 6.9391 3.979e-12 ***  
## hour_21 34 0 3.5557e+00 1.1126e+00 3.1957 0.0013954 **  
## hour_22 35 0 3.4225e+00 1.0698e+00 3.1992 0.0013785 **  
## hour_23 36 0 -2.3987e+00 1.0416e+00 -2.3029 0.0212866 *  
## month_2 37 0 5.2213e+00 6.9419e-01 7.5214 5.480e-14 ***  
## month_3 38 0 1.2214e+01 6.9551e-01 17.5606 < 2.2e-16 ***  
## month_4 39 0 1.4991e+01 7.9216e-01 18.9247 < 2.2e-16 ***  
## month_5 40 0 1.6877e+01 8.9936e-01 18.7651 < 2.2e-16 ***  
## month_6 41 0 2.8414e+01 9.4585e-01 30.0410 < 2.2e-16 ***  
## month_7 42 0 2.8659e+01 9.1942e-01 31.1707 < 2.2e-16 ***  
## month_8 43 0 4.6855e+01 9.0613e-01 51.7093 < 2.2e-16 ***  
## month_9 44 0 3.0546e+01 8.2174e-01 37.1717 < 2.2e-16 ***  
## month_10 45 0 6.8512e+00 7.6636e-01 8.9399 < 2.2e-16 ***  
## month_11 46 0 1.2208e-01 7.2122e-01 0.1693 0.8655874  
## month_12 47 0 1.3598e+01 7.1350e-01 19.0575 < 2.2e-16 ***
```

```

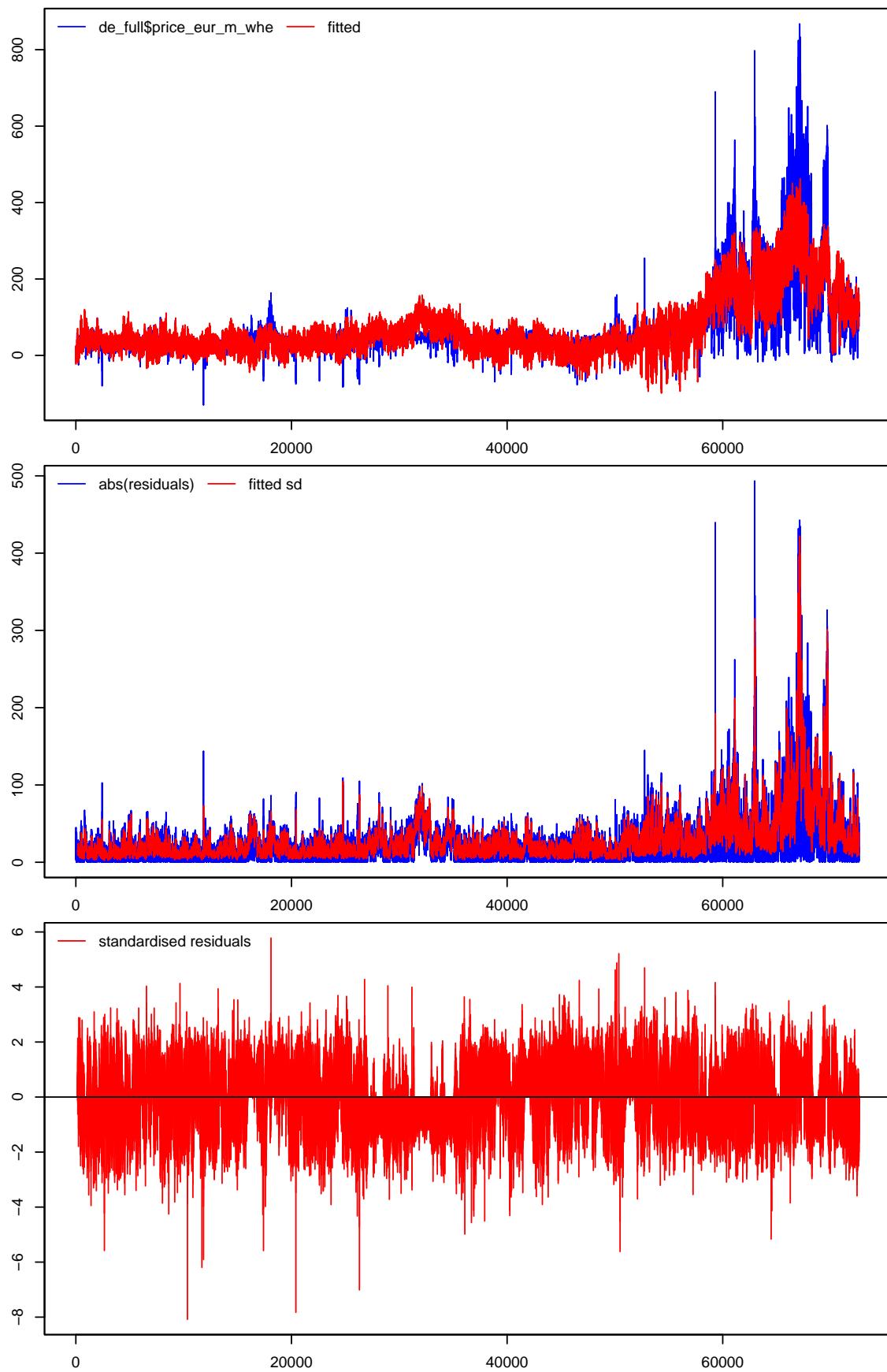
## net_imports_SE      48    0  3.6400e-03  5.7645e-04   6.3145 2.725e-10 ***
## net_imports_PL      49    0 -8.7696e-03  4.1434e-04  -21.1650 < 2.2e-16 ***
## net_imports_CZ      50    0  7.6347e-03  2.4742e-04   30.8570 < 2.2e-16 ***
## net_imports_BE      51    0  2.0309e-02  4.7000e-04   43.2104 < 2.2e-16 ***
## net_imports_CH      52    0  4.8365e-03  1.6588e-04   29.1573 < 2.2e-16 ***
## net_imports_NO      53    0  4.3056e-02  4.1138e-04   104.6608 < 2.2e-16 ***
## net_imports_NL      54    0  6.7013e-03  1.3600e-04   49.2753 < 2.2e-16 ***
## net_imports_LU      55    0 -8.9564e-03  8.8479e-04  -10.1226 < 2.2e-16 ***
## net_imports_FR      56    0 -6.0007e-03  1.2437e-04  -48.2486 < 2.2e-16 ***
## net_imports_DK      57    0  1.9052e-03  1.7998e-04   10.5857 < 2.2e-16 ***
## net_imports_AT      58    0 -5.1762e-05  2.7211e-04  -0.1902  0.8491369
## log_redisp_mwh     59    0 -4.7652e-01  5.1344e-02  -9.2809 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## GUM log-variance equation:
##
##             coef std.error t-stat p-value
## vconst      -0.076130  0.038342  3.9425  0.04708 *
## logEqWMA(12)  0.799385  0.013005 61.4696 < 2.2e-16 ***
## logEqWMA(24)  0.183992  0.015948 11.5372 < 2.2e-16 ***
## logEqWMA(168) 0.061593  0.010397  5.9243  3.15e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Diagnostics:
##
##             Chi-sq df p-value
## Ljung-Box AR(1) 57845  1      0
## Ljung-Box ARCH(1) 25150  1      0

## 9 path(s) to search
## Searching: 1 2 3 4 5 6 7 8 9

##
## Path 1: 14 16 58 46 15 23 17 28 24
## Path 2: 15 46 58 23 17 16 14 28 24
## Path 3: 16 15 58 46 23 17 14 28 24
## Path 4: 17 23 46 58 15 16 14 28 24
## Path 5: 23 46 58 17 15 16 14 28 24
## Path 6: 24 15 58 46 28 16 14 36 17
## Path 7: 28 15 58 46 16 24 14 36 17
## Path 8: 46 23 58 17 15 16 14 28 24
## Path 9: 58 23 46 17 15 16 14 28 24
##
## Terminal models:
##
##             info(sc)      logl      n k
## spec 1 (1-cut): 9.010506 -326977.2 72639 50
## spec 2:          9.010651 -326982.5 72639 50
##
## Retained regressors (final model):
##
## mconst  log_load  log_gen_gas_mw  log_gen_coal_mw  log_gen_hydro_mw  log_gen_geoth_mw  log_g

```

```
de_levs_sel_logewma %>% plot
```



```

reg_mat_shares <- as.matrix(select(de_full,
                                     #'shr_gen_biomass_mw',
                                     #'shr_gen_gas_mw',
                                     #'shr_gen_coal_mw',
                                     #'shr_gen_hydro_mw',
                                     #'shr_gen_geoth_mw',
                                     #'shr_gen_nuclear_mw',
                                     #'shr_gen_oth_ren_mw',
                                     #'shr_gen_solar_mw',
                                     #'shr_gen_wind_mw',
                                     'log_PNGASEU',
                                     'log_PALLFNF',
                                     'log_PINDU',
                                     contains('hour'),
                                     contains('month'),
                                     "log_redisp_mwh",
                                     shr_net_imports)) # 'gen_wind_kw' 'gen_coal_kw', 'gen_hydro_kw'

de_mod_shares <- arx(de_full$price_eur_m_whe,
                      mxreg = reg_mat_shares)

de_mod_shares_logewma <- arx(de_full$price_eur_m_whe,
                               log.ewma = c(12, 24, 168),
                               mxreg = reg_mat_shares)

de_shares_sel <- getsm(de_mod_shares, ar.LjungB = NULL, arch.LjungB = NULL)

```

Shares as Exp. Vars.

```

##  

## GUM mean equation:  

##  

##  

##      reg.no.  keep      coef  std.error   t-stat   p-value  

## mconst           1    0  342.157005  11.476998  29.8124 < 2.2e-16 ***  

## shr_gen_gas_kw  2    0  506.020085   5.163188  98.0053 < 2.2e-16 ***  

## shr_gen_coal_kw 3    0 -159.292461   4.174788 -38.1558 < 2.2e-16 ***  

## shr_gen_hydro_kw 4    0  -69.768475   6.881020 -10.1393 < 2.2e-16 ***  

## shr_gen_geoth_kw 5    0 4240.473991 2139.530578   1.9820  0.0474870 *  

## shr_gen_nuclear_kw 6    0 -363.784607   6.868355 -52.9653 < 2.2e-16 ***  

## shr_gen_oth_ren_kw 7    0 9602.354101 338.542280  28.3638 < 2.2e-16 ***  

## shr_gen_solar_kw  8    0  -71.195809   3.026692 -23.5226 < 2.2e-16 ***  

## shr_gen_wind_kw  9    0 -130.492239   3.078198 -42.3924 < 2.2e-16 ***  

## log_PNGASEU     10   0   87.007975   0.969775  89.7198 < 2.2e-16 ***  

## log_PALLFNF     11   0  154.366176   3.650778  42.2831 < 2.2e-16 ***  

## log_PINDU       12   0 -236.042424   2.278151 -103.6114 < 2.2e-16 ***  

## hour_1           13   0  -1.755027   1.138289  -1.5418  0.1231238  

## hour_2           14   0  -1.497033   1.145231  -1.3072  0.1911528  

## hour_3           15   0  -3.236108   1.153007  -2.8067  0.0050070 **  

## hour_4           16   0  -4.542115   1.158567  -3.9205 8.846e-05 ***  

## hour_5           17   0  -4.810603   1.160486  -4.1453 3.397e-05 ***  

## hour_6           18   0  -1.360620   1.195198  -1.1384  0.2549547  

## hour_7           19   0   3.181494   1.231972   2.5824  0.0098124 **

```

```

## hour_8          20    0   1.532457   1.254420   1.2216 0.2218453
## hour_9          21    0  -3.962872   1.264279  -3.1345 0.0017222 ** 
## hour_10         22    0  -8.565204   1.290766  -6.6358 3.251e-11 *** 
## hour_11         23    0 -10.046090   1.328995  -7.5592 4.104e-14 *** 
## hour_12         24    0 -10.775257   1.346586  -8.0019 1.243e-15 *** 
## hour_13         25    0 -10.671208   1.360772  -7.8420 4.493e-15 *** 
## hour_14         26    0  -8.771060   1.356596  -6.4655 1.016e-10 *** 
## hour_15         27    0  -4.961998   1.338842  -3.7062 0.0002106 *** 
## hour_16         28    0  -0.812991   1.311922  -0.6197 0.5354611
## hour_17         29    0   7.161501   1.294859   5.5307 3.200e-08 *** 
## hour_18         30    0  11.833023   1.269717   9.3194 < 2.2e-16 *** 
## hour_19         31    0  12.164281   1.244801   9.7721 < 2.2e-16 *** 
## hour_20         32    0   6.631773   1.202592   5.5146 3.508e-08 *** 
## hour_21         33    0   0.017882   1.180560   0.0151 0.9879150
## hour_22         34    0  -1.739297   1.165327  -1.4925 0.1355619
## hour_23         35    0  -5.182766   1.150155  -4.5061 6.612e-06 *** 
## month_2          36    0   4.873959   0.762041   6.3959 1.605e-10 *** 
## month_3          37    0  13.940209   0.762768  18.2758 < 2.2e-16 *** 
## month_4          38    0  14.159736   0.836582  16.9257 < 2.2e-16 *** 
## month_5          39    0  14.477640   0.928678  15.5895 < 2.2e-16 *** 
## month_6          40    0  22.608480   0.946742  23.8803 < 2.2e-16 *** 
## month_7          41    0  24.020452   0.900792  26.6659 < 2.2e-16 *** 
## month_8          42    0  44.952552   0.892767  50.3520 < 2.2e-16 *** 
## month_9          43    0  31.260794   0.838983  37.2603 < 2.2e-16 *** 
## month_10         44    0  13.069373   0.811585  16.1035 < 2.2e-16 *** 
## month_11         45    0   7.290577   0.782804   9.3134 < 2.2e-16 *** 
## month_12         46    0  15.754549   0.775938  20.3039 < 2.2e-16 *** 
## log_redisp_mwh   47    0  -0.496050   0.056503  -8.7792 < 2.2e-16 *** 
## shr_net_imports  48    0  14.310542   3.970849   3.6039 0.0003137 *** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ',' 1
## 
## 
## Diagnostics:
## 
##           Chi-sq df p-value
## Ljung-Box AR(1) 67854  1     0
## Ljung-Box ARCH(1) 63977  1     0

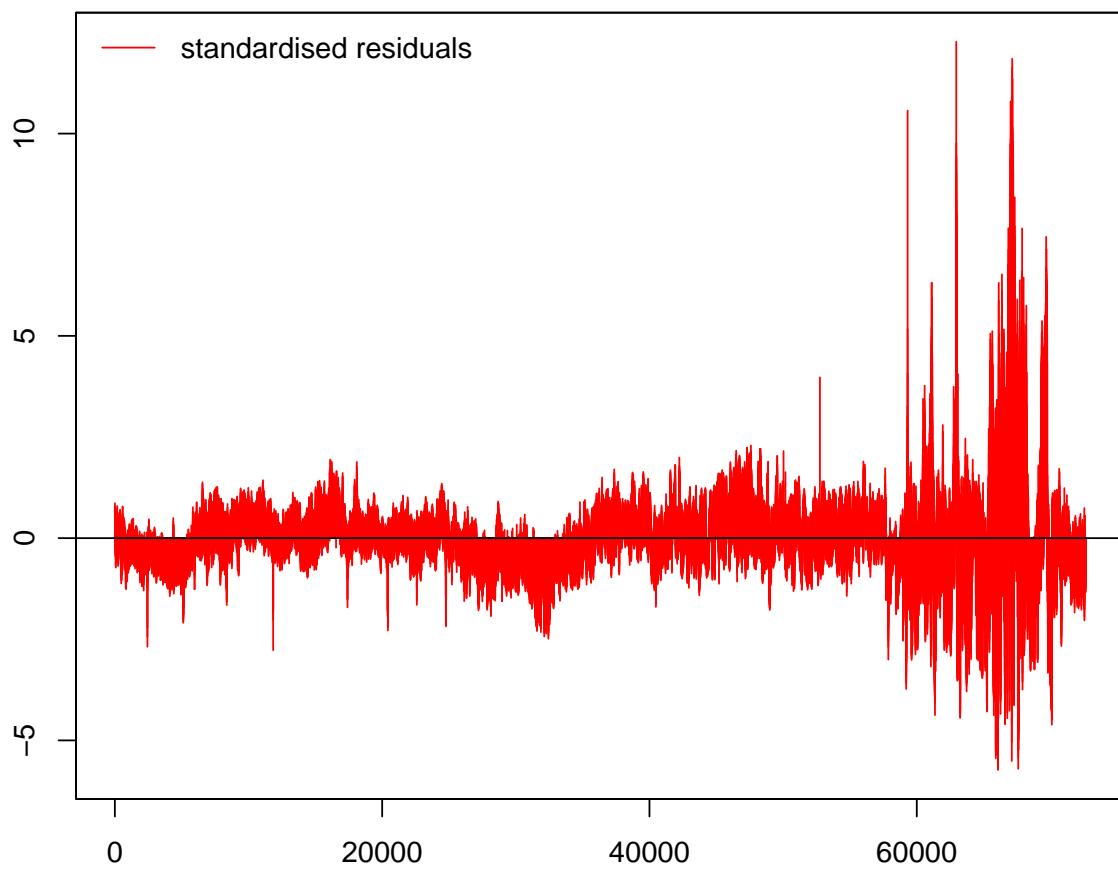
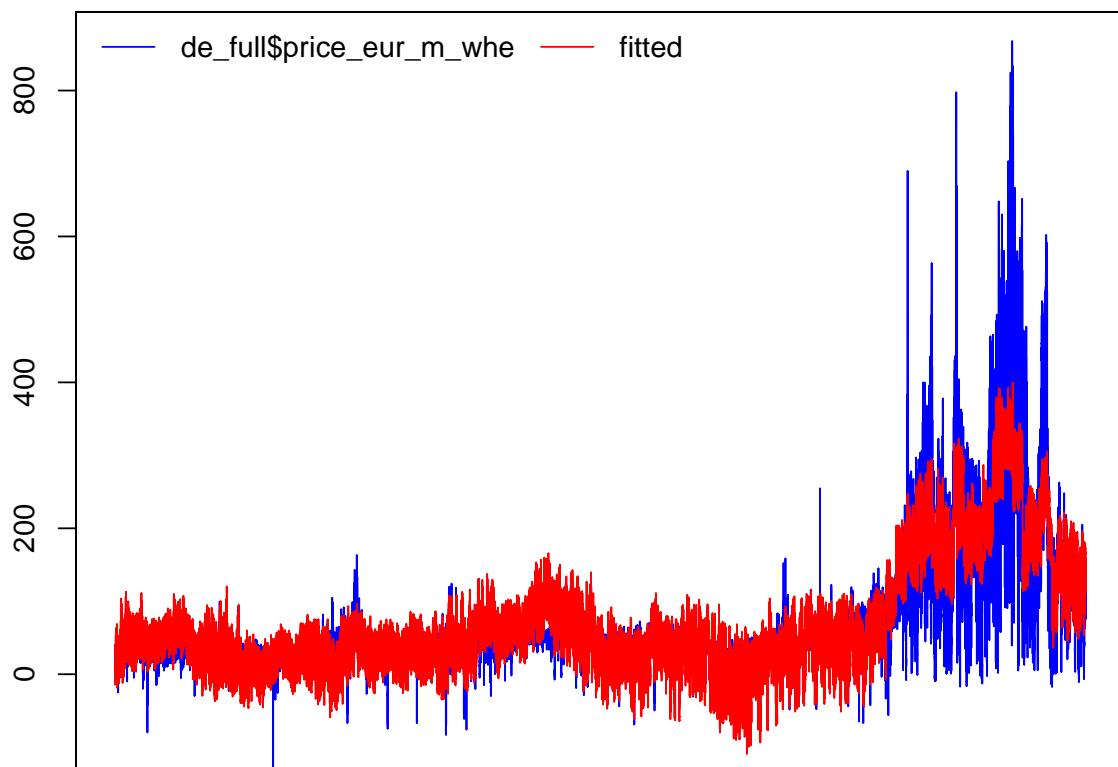
## 7 path(s) to search

## Searching: 1 2 3 4 5 6 7

## 
## Path 1: 13 28 18 14 34 33 5
## Path 2: 14 28 18 34 13 33 5
## Path 3: 18 28 33 14 13 34 5
## Path 4: 20 33 28 14 18 13 34 -5
## Path 5: 28 33 18 14 13 34 5
## Path 6: 33 28 18 14 13 34 5
## Path 7: 34 28 18 14 13 33 5
## 
## Terminal models:

```

```
##  
##           info(sc)      logl      n  k  
## spec 1 (1-cut): 10.35056 -375697.7 72639 41  
## spec 2:          10.35050 -375695.5 72639 41  
##  
## Retained regressors (final model):  
##  
##   mconst   shr_gen_gas_mw   shr_gen_coal_mw   shr_gen_hydro_mw   shr_gen_nuclear_mw   shr_gen_oth_ren  
  
de_shares_sel %>% plot
```



```

de_shares_sel_logewma <- getsm(de_mod_shares_logewma, ar.LjungB = NULL, arch.LjungB = NULL)

##  

## GUM mean equation:  

##  

##  

##          reg.no.  keep      coef  std.error   t-stat  p-value  

## mconst             1    0  342.157005  11.476998  29.8124 < 2.2e-16 ***  

## shr_gen_gas_mw    2    0  506.020085   5.163188  98.0053 < 2.2e-16 ***  

## shr_gen_coal_mw   3    0 -159.292461   4.174788 -38.1558 < 2.2e-16 ***  

## shr_gen_hydro_mw  4    0 -69.768475   6.881020 -10.1393 < 2.2e-16 ***  

## shr_gen_geoth_mw  5    0 4240.473991 2139.530578   1.9820  0.0474870 *  

## shr_gen_nuclear_mw 6    0 -363.784607   6.868355 -52.9653 < 2.2e-16 ***  

## shr_gen_oth_ren_mw 7    0 9602.354101 338.542280  28.3638 < 2.2e-16 ***  

## shr_gen_solar_mw   8    0 -71.195809   3.026692 -23.5226 < 2.2e-16 ***  

## shr_gen_wind_mw    9    0 -130.492239   3.078198 -42.3924 < 2.2e-16 ***  

## log_PNGASEU       10   0   87.007975   0.969775  89.7198 < 2.2e-16 ***  

## log_PALLFNF        11   0  154.366176   3.650778  42.2831 < 2.2e-16 ***  

## log_PINDU          12   0 -236.042424   2.278151 -103.6114 < 2.2e-16 ***  

## hour_1              13   0 -1.755027   1.138289  -1.5418 0.1231238  

## hour_2              14   0 -1.497033   1.145231  -1.3072 0.1911528  

## hour_3              15   0 -3.236108   1.153007  -2.8067 0.0050070 **  

## hour_4              16   0 -4.542115   1.158567  -3.9205 8.846e-05 ***  

## hour_5              17   0 -4.810603   1.160486  -4.1453 3.397e-05 ***  

## hour_6              18   0 -1.360620   1.195198  -1.1384 0.2549547  

## hour_7              19   0   3.181494   1.231972   2.5824 0.0098124 **  

## hour_8              20   0   1.532457   1.254420   1.2216 0.2218453  

## hour_9              21   0 -3.962872   1.264279  -3.1345 0.0017222 **  

## hour_10             22   0 -8.565204   1.290766  -6.6358 3.251e-11 ***  

## hour_11             23   0 -10.046090   1.328995 -7.5592 4.104e-14 ***  

## hour_12             24   0 -10.775257   1.346586  -8.0019 1.243e-15 ***  

## hour_13             25   0 -10.671208   1.360772  -7.8420 4.493e-15 ***  

## hour_14             26   0 -8.771060   1.356596  -6.4655 1.016e-10 ***  

## hour_15             27   0 -4.961998   1.338842  -3.7062 0.0002106 ***  

## hour_16             28   0 -0.812991   1.311922  -0.6197 0.5354611  

## hour_17             29   0   7.161501   1.294859   5.5307 3.200e-08 ***  

## hour_18             30   0  11.833023   1.269717   9.3194 < 2.2e-16 ***  

## hour_19             31   0  12.164281   1.244801  9.7721 < 2.2e-16 ***  

## hour_20             32   0   6.631773   1.202592   5.5146 3.508e-08 ***  

## hour_21             33   0   0.017882   1.180560   0.0151 0.9879150  

## hour_22             34   0 -1.739297   1.165327  -1.4925 0.1355619  

## hour_23             35   0 -5.182766   1.150155  -4.5061 6.612e-06 ***  

## month_2              36   0   4.873959   0.762041   6.3959 1.605e-10 ***  

## month_3              37   0  13.940209   0.762768  18.2758 < 2.2e-16 ***  

## month_4              38   0  14.159736   0.836582  16.9257 < 2.2e-16 ***  

## month_5              39   0  14.477640   0.928678  15.5895 < 2.2e-16 ***  

## month_6              40   0  22.608480   0.946742  23.8803 < 2.2e-16 ***  

## month_7              41   0  24.020452   0.900792  26.6659 < 2.2e-16 ***  

## month_8              42   0  44.952552   0.892767  50.3520 < 2.2e-16 ***  

## month_9              43   0  31.260794   0.838983  37.2603 < 2.2e-16 ***  

## month_10             44   0  13.069373   0.811585  16.1035 < 2.2e-16 ***  

## month_11             45   0   7.290577   0.782804   9.3134 < 2.2e-16 ***  

## month_12             46   0  15.754549   0.775938  20.3039 < 2.2e-16 ***  

## log_redispmwh        47   0 -0.496050   0.056503  -8.7792 < 2.2e-16 ***

```

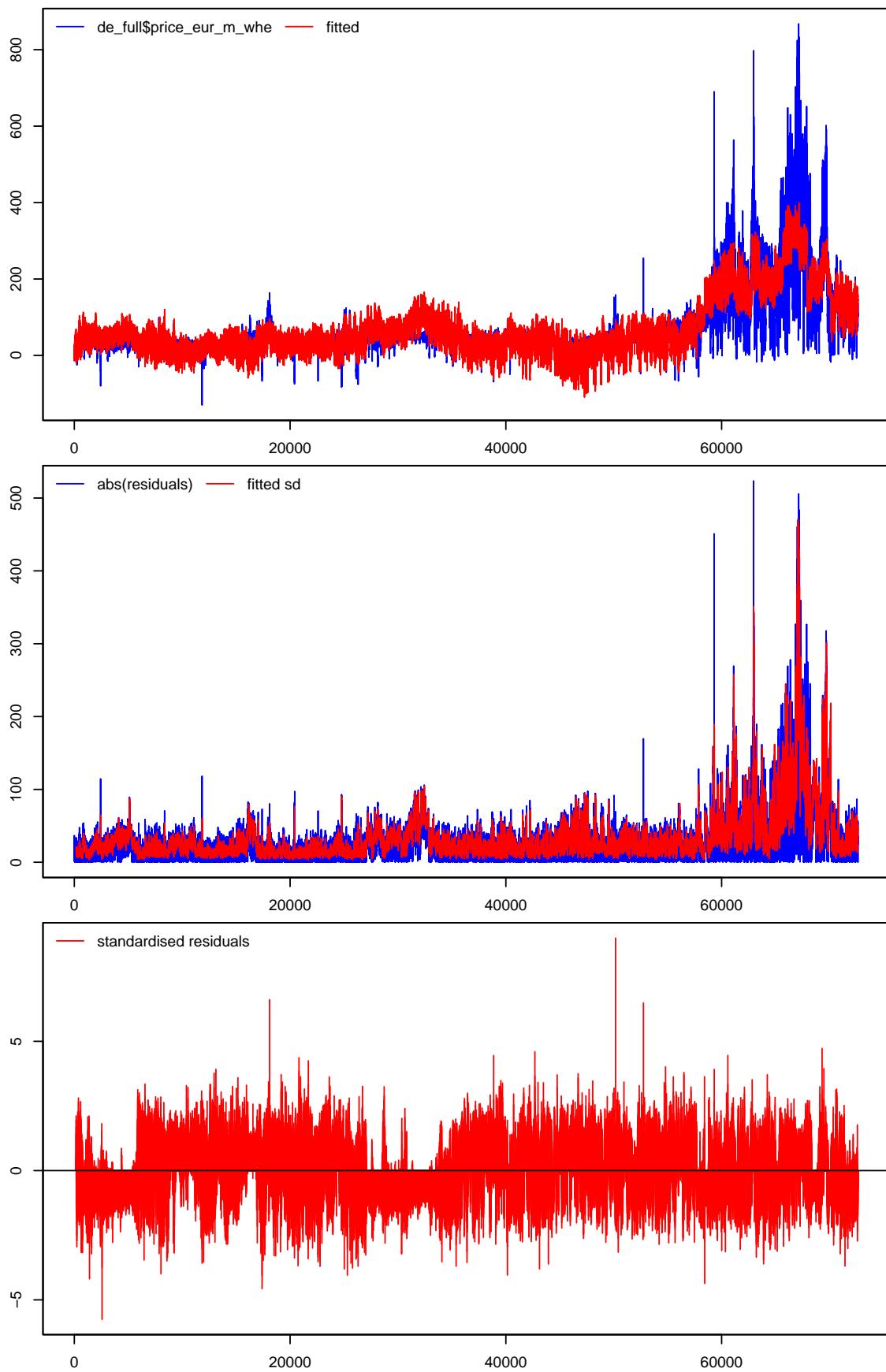
```

## shr_net_imports      48     0   14.310542    3.970849    3.6039 0.0003137 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## GUM log-variance equation:
##
##           coef std.error t-stat p-value
## vconst      -0.154083  0.040555 14.4350 0.0001451 ***
## logEqWMA(12)  0.719664  0.012424 57.9237 < 2.2e-16 ***
## logEqWMA(24)  0.293390  0.015561 18.8546 < 2.2e-16 ***
## logEqWMA(168) 0.041720  0.010367  4.0242 5.723e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Diagnostics:
##
##           Chi-sq df p-value
## Ljung-Box AR(1) 61587  1     0
## Ljung-Box ARCH(1) 31776  1     0

## 7 path(s) to search
## Searching: 1 2 3 4 5 6 7

##
## Path 1: 13 28 18 14 34 33 5
## Path 2: 14 28 18 34 13 33 5
## Path 3: 18 28 33 14 13 34 5
## Path 4: 20 33 28 14 18 13 34 -5
## Path 5: 28 33 18 14 13 34 5
## Path 6: 33 28 18 14 13 34 5
## Path 7: 34 28 18 14 13 33 5
##
## Terminal models:
##
##           info(sc)      logl      n k
## spec 1 (1-cut): 9.251739 -335789.1 72639 41
## spec 2:         9.251361 -335775.3 72639 41
##
## Retained regressors (final model):
##
## mconst  shr_gen_gas_mw  shr_gen_coal_mw  shr_gen_hydro_mw  shr_gen_nuclear_mw  shr_gen_oth_ren
de_shares_sel_logewma %>% plot

```

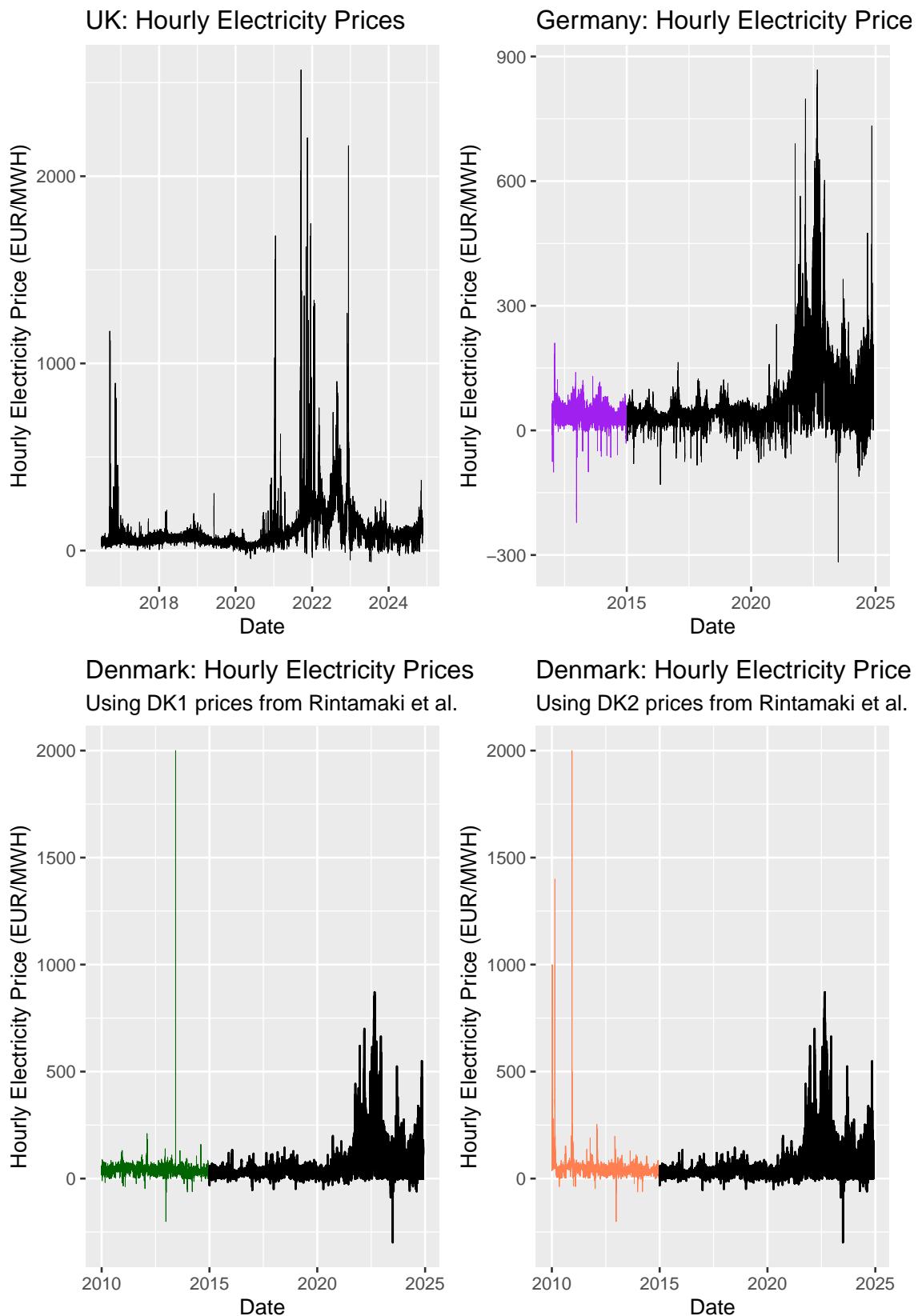


```
# # Perform gets model selection over the log-variance model above
# de_vmod <- getsv(de_mod, t.pval = 0.001, ar.LjungB = NULL)
#
# de_vmod %>% plot
#
# data_de %>% filter(de_price < 0)
```

(TBA) Spotlight on UK, Germany, Denmark

Here, see the time series for Germany and Denmark (to compare to the data available in Rintamaki et al. - color portion of the TS plots). UK spotlight to discuss potential interest to David.

Hourly Wholesale Electricity Price Data UK, Germany, and Denmark
 Black represents EMBER data, colorful data represents data from Rintamaki replication.



(TBA) Full Extension to Europe

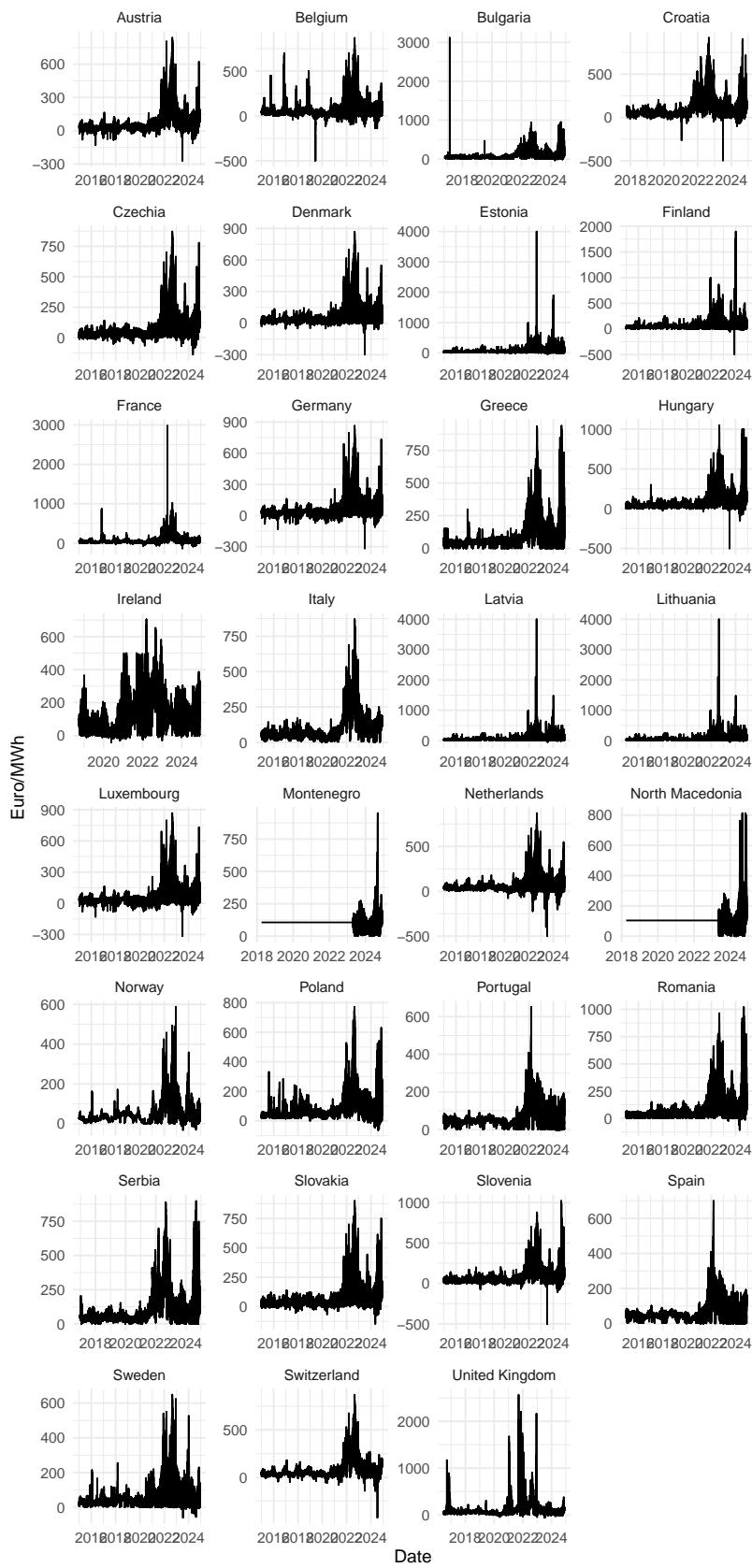
Note: Recall also that you can find our replication of <https://doi.org/10.1016/j.eneco.2016.12.019> Rintamaki et al. https://ebbam.github.io/renvol/code/rintamaki_repliation/rintamaki_repliation.html here. We have yet to analyse the

Hourly Prices

Using data from <https://ember-energy.org/data/european-wholesale-electricity-price-data/EMBER> on European wholesale electricity prices (hourly - daily and monthly also exist), The hourly prices are the day-ahead wholesale prices by country.

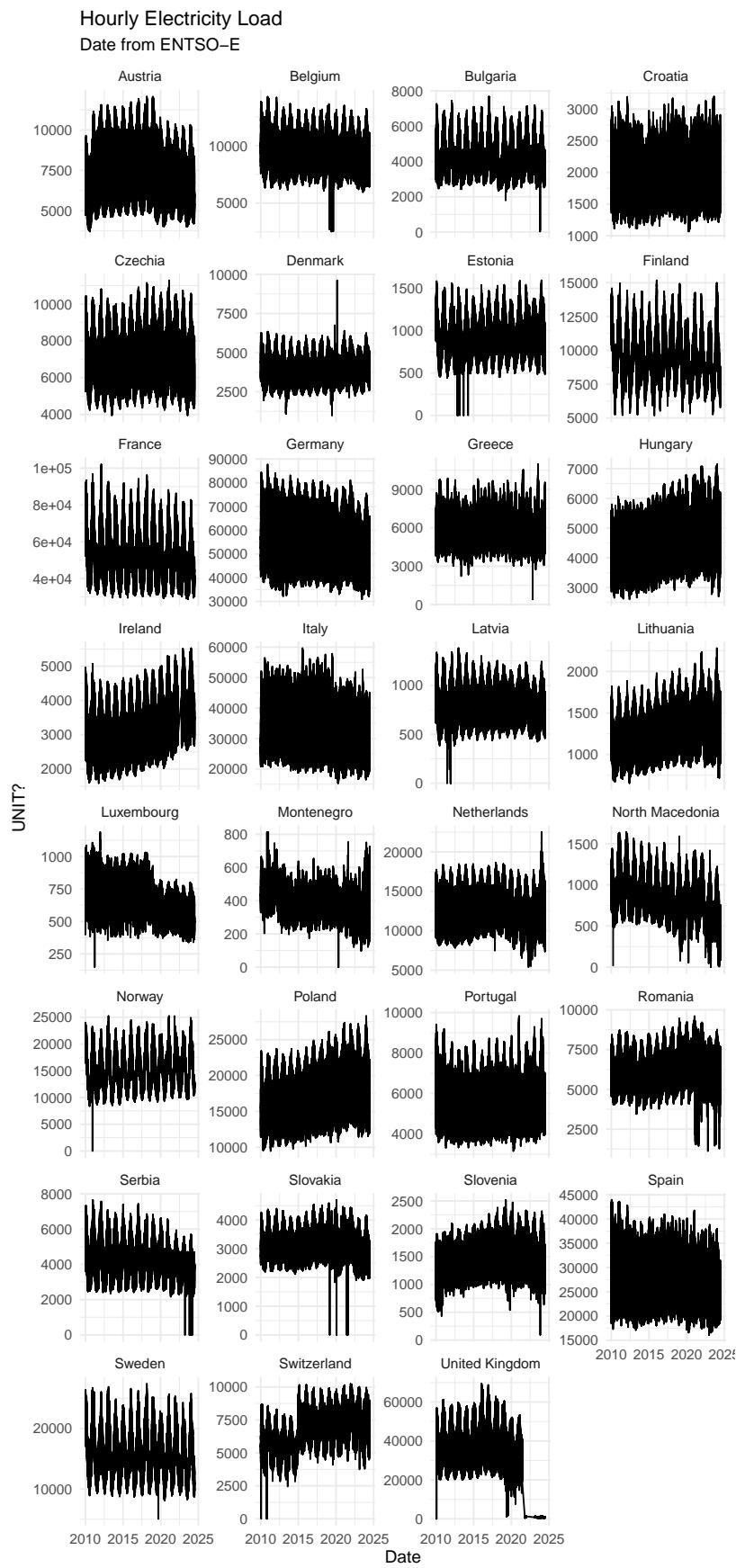
Hourly Wholesale Electricity Prices

Prices are average day-ahead spot prices per MWh sold per time period



The following is data from <https://www.entsoe.eu/data/power-stats/ENTSO-E> on hourly electricity loads.

```
## [1] TRUE
```



Daily and Monthly Prices

Daily and Monthly Wholesale Electricity Prices
Black (blue) represents daily (monthly) prices.

