Predicting Natural Gas Demand in Germany

Johann-Friedrich Salzmann
j.salzmann@students.hertie-school.org

Danial Riaz
d.riaz@students.hertie-school.org

Finn Krueger
f.krueger@students.hertie-school.org

Abstract

Predicting gas demand is one of the most crucial tasks for the German government today in order to limit unprecedented price hikes on consumers and create actionable policy to address the issue. Given the complexity of; the dependence of gas on various segments within the consumer base i.e. residential, industrial, electricity, the consequent impact on prices and the economy and the reverse impact of prices and economic situation on demand itself, makes modelling gas demand particularly challenging. More over, social factors, such as 'responsible demand usage' by consumers is also unprecedented and one needs to introduce a model that is able to both learn from past patterns of annual seasonality as well as modify from recent day-to-day patters of consumption reduction. We have therefore made use of Machine Learning techniques in our paper to see the extent to which we can model the continuously changing daily gas demand in Germany.

We make use of various modelling techniques. We run a SGD and Random Forest model and we compare the performance of chronological and random training data generation. Our results indicate that a random forest model with chronological training data is the most effective in predicting gas demand with an R squared value of greater than 99.9

1. Background

The Russia-Ukraine war has lead Europe towards the largest energy crisis since the oil price shock of 1973. Since mid-2021, natural gas prices have been on a steep rise, with average wholesale prices at the TTF spot market well above 100 €/MWh between October 2021 and mid-2022 and prices of around 240 €/MWh in August. This is about ten times higher than the long-term pre-Covid price levels of 15–20 €/MWh (Ruhnau et al 2022).

The average price that industry paid for natural gas has quintupled compared to pre-crisis levels. Ammonia and aluminum industries were the first ones to reduce production followed by paper, brick, and steel industries in March 2022 as the spot prices increased above 200 €/MWh. The price related response to industrial consumption is complicated to predict given the complex relationship of the elasticities of these commodities as well as the competitiveness of the industries in which they operate in.

Average German residential retail prices have also increased, but with an even larger time lag. Initial survey reports as well as market data from the German regulator BnetzA as well as German gas market area manager Trading Hub Europe (THE) have shown a reduction of gas consumption by households. Price is of course not the only factor resulting in this reduced consumption, surveys have highlighted ethical considerations by consumers too in response to the Russian invasion of Ukraine.

Germany is an interesting case study to explore gas demand, as it is the largest export market for Russian natural gas. Furthermore, natural gas plays an essential role in Germany's industrial production as well as space heating. Reductions in Germany can therefore make a substantial contribution to solving the crisis at a European level.

2. Proposed Method

The project's aim is to be able to nowcast and forecast German daily national gas demand. Depending on forecast data availability of our predicting variables, we may either use the forecast data to predict gas demand, or compute forecast approximators ourselves. We expect us to be able to do a decently well performing forecast of next-day and potentially next-week demand. Throughout the model refinement process, we will keep monitor how our models perform vs. what demand is actually realised few days later. Thereby, we will have the opportunity to make use of short feedback and learning cycles.

For a discussion of what data is required to predict gas demand in general, see our project proposal.

Much of the literature concerning the forecast of gas demand applies Time series (TS) models. They can be applied for a wide range of forecasting horizons (from annual to hourly) (Zhang, 2020). We also intend to build different TS models.

One of the models often discussed is a Autoregressive Integrated Moving Average (ARIMA), which is widely applied in demand forecasting. ARIMA models use differencing to convert a non-stationary time series into a stationary one, and then predict future values from historical data. These models use "auto" correlations and moving averages over residual errors in the data to forecast future values. For short term predictions, ARIMA models require only prior data of a time series to generalize the forecast. They are however, poor long term forecasting models and not ideal for identifying unprecedented events. ARIMA model was applied to forecast annual or monthly gas demand of Turkey, with the consideration of GDP and price of gas (E. Erdogdu, 2010) .

Seasonal Auto-Regressive Integrated Moving Average with exogenous factors, or SARIMAX, is an extension of the ARIMA class of models. SARIMAX extends on this framework by adding the capability to handle exogenous variables. Unlike the ARIMA, it is able to capture seasonality in the model. This enables it to provide qualified annual forecasting for demand of gas.

In our analysis, we base our demand prediction models for the mid-term report on two regression models. Since the present value of national gas demand depends on its past value (i.e. autocorrelated) we have included seven day lagged predictor variables for all independent variables that arent time implicit. In comparison to the proposed ARIMA model, we have started of by adding the lagged predictor variables manually and have not applied differencing the lagged vs non lagged predictors. Manually adding the lagged variables and not immediately applying ARIMA, allows us to to get a better understanding of Times-Series regression models by comparing our manual regression model to ARIMA. This will provide us with first-hand insight into various effects such as differencing on our predictive capabilites and facilitate our learning experiences.

The two regression models that we have used are:

Stochastic Gradient Descent (SGD) is an efficient optimization algorithm used to find the values of parameters/coefficients of functions that minimize a function. In other words, it is used for discriminative learning of linear classifiers under convex loss functions such as SVM and Logistic regression. It has been successfully applied to large-scale datasets because the update to the coefficients is performed for each training instance, rather than at the end of instances.

Decision Trees are versatile Machine Learning algorithms that can perform both classification and regression tasks, and even multioutput tasks. They are very powerful algorithms, capable of fitting complex datasets. They are a fundamental components of Random Forests. Random forests or random decision forests is an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time. They work particularly well when we have a large number of correlated predictors. In a Random forest, when building these decision trees, each time a split in a tree is considered, a random sample of m predictors is chosen as split candidates from the full set of p predictors. The split is allowed to use only one of those m predictors.

With that, we built two Machine Learning models that we keep performing, allowing us to constantly monitor how they perform over time while real-world data changes.

Our baseline model, as described in our previous report, is a simpler version of the SGD model.

3. Experiments

We have implemented the models with a tech stack of R (used for data merging, cleaning, imputation feature engineering) and Python (used for actual ML modeling)[1].

Data: We compiled a dataset consisting of 1833 daily observations (roughly 5 years) and 510 predictor variables, including transformations. In order to predict the daily national gas demand in Germany we use weather data from various cities within Germany. We combine them with general economic and day-to-day market price data.

- Daily national gas demand (merged): national gas demand measured in kWh, retrieved from the THE (Trading Hub Europe, joint company of all German regional gas net 2 work providers). Before October 2021, NetConnect and Gaspool shared the market area, so THE has made available historic demand data from both markets.

- Gas prices (merged and imputed): Daily gas spot price data was obtained through the Dutch TTF Natural Gas price data (Futures) available on investing.com and combined with Spot Market data retrieved from the European Energy Exchange (the European Energy Exchange (EEX) is an integration between Powernext and Gaspoint Nordic that publishes market data every business day after the conclusion of its settlement window)

- Weather data (cleaned and imputed): Weather data Average, minimum and maximum daily temperature measured in Celsius as well as other factors including precipitation, wind speed, and sun hours taken from Frankfurt, Berlin, Munich and Bonn retrieved from

[1]Code is available at `https://github.com/jfsalzmann/gasprices.git`

Meteostat (Meteostatis one of the largest vendors of open weather and climate data. The platform provides access to long-term time series of thousands of weather stations), allowing to feed the model with sufficient data on different weather conditions within the country.

- Co2 Prices: Carbon Emissions Futures Historical Data over the last 5 years retrieved from investment.com , reflecting daily Emissions pricing in Europe.

- Electricity Price: We obtained German electricity price data for the last 5 years through the DBEc1 Electricity futures data retrieved from investing.com.

- DAX: The DAX, (Deutscher Aktienindex, German stock index) is a stock market index consisting of the 40 major German blue chip companies trading on the Frankfurt Stock Exchange. Data on DAX is publicly available on several platforms and reflects the general German economic performance and general fluctuations.. We obtained the daily DAX data over the past 5 years retrieved from from Yahoo Finance.

- Quarterly GDP: German quarterly GDP data was obtained from data.worldbank.org for the last 5 years, reflecting the general German economic performance, too.

We use a variety of engineered features (in the following, $n \in N$ is the number of observations, whereas $i \in I$ are the model prediction variables).

- A number of numeric date based features is included (day, week, month, weekday, year), as well their cyclic sin and cos projections:

$$x^n_{i,sin} = \sin\left(2\pi \max_{n \in N}(x^n_i)\, x^n_i\right)$$

$$x^n_{i,cos} = \cos\left(2\pi \max_{n \in N}(x^n_i)\, x^n_i\right)$$

- For all variables, including date transformations, we add ln, square and elasticity transformations:

$$x^n_{i,ln} = \ln\left(x^n_i\right)$$

$$x^n_{i,sq} = \left(x^n_i\right)^2$$

$$x^n_{i,el} = \left((x^n_i)^2\right)^{-1}$$

- Additionally, a 7-day lagged copy of all variables except date features is added.

$$x^n_{i,lag(7)} = x^{n-7}_i$$

Evaluation method: We utilise and compare two different models: An SGD regressor prediction, and a Random Forest model with number of estimators set to 20.

Both models are run with a chronological train/test split of the data, and a random split. We compare the performance of both models for each split configuration in order to assess baseline (random split) results, which may be subject to overfitting, and realistic (chronological split) results, as new and unknown data the model is eventually designed to predict on always comes chronologically after the training data that is used.

We compare model results based on a visual, qualitative general impression of fit, as well as we compare quantitative indicators such as MSE and R-Squared.

Experimental details: Most of the data gets pulled automatically daily by API scripts or web scraping approaches. That will soon allow the models to be run every day over a period of several days in order to observe performance robustness.

Results: In our previous report, we proposed a baseline model with much less data that was generally uncleaned, unimputed and only for a short time frame of about a year, without any lagged features.

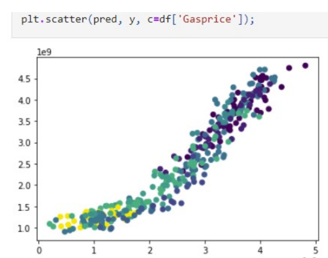

Figure 1. Baseline model: predicted vs. actual gas demand, training data

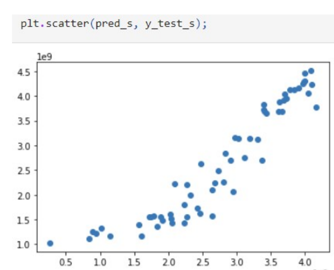

Figure 2. Baseline model: predicted vs. actual gas demand, test data

We had seen that the model performance was overall good based on qualitative assessment of the model plots. However, systematic model biases were visible and shaped our further research and modeling work.

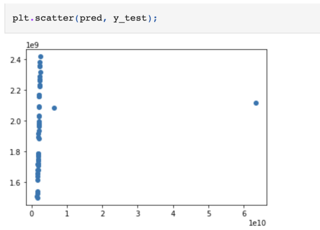

Figure 3. SGD model, chronological split: predicted vs. actual gas demand, test data

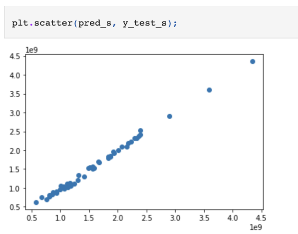

Figure 4. SGD model, random split: predicted vs. actual gas demand, test data

Our SDG model performs extremely well in the random split variant, however very badly in the chronological split variant. This suggests that we may have an issue with overfitting in place. Further investigation is needed how well the random split model is able to perform on unknown fresh data. The performance of the SGD chronological split may be further improved through configuration, resampling, and hyperparameter tuning.

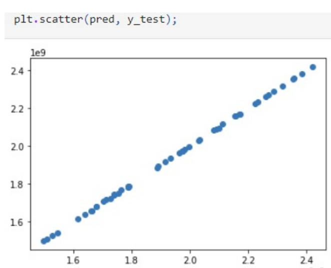

Figure 5. RF model, chronological split: predicted vs. actual gas demand, test data

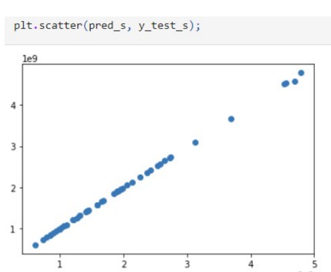

Figure 6. RF model, random split: predicted vs. actual gas demand, test data

Model	Quality	MAE	MSE	RMSE	R2
Baseline	Okay	*NA*	*NA*	*NA*	*NA*
SGD Chron.	Bad	1.6e9	7.5e19	8.7e9	-1131.1
SGD Rand.	Good	4.2e7	3.1e15	5.5e7	0.9947
RF Chron.	V. Good	1.8e6	5.1e12	2.3e6	0.9999
RF Rand.	V. Good	4.7e6	2.6e14	1.6e7	0.9997

Table 1. Model performance. We only report quantitative measures for SGD Linear, and Random Forest models.

Our Random Forest model performs extremely well, regardless of the splitting method. Further investigation is needed how well the random split model is able to perform on unknown fresh data, but signs are good that the model does the trick and was able to catch the underlying principles of gas demand.

Based on the quantitative measures, we can confirm that the Random Forest models, as appearent, perform best. Noteably, the chronological split Random Forest model even outperforms the random split, indicating that the model is able to understand the chronological features and time-based kind of the data.

4. Future work

In the following weeks we will compare our manual time series model to proposed methods such as ARIMA and SARMIAX to further our understanding into different effects of differencing. Additionally, we will assess how changing different parameters within our existing models can improve model performance, including hyperparameter tuning, and resampling methods. Our objective will be to identify the model that is the least computationally complex and is able to predict the dependent variable with the greatest efficiency.

Using new data in the next few weeks, we will assess the efficiency of the model in forecasting as well as nowcasting daily national aggregate gas demand. We will experiment how well the models will perform on a forecast base (e.g., weather forecast data for the next 10 days). After the forecast period has passed, we will compare the model performance with the actual materialised demand and take away further learnings of how well the model can work based on forecast predictors, which would be closest to a real world application scenario.

References

1. Al-Jazeera, "How bad could Europe's energy crisis get this winter?", 4th Oct 2022 https://www.aljazeera.com/news/2022/10/4/how-bad-could-europes-energy-crisis-get-this-winter

2. Erdogdu, E. (2010). Natural gas demand in Turkey.Applied Energy,87(1), 211-219.

3. J. D. Hunter, ”Matplotlib: A 2D Graphics Environment” (2007), Computing in Science Engineering, vol. 9, no. 3, pp. 90-95

4. Pandas development team, T., 2020. pandas-dev/pandas: Pandas. URL: https://doi.org/10.5281/zenodo.3509134, doi:10.5281/zenodo. 3509134.

5. Pedregosa, F., Varoquaux, Gael, Gramfort, A., Michel, V., Thirion, B., Grisel, O., … others. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12(Oct), 2825–2830.

6. Ruhnau, Oliver; Stiewe, Clemens; Muessel, Jarusch; Hirth, Lion (2022) : Gas demand in times of crisis: energy savings by consumer group in Germany, ZBW – Leibniz Information Centre for Economics, Kiel, Hamburg

7. Sadamori, IEA Press Release, 3rd Oct 2022 https://www.iea.org/news/natural-gas-markets-expected-to-remain-tight-into-2023-as-russia-further-reduces-supplies-to-europe

8. Su, H., Zio, E., Zhang, J., Xu, M., Li, X., Zhang, Z. (2019). A hybrid hourly natural gas demand forecasting method based on the integration of wavelet transform and enhanced Deep-RNN model.Energy,178, 585-597.”

9. Vinayak Sharma, Ümit Cali, Bhav Sardana, Murat Kuzlu, Dishant Banga, Manisa Pipattanasomporn (2021). Data-driven short-term natural gas demand forecasting with machine learning techniques. Journal of Petroleum Science and Engineering, (), –. doi:10.1016/j.petrol.2021.108979

10. van der Walt, S., Colbert, S.C., Varoquaux, G., 2011. The numpy array: A structure for efficient numerical computation. Computing in Science Engineering 13, 22–30.