Data^2

Group Project Data Mining

Lara-Aida Jopp (1978974)
Maximilian Heilmann (1979826)
Max Lautenbach (1980683)
Babett Müller (1979887)
Gregor Münker (1980671)
Niklas Weidenfeller (1977441)

submitted to the
Data and Web Science Group
Prof. Dr. Heiko Paulheim
University of Mannheim

December 2023

Contents

Chapter 1

Predicting the Residual Load

Due to the Ukraine war and its implied energy crisis, more and more countries, especially in Europe, are reviewing their national power supply. This energy crisis leads to an increasing importance of renewable energies. Nevertheless, the transformation to a renewable power supply is challenging. With its volatile production, studies reveal a gap of many gigawatts of ensured power supply [3]. To ensure the security of the energy supply, grid operators have to predict if they can meet the energy needs with renewable energies. The difference between renewable supply and energy need is called residual load. With a prediction of the residual load, grid operators could limit, for example, the energy supply for EVs or heat pumps [2].

Regression analysis was selected as the optimal method for accurately predicting residual load, given the continuous values of residual load measured in kilowatt-hours (kWh). It is assumed in this report that weather and seasonality have the most significant impact on residual load, both from an energy production and usage standpoint.

The consumed data set consists of two sources establishing all necessary attributes to predict the target for four locations scattered over Germany representing the German electricity grid. While the target is consumed through the available SMARD API[1], the data set is enriched by weather data provided through the so-called 'Deutscher-Wetter-Dienst' (DWD)[2]. In addition, the data set is expanded with situational information such as timestamps, quarters, and months to construct a model that can forecast as accurately as possible. Further details on the data set construction follow in the next chapter.

[1] https://www.smard.de/app/chart_data/4359/DE/4359_DE_hour

[2] https://dwd.api.bund.dev/

Chapter 2

Data Selection, Preprocessing & Transformation

2.1 Data Selection, Understanding & Gathering

As stated before, the data is consumed from two sources to build the data set. First, the target data is being occupied to forecast the residual load provided by the SMARD API. Timestamps are added to the target values to join the target data with the hourly weather data, which will also work as the index of the data set. Due to this 1:1 assignment of timestamps to the residual load in a time-series manner, a sorted target data set is provided. The target data set begins with a record of 2014-12-31 23:00:00, inheriting a residual load of 35,955.75 kWh, and ends with the record on the date 2023-10-19 13:00:00 having a residual load of 39151.00 kWh. The target data's histogram shows a normal distribution with a mean of around 40,000 kWh.

In addition, the weather data set provided via the DWD will be joined with our target data set. Previously, it was stated that weather data was being taken for four locations: Potsdam, Hannover, Stuttgart, and Munich. Due to many possible weather factors that can be considered influencing, the focus was on four attributes for each location: wind velocity, sun duration, air temperature and precipitation amount. It was determined that these factors and the explanations given by the DWD are the most important factors when looking at the residual load because of their direct influence on the supply of renewable energy, e.g., wind and solar energy. A negative correlation can be observed in all relations by calculating the correlation between the residual load and the weather factors. The strongest correlation of -0.48, respective -0.41, comprises the wind velocity of the northern locations and the residual load. One can expect the observed effect by considering

the energy mix of Germany, which is composed of 30% wind energy. The factors of the remaining wind velocities, sun duration, and air temperature are slightly correlated. The precipitation amount of the two northern weather stations is also slightly correlated, but the southern ones are not correlated.

Consequently, the data within the four categories for each weather station were downloaded, providing each attribute in a .txt file where the attribute types were unclear and had to be formatted initially. For this reason, the timestamps were formatted to the data type, removing non-necessary values and renaming the attributes wanted to receive intuitively, e.g., from ' FF' into 'Wind Velocity for Station + station.id'. The documentation provided by the DWD helped to identify necessary and unnecessary values, which gave information about each column's informative value. After these steps, the target data set was joined with the weather data set to start the preprocessing and transformations.

Regarding the data quality of the target data set, the SMARD API already provided it in a time series sorted way, so only timestamps had to be added. The data quality of weather data has proven to be a more intricate matter. This is due to the need for a comprehensive review of documentation to discern the relevant values, establish their attribute structure, and ultimately integrate them with the target data set. Furthermore, the weather data set contains missing values other than the energy data set.

2.2 Data Preprocessing & Transformation

After the data was selected and prepared, various preprocessing methods were applied to obtain a suitable data set. Initially, all rows containing missing data in at least one column were removed. Then, two additional features were created to include situational information: quarter and month. The timestamp-based index enabled a numerical quarter and month assignment to the data set from 1 - 12 and 1 - 4. Subsequently, a data set of 51.823 rows and 19 columns was obtained.

In addition, the holdout method was applied as a test-train-split without shuffling to keep the time series sorting, using a train-test-split of 80% / 20%. Afterwards, outliers were removed from the training data set. Outliers were identified by observing the scatter plots of each attribute, given their residual load. Examples of outliers were values like -999, which were replaced by Not-a-Number (NaN) values. Because missing values were removed before, too, this step was repeated to remove all rows that were replaced before with NaN to eliminate outlier data. At last, normalization was applied in the form of a *MinMaxScaler* to transform all columns except for the last three of the right-hand side of the table onto a scale of 0 - 1 to make each column more comparable, e.g., wind velocity and sun duration.

Chapter 3

Data Mining Algorithms

3.1 Simple Regression Methods

Linear regression is a widely used model for applying continuous data. All variables were considered as influencing factors when applying linear regression in the report presented. However, the test data set showed a Root Mean Squared Error (RMSE) of 18,152 MWh, indicating that linear regression performed significantly worse than the baseline. The initial application of the multivariate regression approach, which considers all variables, yielded suboptimal outcomes. The Lasso and Ridge regression techniques built on linear regression were utilized to enhance performance.

Hyperparameter optimization through grid search with time series splitting consisting of five splits was used to optimize the Lasso and Ridge models. The grid search produced an optimal alpha parameter of 10 for the Ridge regression, resulting in an RMSE of 12,845 MWh. The result indicates that particular variables did not significantly impact the residual load.

The Lasso regression method was employed to enhance the model's accuracy further. Fine-tuning the Lasso regression model with an alpha value of 2.7 significantly improved compared to the Ridge regression, with a root mean square error (RMSE) of 10,150 MWh. The efficacy of the Lasso regression method in weakly influencing parameters, thereby rendering specific parameters irrelevant to the residual load, was underscored by this regression.

In summary, linear regression may not be effective when using basic regression techniques due to the presence of non-influential variables. However, the Ridge and Lasso regression methods can be implemented to mitigate the impact of these variables, resulting in lower RMSE than that of both linear regression and the baseline.

3.2 k Nearest Neighbour

Additionally, we used *KNN-Regression* on our Data, as a fist Baseline we ran with all Parameters and optimized for k resulting in a k of 38 and a RMSE of 10831. As we saw in the Simple Regression Models that some variables are irrelevant to the residual load, we tried to focus on specific column sets of the Data. Predicting on Data from just one Weather Feature (e.g. Wind Data) did not provide improved results. The best 4 column Combination overall consisted of Air Temperature for Station 04928 and Wind Velocity of the other stations, optimizing k in this constellation resulted in a k of 400. With RMSE in Cross-Validation not dropping below 10500 in the best results, these combinations did not result in visibly better results than the Knn-Baseline.

3.3 Decision Tree & Random Forest

We experimented with Decision Tree and Random Forest regressors using sklearn's implementations. For hyperparameter tuning, we explored *GridSearchCV* and *RandomizedSearchCV*.

Initially, we set baseline regressors to understand the performance of each model. With *GridSearchCV*, narrowing down parameter ranges was crucial due to computational constraints. However, combining all parameters did not consistently improve RMSE, hinting at dependencies between the individual parameters, affecting predictive capability. *RandomizedSearchCV* showed promising results by providing narrower ranges and better RMSE scores. After all, using *GridSearchCV* did not match *RandomizedSearchCV*'s performance.

3.3.1 Using TimeSeriesSplit

After a while, we realized that *K-Fold cross-validation* might not adequately address the nuances of time series data. Consequently, we implemented *TimeSeriesSplit*, aiming to optimize *RandomizedSearchCV* within this context [1].

Since we are dealing with time series data, we have to remember that regular cross-validation will mix up the order of our data points - which is critical when working with time series data, where the order of occurrence is significant for the model to abstract correctly. *TimeSeriesSplit* will keep the order of the provided data. For a given number of splits, say ten splits, it will use the first 10% of data in the first iteration, the first 20% of data in the second iteration, and so on. Out of the data considered, a small portion "at the end", e.g., the last 20% of the current CV iteration, is used for validation.

Eventually, it proved that the results achieved with *TimeSeriesSplit* were the most promising we could achieve for Random Forest Regression, yielding a **RMSE of 9870 / normalized RMSE of 0.146**. Both K-Fold CV with RandomForest (RMSE approx. 10,500) and all Decision Tree approaches (RMSE approx. 11,000) did not yield better results.

The best Decision Tree was found using the following hyperparameters:

```
'ccp_alpha': 0.7,
'max_depth_': 13.0,
'max_features': 18.0,
'max_leaf_nodes': 35.0,
'min_impurity_decrease': 0.9,
'min_samples_leaf': 0.012,
'min_samples_split': 0.01,
'min_weight_fraction_leaf': 0.0
```

and resulted in the tree shown in Figure Figure 3.1.

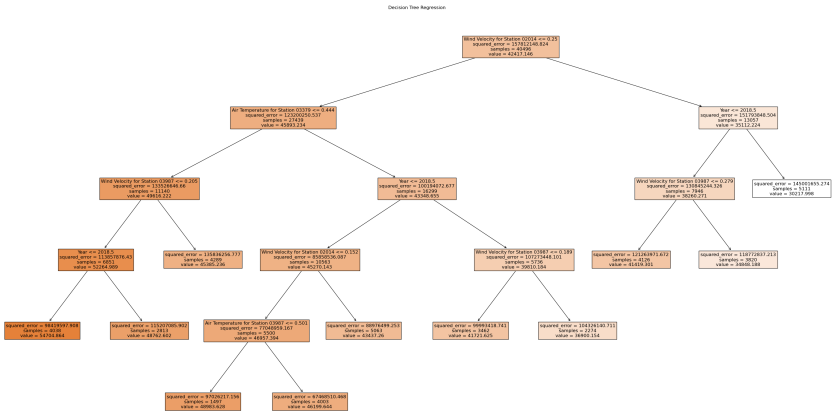

Figure 3.1: Best Computed Decision Tree

However, for the best computed Random Forest, the following hyperparameters applied (using RandomizedSearch with five TimeSeriesSplits):

```
'n_estimators': 283,
'min_samples_split': 7,
'min_samples_leaf': 2,
'max_samples': 0.714,
'max_leaf_nodes': None,
```

```
6     'max_features': 'log2',
7     'max_depth': 75
```

3.3.2 Identified Obstacles

Throughout the project, several obstacles surfaced, impacting the efficiency and effectiveness of our approach:

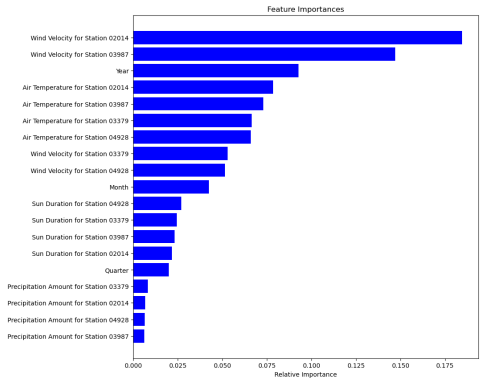

Figure 3.2: Bar Graph for Feature Importances

- **Influence of Iterations:** The number of iterations, particularly noticeable in *RandomizedSearchCV*, exerted a significant influence on results. Higher iterations often yielded better outcomes, yet causing substantial computational cost.
- **Random Seed Impact:** The choice of initial random seed wielded substantial influence, occasionally resulting in exceptionally favourable outcomes during randomized searches. This element of chance, while sometimes beneficial, led to unpredictability in the model's performance.
- **Computational Expense in GridSearchCV:** Searching through an extensive range of hyperparameters using *GridSearchCV* proved to be excessively computationally demanding, especially considering the exhaustive exploration of parameter spaces.
- **Feature Importances:** As can be seen in Figure Figure 3.2 not all features yield a significant importance for the training of our models, which lead to inaccurate results and high RMSEs. This is due to some features being unimportant and therefore having no direct influence on the prediction.

Chapter 4

Evaluation of Results & Limitations

In this chapter, we will present the results of our regression efforts. As mentioned initially, our measure for comparing the individual approaches is the *Root Mean Squared Error* (RMSE). The individual results of our applied approaches are depicted in Figure 4.1.

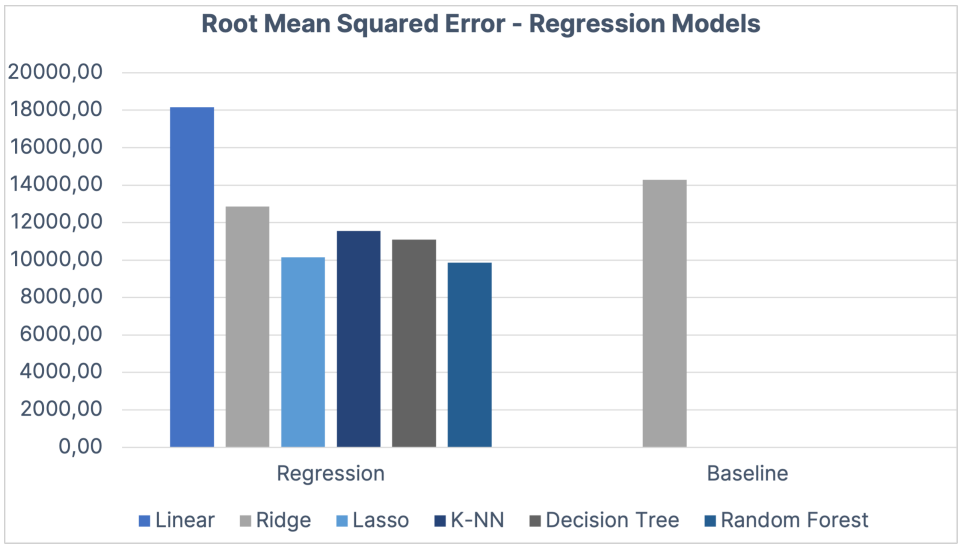

Figure 4.1: Applied Regression Approaches by RMSE

When comparing linear regression to our baseline (the mean value of all data

points), linear regression performs worse than the baseline. One of our experiments' most significant findings is that simple algorithms like Lasso Regression perform equally well as more sophisticated methods like Decision Trees or Random Forests. Given that Random Forests tend to overfit the provided data, we consider the Lasso Regression approach the most promising regression approach for our unique problem.

However, a lasso regression approach would only yield interpolating results, which probably is not what we strive for when aiming to predict the future share of renewable energies. Since renewable energies tend to grow in Germany, using an extrapolating regression method makes sense to allow meaningful outlooks into the future.

Let's dive into the quality of the results. The best regressor used gave an RMSE of slightly under 10,000 MWh. This is an improvement from the initial baseline RMSE of around 15,000 MWh, which was a promising start. However, when considering the scale of the target variable, with a mean value of roughly 42,500 MWh, it must be acknowledged that the results are off by about 23%.

This discrepancy is significant, especially considering that the goal is to predict the total consumption of renewable energies in Germany. A 23% difference translates into a significant variance in actual energy usage. Initially, it was believed that wind velocity, air temperature, sun duration, and precipitation amount would considerably impact the residual load. However, given the observed deviation, it must be conceded that other factors are also at play. For instance, situational data could affect the target variable.

Given the limitations of the findings, relying on them to make decisions related to energy demand planning would be unwise. It is essential to consider additional factors and conduct more research before making critical decisions.

The python code of this project can be found in this GitHub Repository[1].

[1] https://github.com/maxlautenbach/data-quadrat.git

Bibliography

[1] sklearn.model_selection.TimeSeriesSplit. https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.TimeSeriesSplit.html. [Accessed 04-12-2023].

[2] Bundesnetzagentur. Bundesnetzagentur - Presse - Bundesnetzagentur legt Regelungen zur Integration steuerbarer Verbrauchseinrichtungen fest — bundesnetzagentur.de. https://www.bundesnetzagentur.de/SharedDocs/Pressemitteilungen/DE/2023/20231127_14a.html, 2023. [Accessed 05-12-2023].

[3] Dr. Tim Höfer, Julius Ecke, Christoph Pfister, and Miltiadis Zervas. Marktdesign für einen sicheren, wirtschaftlichen und dekarbonisierten strommarkt - studie. https://gas.info/fileadmin/Public/PDF-Download/studie-marktdesign-strommarkt-zukunft-gas-enervis.pdf, enervis energy advisors GmbH, Berlin, 2022. [Accessed 05-12-2023].