Mendy Hsu

Capstone - Project Report

Forecasting Energy Load with Seasonal ARIMA

Problem statements

Energy suppliers/companies need to ensure the electricity supply always meets their customers' demands (reliability of supply) and prevents the supply shortage. As the energy consumptions depend on the time, the day, seasons, and even weather, etc. Companies must estimate/forecast how much electricity to generate, buy from, the price to sell in advance, and whether to invest in various electrical infrastructure.

This project aims to build a **time series forecasting model** that can **predict the electrical consumption of the next day**, given the information about the past electricity loads. Here, we focus on the four years of data in Spain. The final goal is to have a model forecasting the range (minimal and maximum) of electricity load from day to day.

1. Data Sources

Hourly energy demand generation and weather (6.3 MB)

File descriptions - energy_dataset.csv (35064 rows, 29 columns)

- Period and location: **four years (2015 - 2018)** recording in Spain.
- Number of fields: **29** (electrical consumption, generation, and pricing, etc.)
- Number of rows: **35064 hourly records.**
- It has features/columns:
    - A **Time** column tracked the datetime of when the rest feature values were gathered.
    - **Forecasted estimates**: energy generated by solar/wind (3 columns),
      total load, and price (2 columns).
    - **Actual recordings**: energy generated by different sources (21 columns),
      total load, and price (2 columns).

Take-away

- We aim to train a forecasting model that predicts total load actual (dependent/ response variable) for a given time (independent/input variable).

- Note that total load actual contains **Nan**.

2. Data Cleaning and Wrangling

time: converted to *pd.datetime* object.

total load actual: **forward filling in Nan** with linear interpolation.

dayMax, dayMin: extract the daily maximal (red line) and minimum (blue) total loads (green, Fig. 1).

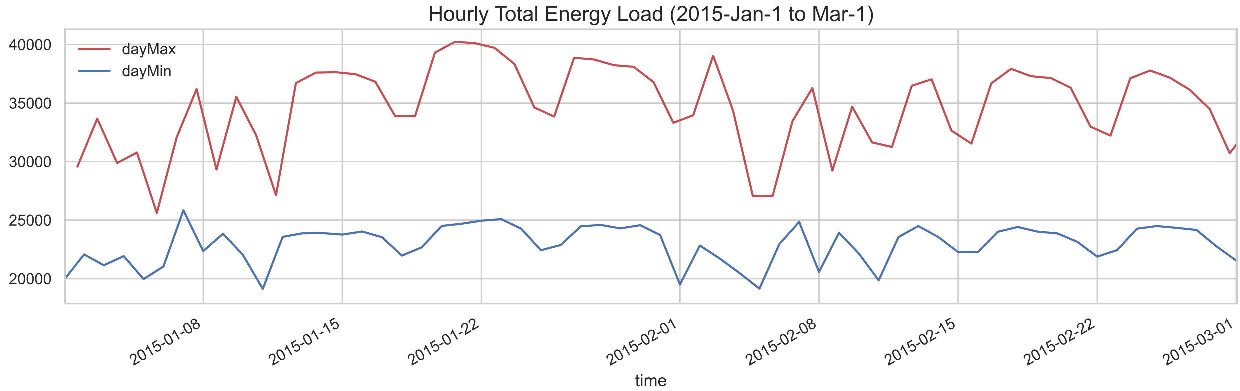


Figure 1: Hourly total load over the 3 months (green line). The red (blue) line shows the maximum (minimum) load in a day.

3. Problem Formulation

Our initial goal was to predict the *total load* hourly in 2018 based on 2015 – 2017 records.

Issue: With **Seasonal ARIMA** model, it takes much longer time to train hourly points over 3 years.

→ Option #1: train the model for sort-term period (ex: train data of 2 to 6 months; predict the next month).

→ Option #2: train the model of daily load (down sampled the time series).
This reduced the number of points by 1/24.

We shifted to the **alternative goals**:

1. Predicting daily **minimum** and **maximum** *total load*. → **Seasonal ARIMA (this is our focus).**
2. The typical pattern of *hourly total loads* each day. → **Time series decomposition**

4. Training Data Development

Split Train-test Sets - We split the data into a **training (~75%)** and a **test (~25%)** set, and pick the best model based on the one with minimal AIC score.

- **Train** (~75%): 2015-1-1 to 2017-12-31
- **Test** (~25%): 2018-1-1 to 2018-12-31
- **Evaluation score**: Akaike Information Criterion (AIC).

Potential Improvement – Applying the **rolling cross validation** on training set help find model parameters that are generalized to unseen data and avoid overfitting. This can be achieved via `pmdarima` API:

```
Pmdarima.model_selection.RollingForecastCV(...)
```

5. Exploratory Data Analysis

5.1. Rolling Average Over Different Time Windows

Let’s analyze hourly data by aggregating over different time windows: **day**, **week**, and **month**, to get ideas of the trend and spread of observations.

Daily average total load (24 hours) shows weekly oscillations (Figure 2). There were higher loads during weekdays (hills) than weekends (dips). Also, there were peaks appear at the beginning of each month.

The range of daily load also shows the weekly pattern (Figure 3, Figure 4). **The differences of maximum and minimum in a day** (*dayMax* – *dayMin*, Figure 4) were higher during weekdays than the weekends. Again, there was a spike at the beginning of every month.

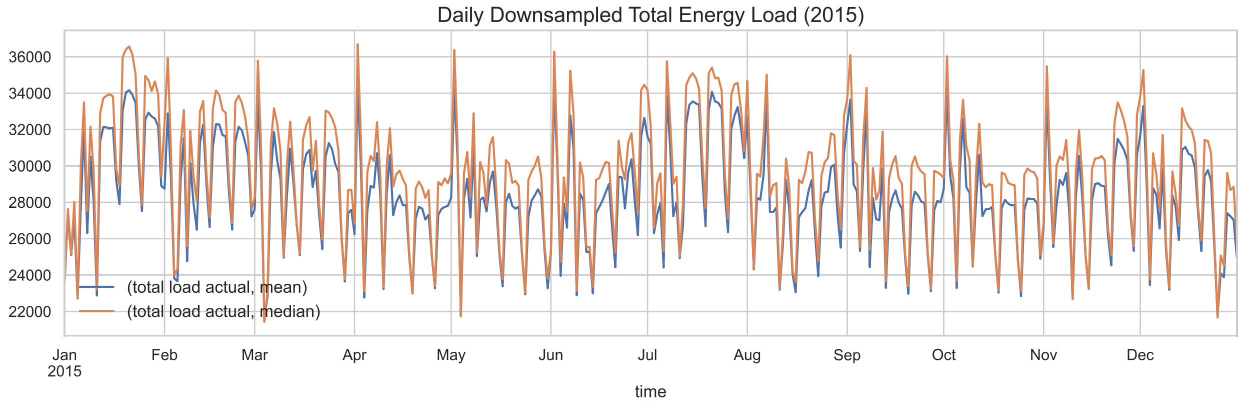


Figure 2: Daily average total load.

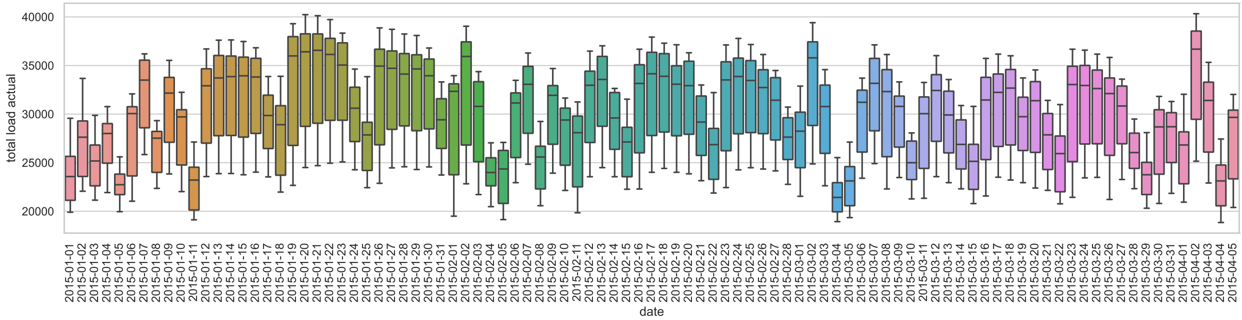


Figure 3: The range of daily load. The ranges (boxplot) during the weekdays were similar in size and broader than those during the weekends.

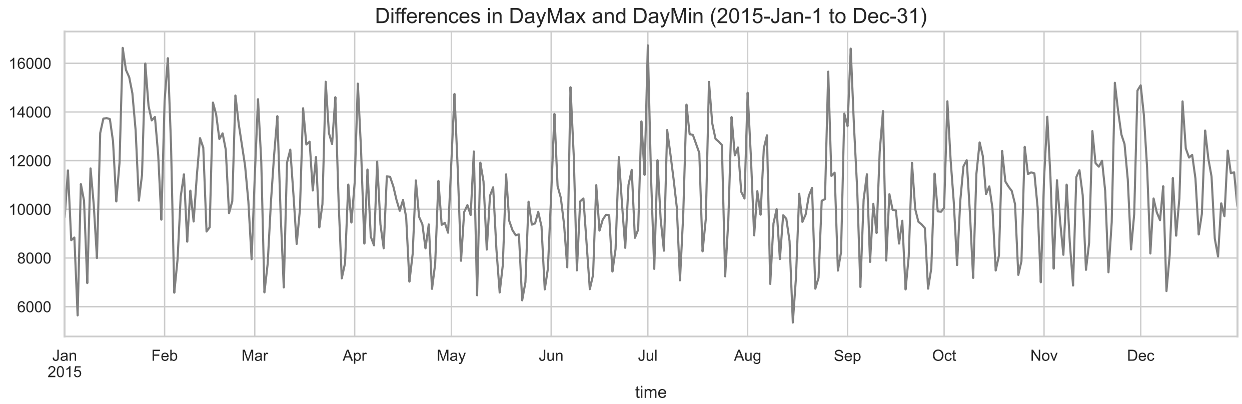


Figure 4: The range of daily load.

Weekly average total load (7 days) shows monthly oscillations with changing amplitude (Figure 5). There were higher loads during weekdays (hills) than weekends (dips). Also, there were peaks appear at the beginning of each month.

Monthly average total load shows weekly oscillations (Figure 6). There were higher loads during weekdays (hills) than weekends (dips). Also, there were peaks appear at the beginning of each month.

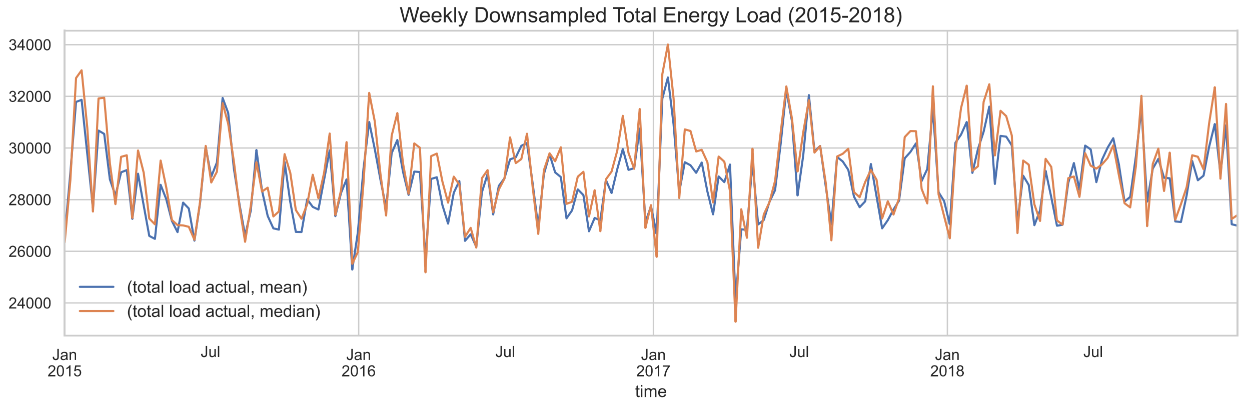


Figure 5: Weekly average total load.

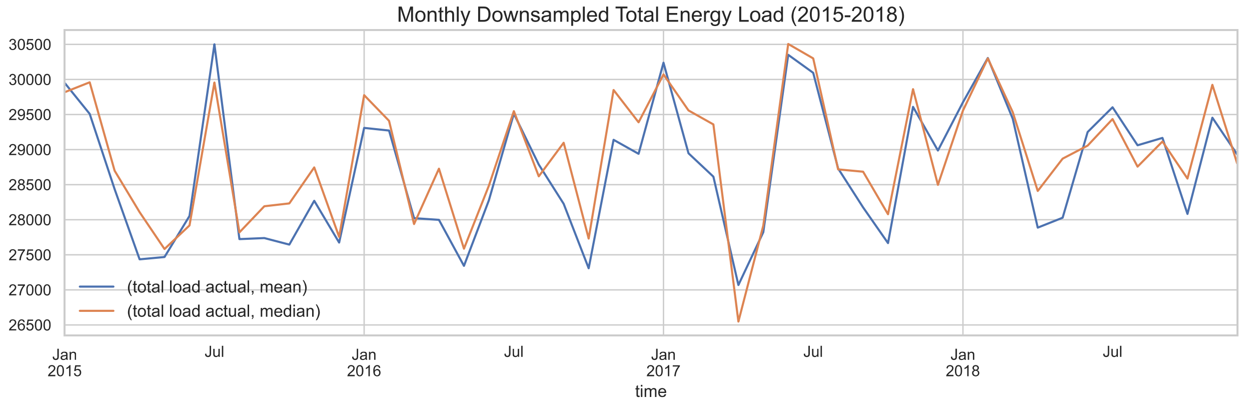


Figure 6: Monthly average total load.

5.2. Seasonal Decomposition ($y_t = T_t \times S_t \times R_t$)

We used seasonal decomposition to find the **repeatedly pattern within a day** (the third panel, seasonal trend).

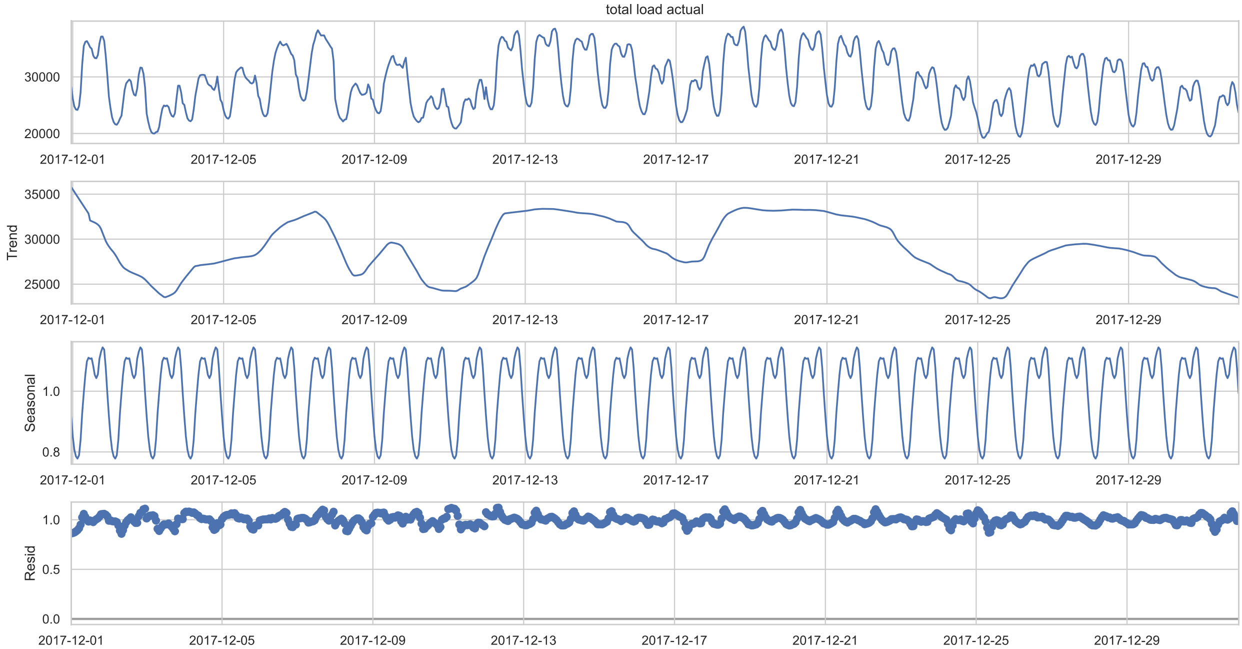


Figure 7: Multiplicative seasonal decomposition for dayMax. The original time series is the multiplication of Trend (T), Seasonal (S) and Resid (R). The Seasonal panel shows the repeated daily patterns (every 24 hours).

5.3. Stationarity Tests: Augmented Dickey-Fuller (ADF) and Kwiatkowski–Phillips–Schmidt–Shin (KPSS)

Both ADF and KPSS are statistical tests to check whether the time series stationary (Ref. 2).

	ADF (H_0: Series is non-stationary, or series has a unit root. H_A: Series is **stationary,** or series has no unit root.)	KPSS (H_0: Series is trend **stationary**, or series has no unit root. H_A: Series is stationary, or series has no unit root.)	**Stationarity**
dayMax	Test Statistic: -6.44 p-value: 1.61e-08 → H_A	Test Statistic: 0.228 p-value: 0.10 → H_0	True
dayMax with 1 step differencing	Test Statistic: -14.88 p-value:1.60e-27 → H_A	Test Statistic: 0.044 p-value: 0.10 → H_0	True
dayMin	Test Statistic: -6.16 p-value: 7.37e-08 → H_A	Test Statistic: 0.735 p-value: 0.01 → H_A	ADF indicates stationarity, and KPSS indicates non-stationarity → **The series is difference stationary.**
dayMax with 1 step differencing	Test Statistic: -12.06 p-value: 2.489e-22 → H_A	Test Statistic: 0.071 p-value: 0.10 → H_0	True

6. Modeling - Seasonal ARIMA (SARIMAX)

6.1.SARIMAX Hyperparameters

SARIMAX stands for

- ARIMA - **A**uto-**R**egressive **I**ntegrated **M**oving **A**verage
- X – e**X**ogenous factors

Hyperparameters: **non-seasonal orders (p, d, q)** and **seasonal orders (P, D, Q, s)**

- P, p – autoregressive
- D, d – differencing order
- Q, q – moving average
- s – number of steps for a season

Hyperparameter for non-seasonal p, d, q

- Differencing order (d) – the time series is **stationary** after we applied d times differencing with a normal step of 1.
- AR(p)-MA(q) based on two plots:
  - **A**uto**C**orrelation **F**unction (ACF)
  - **P**artial **A**uto**C**orrelation **F**unction (PACF)

	AR (p)	**MA(q)**	**ARMA(p, q)**
ACF	Tail off	Cuts off after **lag q**	Tail off
PACF	Cuts off after **lag p**	Tail off	Tail off

Hyperparameter for seasonal P, D, Q

- Seasonal differencing order (D) – the time series is **stationary** after we applied D times differencing with seasonal steps (s).
- AR(P)-MA(Q) based on two plots:
  - **A**uto**C**orrelation **F**unction (ACF)
  - **P**artial **A**uto**C**orrelation **F**unction (PACF)

	AR (P)	**MA(Q)**	**ARMA(P, Q)**
ACF	Tail off	Cuts off after **lag (Q*s)**	Tail off
PACF	Cuts off after **lag (P*s)**	Tail off	Tail off

6.2. Hyperparameter search: seasonal steps s = 7

ACF plot of ***dayMax*** shows the periodically **peaks** for **every 7 days**.

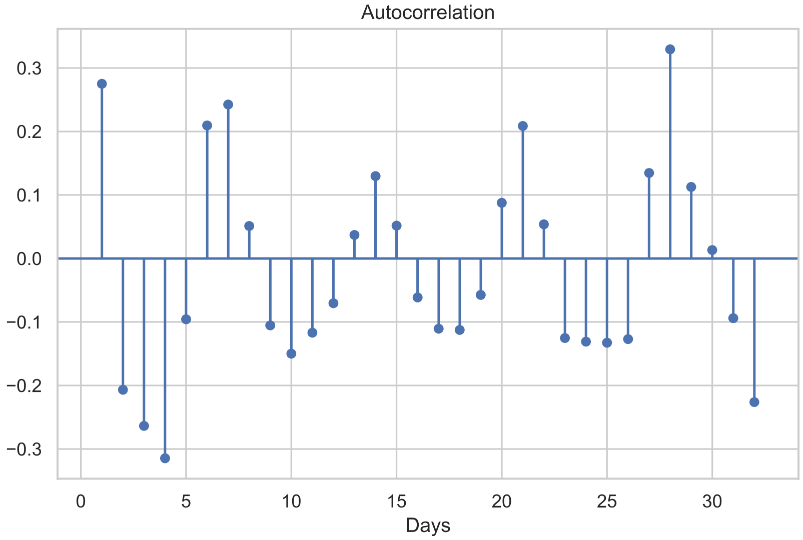
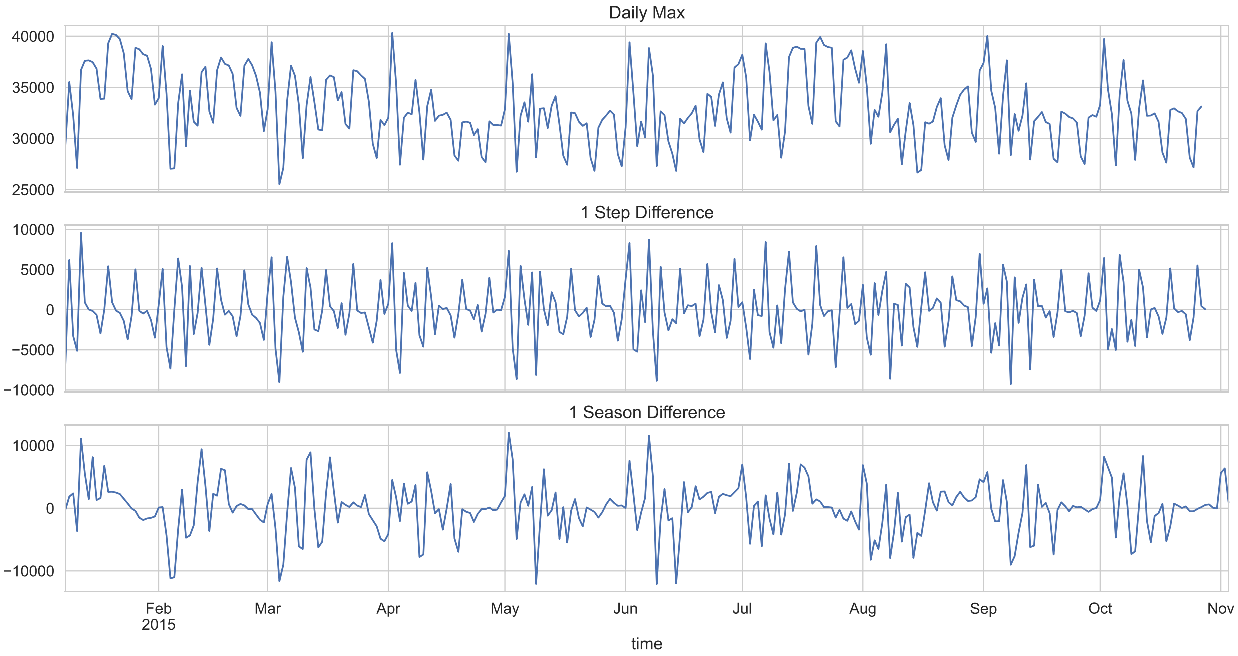


Figure 8: ACF plots for dayMax.

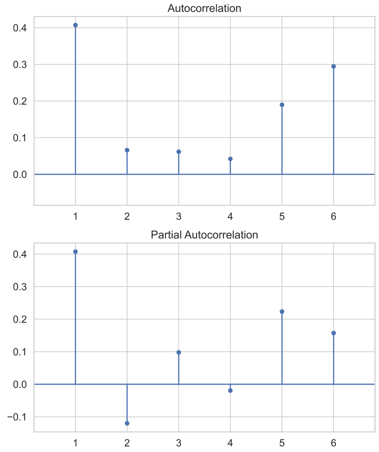
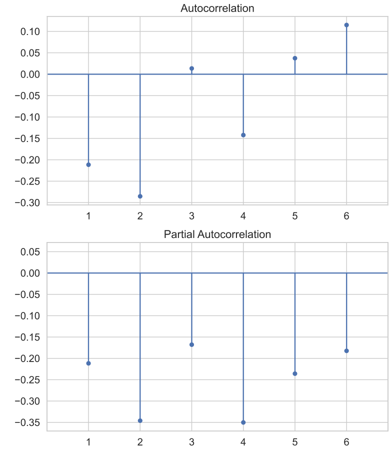
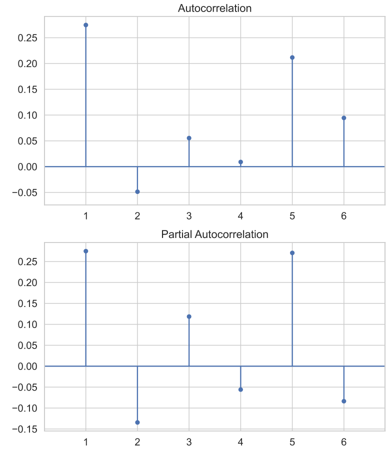
6.3. Hyperparameter search: differencing order (d, D)

Here we take a one-year **dayMax** as examples to show how the time series transforms after applying the normal (d = 1) and seasonal differencing (D = 1). The goal of differencing is to make the statistics of the time series stationary. That is, there should be no significant changes in mean $\langle x_t \rangle$ and standard deviation $\mathrm{Var}(x_t)$ between different time windows.



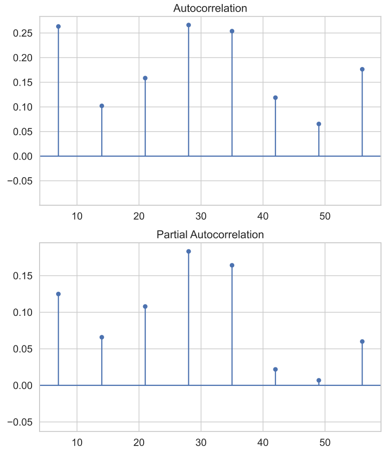
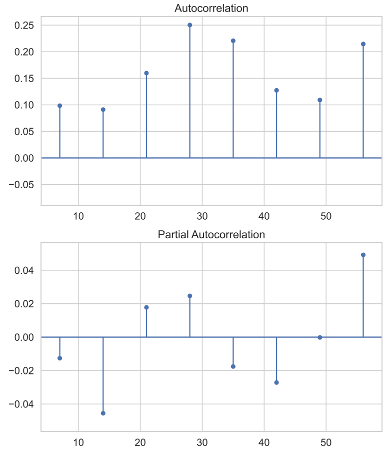
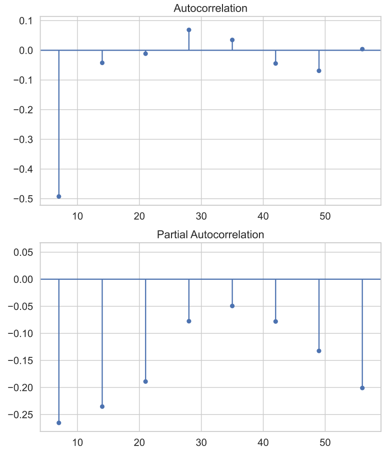
6.4. ACF and PACF for non-seasonal (p, q)

The **dayMax** cannot be modeled neither the pure AR nor MA model as both ACF and DACF tailed-off. Instead, **it requires both parameters (p, q), the ARMA model.**

(d, D) = (0, 0)	(d, D) = (1, 0)	(d, D) = (0, 1)
 Autocorrelation Partial Autocorrelation 	 Autocorrelation Partial Autocorrelation 	 Autocorrelation Partial Autocorrelation

6.5. ACF and PACF for seasonal (P, Q)

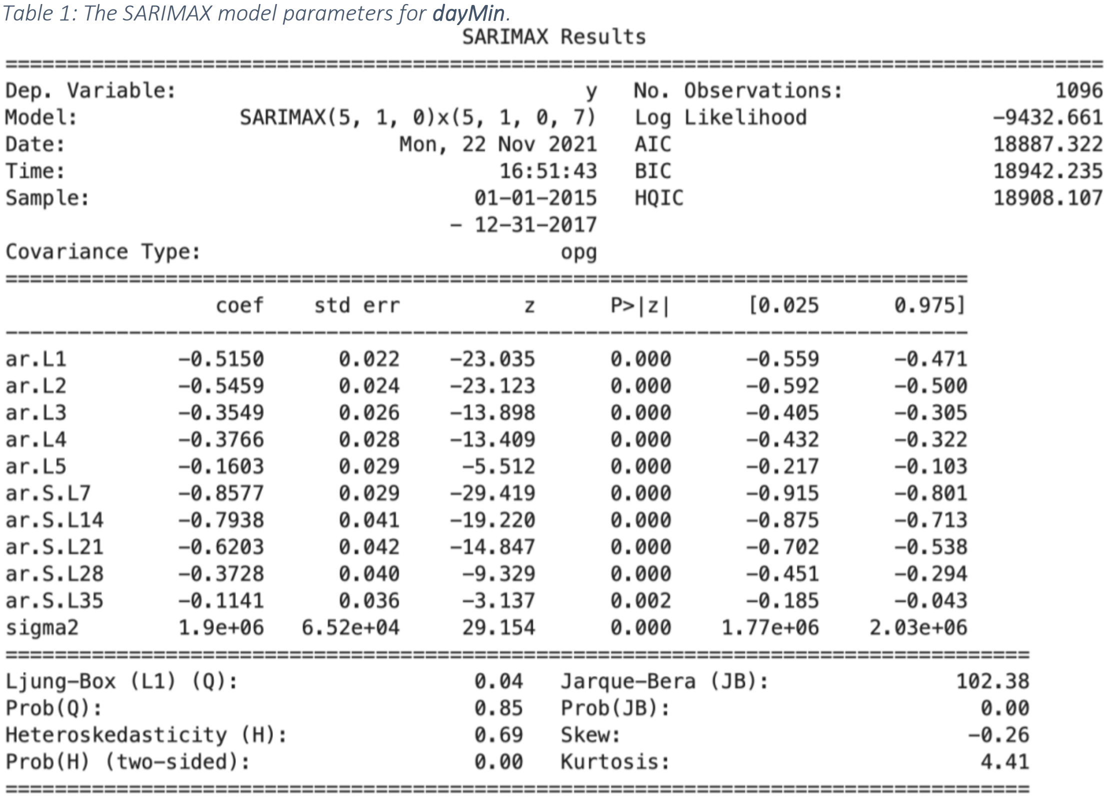
Again, the **dayMax** cannot be modeled neither the pure seasonal AR nor MA model as both ACF and DACF tailed-off. We need **both parameters (P, Q) to model seasonal ARMA.**

(d, D) = (0, 0)	(d, D) = (1, 0)	(d, D) = (0, 1)
 Autocorrelation Partial Autocorrelation 	 Autocorrelation Partial Autocorrelation 	 Autocorrelation Partial Autocorrelation

6.6.Hyperparameter searching using Auto-ARIMA

- Model evaluation score: **Akaike Information Criterion (AIC)**
  18887 for **dayMax**; 20430 for **dayMax**.
- Best seasonal ARIMA parameters:
  (p, d, q) = (5, 1, 0) and (P, D, Q, s) = (5, 1, 0, 7) for both the **dayMin** and **dayMax**.

*Table 1: The SARIMAX model parameters for **dayMin**.*



```
                                     SARIMAX Results                                     
=========================================================================================
Dep. Variable:                                 y   No. Observations:                 1096
Model:             SARIMAX(5, 1, 0)x(5, 1, 0, 7)   Log Likelihood               -9432.661
Date:                           Mon, 22 Nov 2021   AIC                          18887.322
Time:                                   16:51:43   BIC                          18942.235
Sample:                               01-01-2015   HQIC                         18908.107
                                    - 12-31-2017                                         
Covariance Type:                             opg                                         
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1         -0.5150      0.022    -23.035      0.000      -0.559      -0.471
ar.L2         -0.5459      0.024    -23.123      0.000      -0.592      -0.500
ar.L3         -0.3549      0.026    -13.898      0.000      -0.405      -0.305
ar.L4         -0.3766      0.028    -13.409      0.000      -0.432      -0.322
ar.L5         -0.1603      0.029     -5.512      0.000      -0.217      -0.103
ar.S.L7       -0.8577      0.029    -29.419      0.000      -0.915      -0.801
ar.S.L14      -0.7938      0.041    -19.220      0.000      -0.875      -0.713
ar.S.L21      -0.6203      0.042    -14.847      0.000      -0.702      -0.538
ar.S.L28      -0.3728      0.040     -9.329      0.000      -0.451      -0.294
ar.S.L35      -0.1141      0.036     -3.137      0.002      -0.185      -0.043
sigma2         1.9e+06   6.52e+04     29.154      0.000    1.77e+06    2.03e+06
===================================================================================
Ljung-Box (L1) (Q):                   0.04   Jarque-Bera (JB):               102.38
Prob(Q):                              0.85   Prob(JB):                         0.00
Heteroskedasticity (H):               0.69   Skew:                            -0.26
Prob(H) (two-sided):                  0.00   Kurtosis:                         4.41
===================================================================================
```

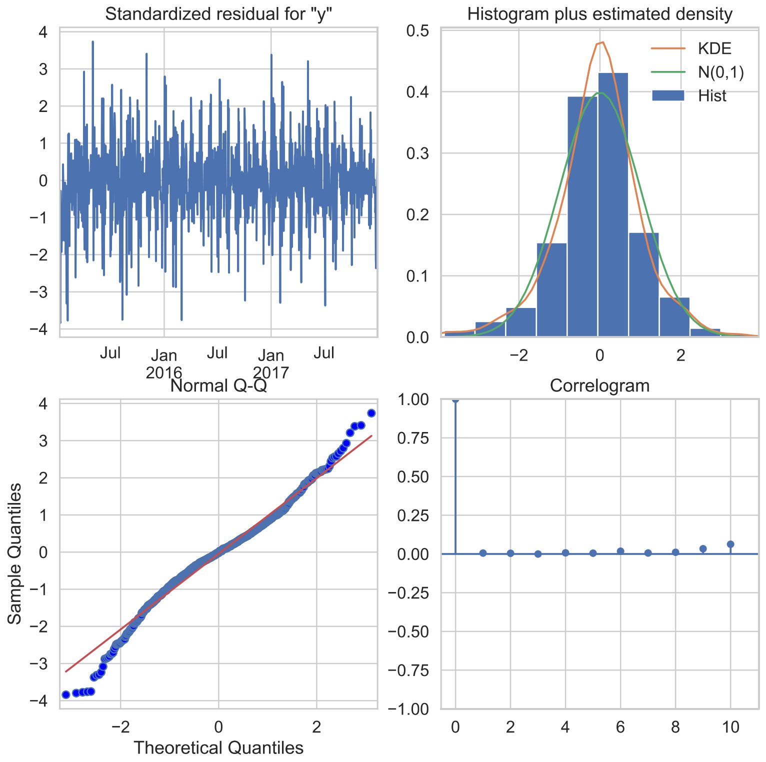
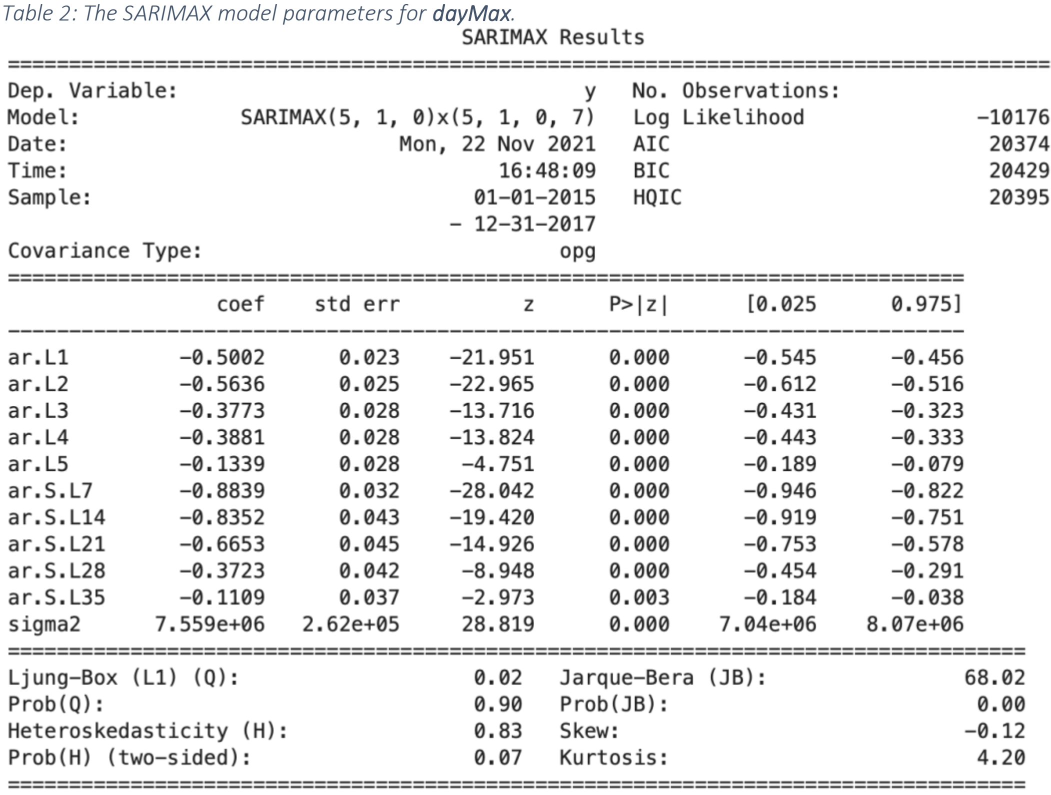


*Figure 9: The residual analysis for **dayMin**. The residual analysis for **dayMax**. The residual distribution is nearly normal distribution (upper right). The auto correlation of residual is ignorable or small (t=10) (ACF plot on the lower right panel).*

*Table 2: The SARIMAX model parameters for **dayMax**.*



```
                                      SARIMAX Results
==========================================================================================
Dep. Variable:                                  y   No. Observations:                 1096
Model:             SARIMAX(5, 1, 0)x(5, 1, 0, 7)   Log Likelihood              -10176.442
Date:                            Mon, 22 Nov 2021   AIC                          20374.884
Time:                                    16:48:09   BIC                          20429.797
Sample:                                01-01-2015   HQIC                         20395.669
                                     - 12-31-2017
Covariance Type:                              opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1         -0.5002      0.023    -21.951      0.000      -0.545      -0.456
ar.L2         -0.5636      0.025    -22.965      0.000      -0.612      -0.516
ar.L3         -0.3773      0.028    -13.716      0.000      -0.431      -0.323
ar.L4         -0.3881      0.028    -13.824      0.000      -0.443      -0.333
ar.L5         -0.1339      0.028     -4.751      0.000      -0.189      -0.079
ar.S.L7       -0.8839      0.032    -28.042      0.000      -0.946      -0.822
ar.S.L14      -0.8352      0.043    -19.420      0.000      -0.919      -0.751
ar.S.L21      -0.6653      0.045    -14.926      0.000      -0.753      -0.578
ar.S.L28      -0.3723      0.042     -8.948      0.000      -0.454      -0.291
ar.S.L35      -0.1109      0.037     -2.973      0.003      -0.184      -0.038
sigma2      7.559e+06   2.62e+05     28.819      0.000    7.04e+06    8.07e+06
====================================================================================
Ljung-Box (L1) (Q):                   0.02   Jarque-Bera (JB):                68.02
Prob(Q):                              0.90   Prob(JB):                         0.00
Heteroskedasticity (H):               0.83   Skew:                            -0.12
Prob(H) (two-sided):                  0.07   Kurtosis:                         4.20
====================================================================================
```

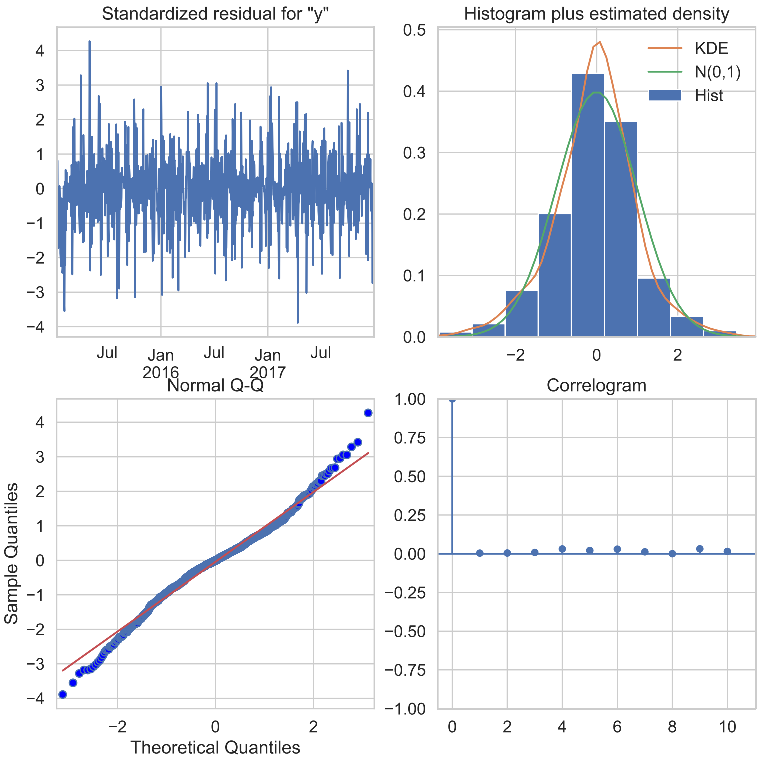


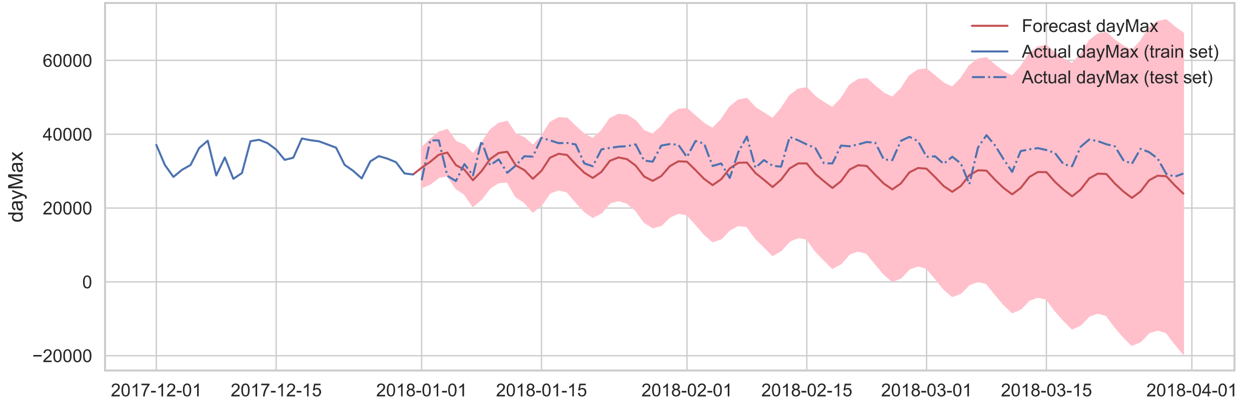
*Figure 10: The residual analysis for **dayMax**. The residual distribution is nearly normal distribution (upper right). There is no significant auto correlation in the residual (t>0, ACF plot on the lower right panel).*

7. Model Prediction and Evaluation

We will fist show <u>the long-term forecasted</u> **dayMax** when we do not update the trained model with the new observations. Then, we show the predictions when the trained model was updated with the new points up to the day before forecasted.

7.1. Forecasting (without updating the new point)

The predicted result sits around the actual ***dayMax*** for **about two weeks** and starts to deviate from actual one. Yet, the prediction still **captures the weekly oscillation** in long-term.



7.2. Forecasting (daily updating the new point)

If we updated the new observation a day before forecast, the predicted result goes around the actual **dayMax** and **dayMin**. However, the prediction cannot predict the **spike at the beginning of each month**.

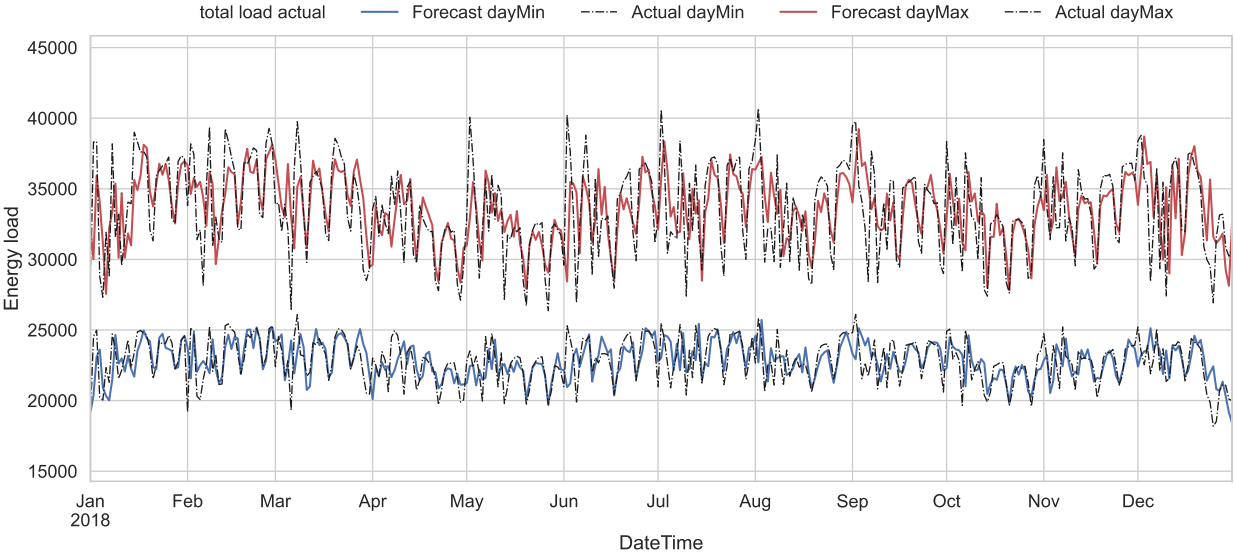


*Figure 11: The day-by-day forecast **dayMax** (red line) and **dayMin** (blue line). The shaded area are the confidence intervals. The gray line is the hourly load time series.*

7.3. Model Evaluation on Test Sets

$$\text{MAPE}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} \frac{|y_i - \hat{y}_i|}{max(\epsilon, |y_i|)}$$

	dayMin	dayMax
R-square (test set)	0.22	0.310
MSE (test set)	1884188	7084441
MAE (test set)	985.7	1987.4
MAPE (test set)	1.997	2.547
Correlation (test set, prediction) p-value	0.547 0	0.577 0

7.4. Residual Analysis on Test Set

The residual plots and their autocorrelation for predicted dayMax, and dayMin. There are rooms to improve the model as it did not fully capture some of the significant time correlation components.

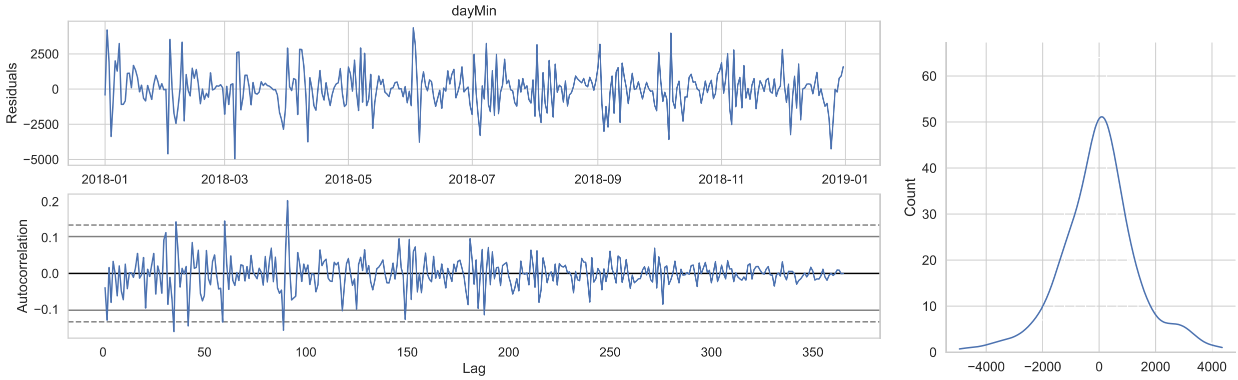


*Figure 12: Residual time series for **dayMin** (upper left panel) and its auto correlation (lower left).* The horizontal lines in the plot correspond to 95% (solid) and 99% (dashed) confidence bands. *The residual distribution (right) shows some higher moments (long tails on both side).*

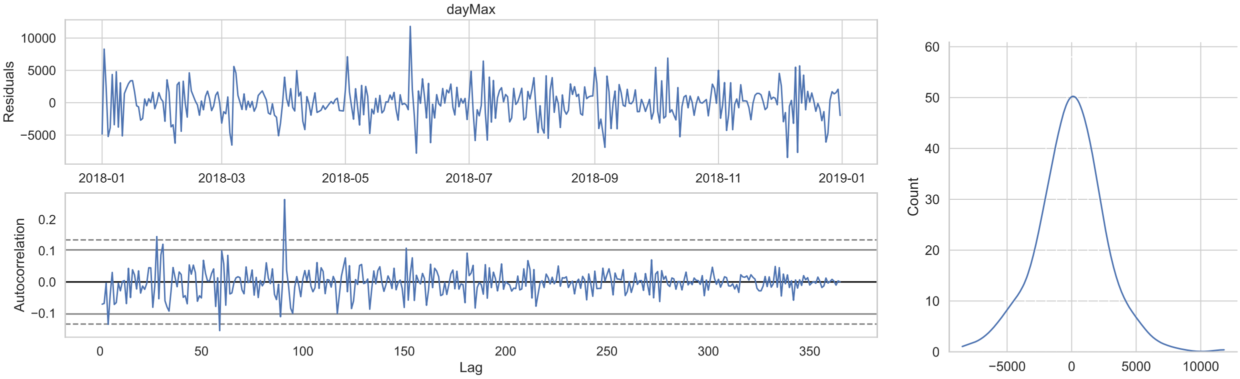


*Figure 13: Residual time series for **dayMax** (upper left panel) and its auto correlation (lower left).).* The horizontal lines in the plot correspond to 95% (solid) and 99% (dashed) confidence bands. *The residual distribution (right) shows some higher moments (long tail on right side).*

8. Conclusion

- **Seasonal ARIMAX** model for forecasting daily **min/max** energy load.
- Auto ARIMA for hyperparameter tunning.
  The best set of order parameters are **(p, d, q) = (5, 1, 0)**, and **(P, D, Q, s) = (5, 1, 0, 7)**.
- Model Evaluation:

  MAPE (test set) = 2 (dayMin) and 2.55 (dayMax).

  Residual analysis on the test set shows the current model did not capture some auto-correlation components (3 peaks from the autocorrelation plots). This uncaptured feature could be related to the **higher peak load** happening at the beginning of each month (Figure 1).

9. Future Work

- Combining the predicted daily Min/Max and hourly load pattern of a day (decomposition).
- Other models for time series forecasting:
  LSTM (Long-Short Term Memory); **NNETAR** (Neural NETwork AutoRegression).

10. Reference:

1. https://otexts.com/fpp2/
2. https://www.analyticsvidhya.com/blog/2021/06/statistical-tests-to-check-stationarity-in-time-series-part-1/
3. https://towardsdatascience.com/forecasting-energy-consumption-using-neural-networks-xgboost-2032b6e6f7e2
4. https://www.sciencedirect.com/science/article/pii/S1474667016319516