Thermal and New-Energy Electricity Generation Time Series Analysis

December 2022

STAT4601 Time Series Analysis
Department of Statistics & Actuarial Science
Faculty of Science
the University of Hong Kong

FAN Zixian u3577161@connect.hku.hk
LI **
LU **
SUI **
ZHU **

Course Co-ordinator Prof G Li

Contents

1 Introduction

Thermal power, which utilizes the energy produced from burning coal, liquid natural gas and oil, has been the most fundamental source of electricity generation ever since the Industrial Revolution. However, thermal power generation has received more and more criticism for its severe side effects, for example, causing air pollution and accelerating global warming. In contrast, renewable energy, including solar energy, wind energy, hydroelectric power and other non-carbon sources, has experienced an increasing attention for its higher sustainability during the past few decades. [1] The transition between two energy sources is taking place. The graph below illustrates the climbing and then declining trend in the amount of electricity generated by thermal power, as well as the constant growth in the electricity generated by new energy.

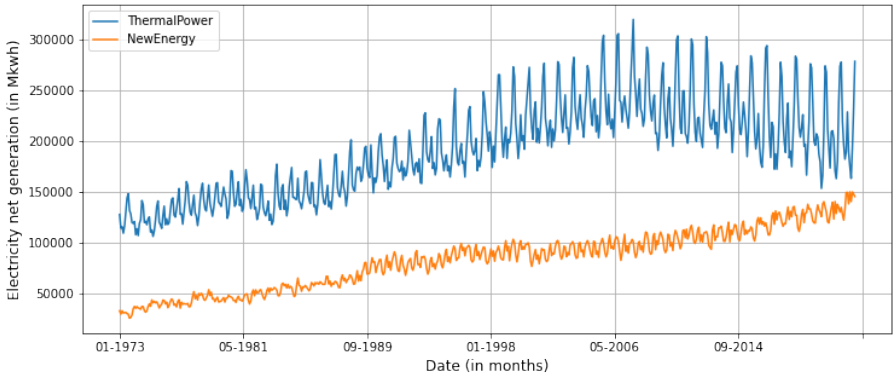

Figure 1.1: Electricity net generation by thermal power and new energy

This report aims to study the changes in electricity generation by thermal power and new energy sources in the U.S. in the last half-century by time series. The data analyzed consisted of 588 monthly electricity generation data from 1973 to 2021 for both sources, obtained from the U.S. Energy Information Administration website. The first 595 data will be used for model-building, and the most recent seven month's data will be conserved as the true value for prediction evaluation.

In the following three sections, we will first study thermal power generation by exploring data stationarity, applying the standard Box-Jenkins model-building strategy, and then predicting future values using the optimal model. New energy generation data will be analyzed with the same methodology. Furthermore, we will extend our study by employing different modelling methods. In section five, we adopt a machine learning approach for more effective predictions, and in section six, a GARCH model is built to handle the heteroskedastic problem in time series. Section seven concludes the report by summarising and comparing the investigated models.

2 Data Stationarity

2.1 Thermal Power Generation

We examine the data stationarity by its time series plot. Figure 2.1 shows an upward trend on the series mean value and variance value, indicating the need for both log transformation and differencing to achieve a stationary time series data.

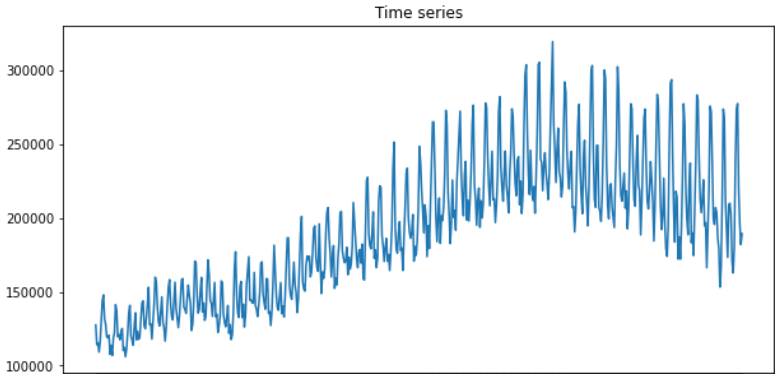

Figure 2.1: Electricity net generation by thermal power and new energy

Firstly, log transformation is performed on the data to adjust the increasing variance. The data is logged up to four times, and the corresponding p-value after each log transformation is recorded in Table 2.1. The first log is selected as it results in the largest decline in the p-value. After taking the log, the times series plot shows a more steady variance (Figure 2.2).

Table 1: p-value for different log transformation

Times of log	p-valud (ADF)	p-value decreased
1	0.1984	0.1367
2	0.1887	0.0097
3	0.1849	0.0038
4	0.1808	0.0041

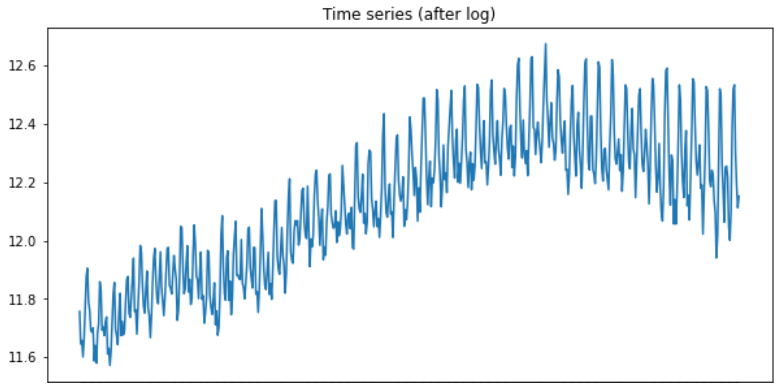

Figure 2.2: Timse series after taking log

The upgoing trend on mean is observed, thus, we further take difference to the data. The time series after first log and first difference transformation shows a high improvement in stationarity, and in Table 2.2, the Augmented Dickey-Fuller (ADF) test also gives a p-value(2.0785e-7) smaller than 0.05, rejecting the null hypothesis of non-stationarity.

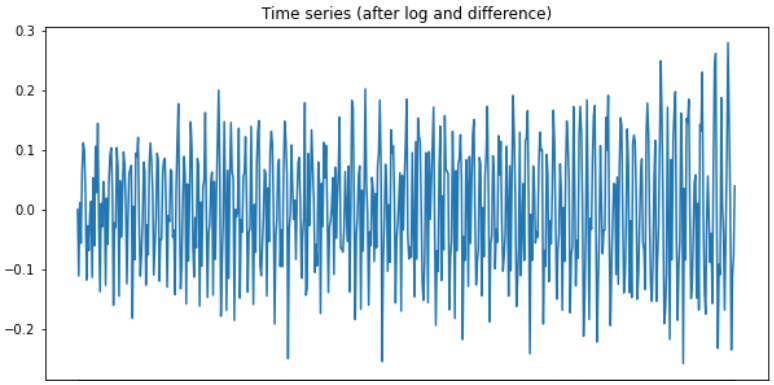

Figure 2.3: Time series after first log and first difference

Table 2: ADF test result for data after first log and first difference

ADF	p-valud	Used lag	Nobs	Critical values	Icbest
-5.9573	$2.0785e^{-7}$	19	568	{'1%': -3.4419, '5%': -2.8666, '10%': -2.5695}	-1879.4611

After log and difference transformation, we reach a stationary time series data. A Ljung-box test with 24 lags ($588^{0.5} \approx 24$) is conducted, and the test statistic as well as p-value are 2022.6217 and less than 0.000 respectively, which rejects the null hypothesis of white noise, encouraging further model-building. Moreover, when examining the data's autocorrelation function (ACF) plot, significant peaks at lag 12, 24 and 36 are observed (Figure 2.4), suggesting a yearly trend on data since one year consists of twelve months. Therefore, we will consider seasonal differencing of period twelve and building SRIMA models in the next modelling section.

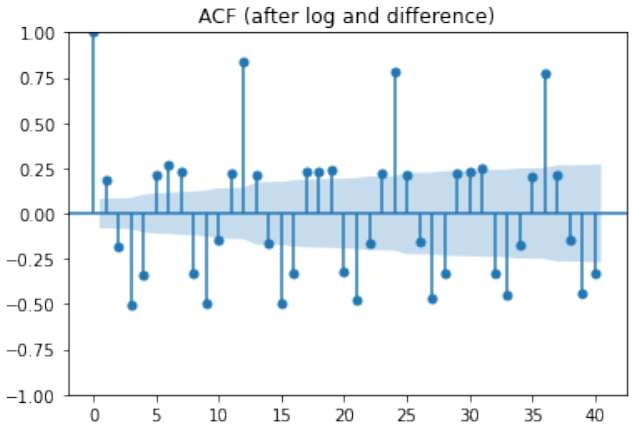

Figure 2.4: ACF after first log and first difference

2.2 New Energy Generation

We examine the time series data on the electricity generated by new energy (Figure 2.5), which presents a continuously ascending trend on mean, and slightly growth on variance. We take first log and first difference to tackle these two problems, and the resulting stationary time series is shown in Figure 2.6.

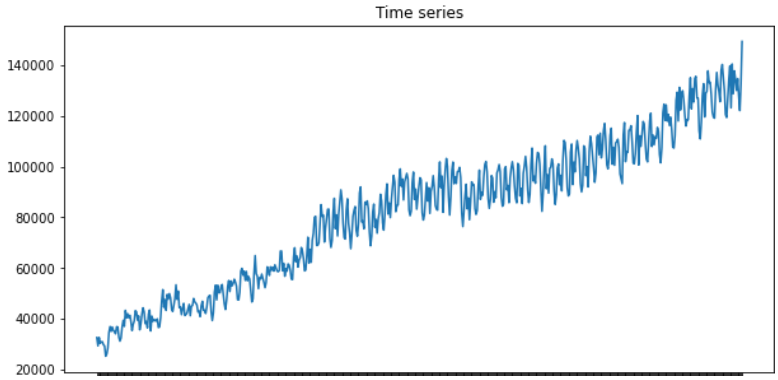

Figure 2.5: Time Series of new energy generation data

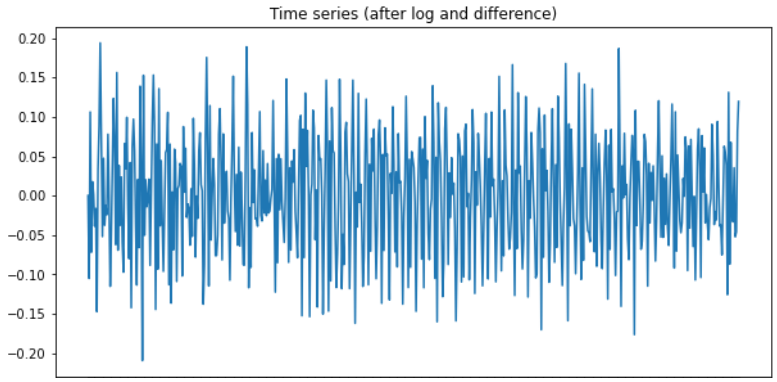

Figure 2.6: Time series after first log and first difference (New Energy)

Similar to the previous thermal power generation data exploration, we observed significant signals at lag 12, 24, and 36 on the ACF plot (Figure 2.7). Hence, we will start building the models with care taken on the data's seasonality.

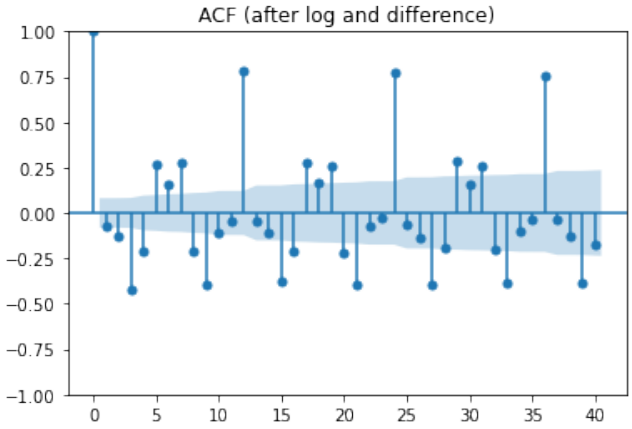

Figure 2.7: ACF after first log and first difference (New Energy)

3 Model Analysis

3.1 Seasonality Detection

We decompose the logged and differentiated data with period 12 and obtain the following figures. The first graph shows a fluctuation between -0.3 and 0.3. The second graph displays

the data with seasonality removed, which fluctuates between -0.15 to 0.1. The third graph demonstrates clear seasonality; thus, we conclude the suitability of using a seasonal model.

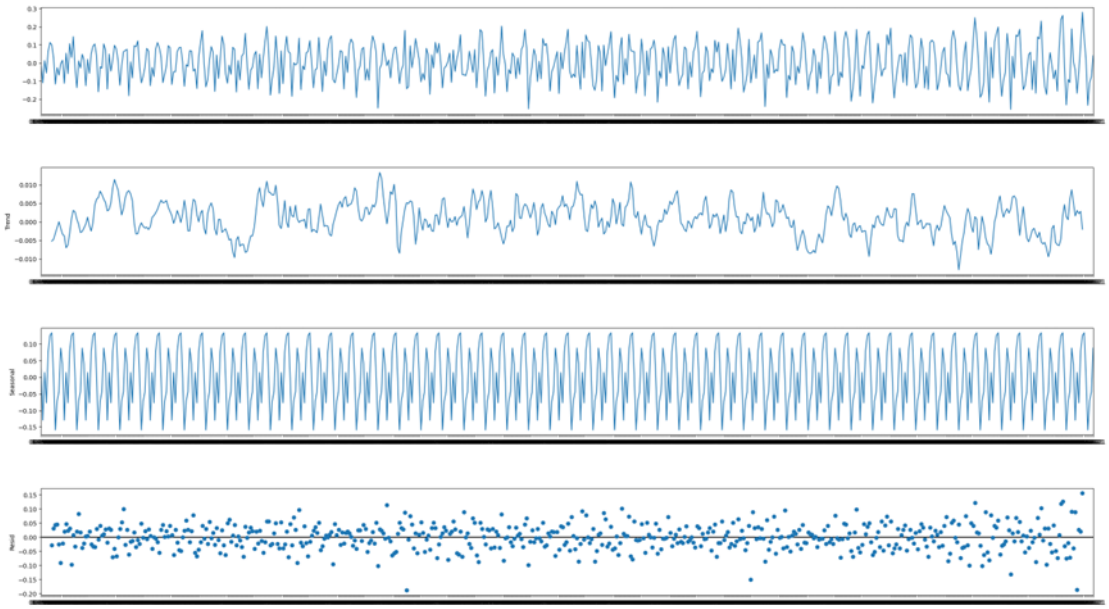

Figure 3.1: Seasonality plot

Next, we difference by the period of twelve months. From the figures of ACF and PACF, we observe in the ACF and PACF graphs below that lag 12 exceed the boundary of Bartletts' approximation negatively, compared to the postive exceed in the ACF figure without seasonal differencing. It is suspected that the model is overfitting. The PACF model has even more excessive values. These imply that a satisfactory model cannot be obtained by simply differencing once.

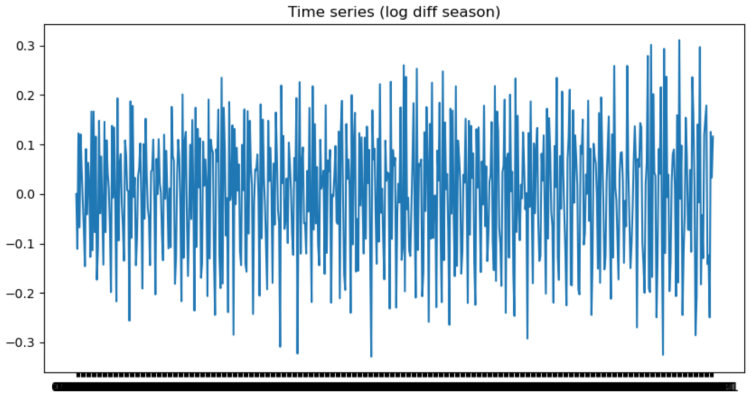

Figure 3.2: Difference with seasonality

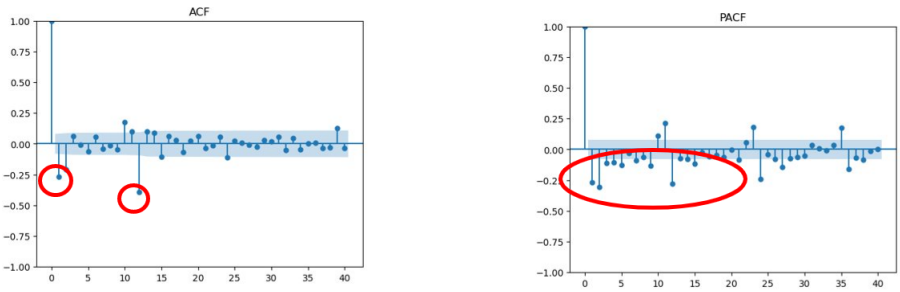

Figure 3.3: ACF and PACF when SARIMA D=1

Since the seasonal effect is still present, we revisit the ACF figure without differencing again.

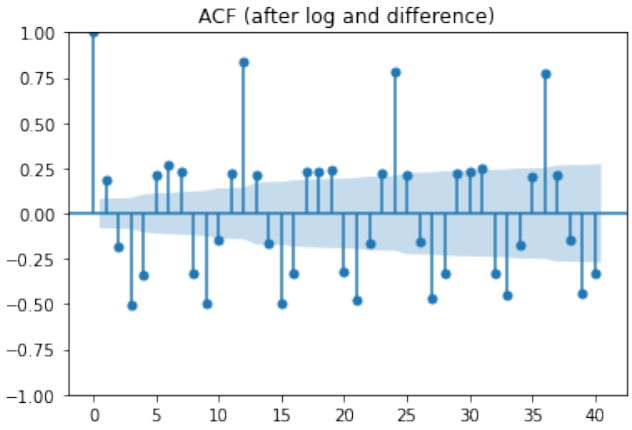

Figure 3.4: ACF after first log and first difference

It can be noticed that the values of lag 11 and 13 also exceed the Bartlett's approximation positively, while the lag 9, 10, 14 and 15 exceed the Bartlett's approximation negatively. This kind of pattern is similar to the pattern of the combination of MA and SMA process. Moreover, there are still many outstanding lags after lag 12, with slowly decreasing trend, which can be generated by the combination of MA and SAR process. Therefore, in the next section, SMA and SAR process will be considered for model fitting.

3.2 Model Specification

In this section, we use the value of BIC, instead of AIC, as the selection criteria because BIC's penalty for the number of parameters is higher than AIC's and is better for large datasets and industrial data. The reasons why we do not choose models mainly based on Q-Q plot, histogram, ACF and PACF figures are: 1) during the selection process, the first two figures show excellent results in most cases, and it is difficult to identify which model is better; 2) both ACF and PACF have many values of lags exceed Bartletts' approximation and cannot provide strong support to judgement.

To test the necessity of adding the seasonal effect, we set aside seasonality for now and apply grid search for the lowest possible BIC value to ARIMA(p, 1, q). The result suggests ARIMA(3, 1, 1), with a BIC value of -1248.44. The diagnostic plots for this model suggest the residuals have a mean of zero but a left-tailed variance. The histogram plus estimated density shows a skewed KDE. The Normal Q-Q plot is not a well fit, excluding the normality of residuals. The ACF has several lags outside the boundaries; thus, we conclude the ARIMA(3, 1, 1) is not accurate enough.

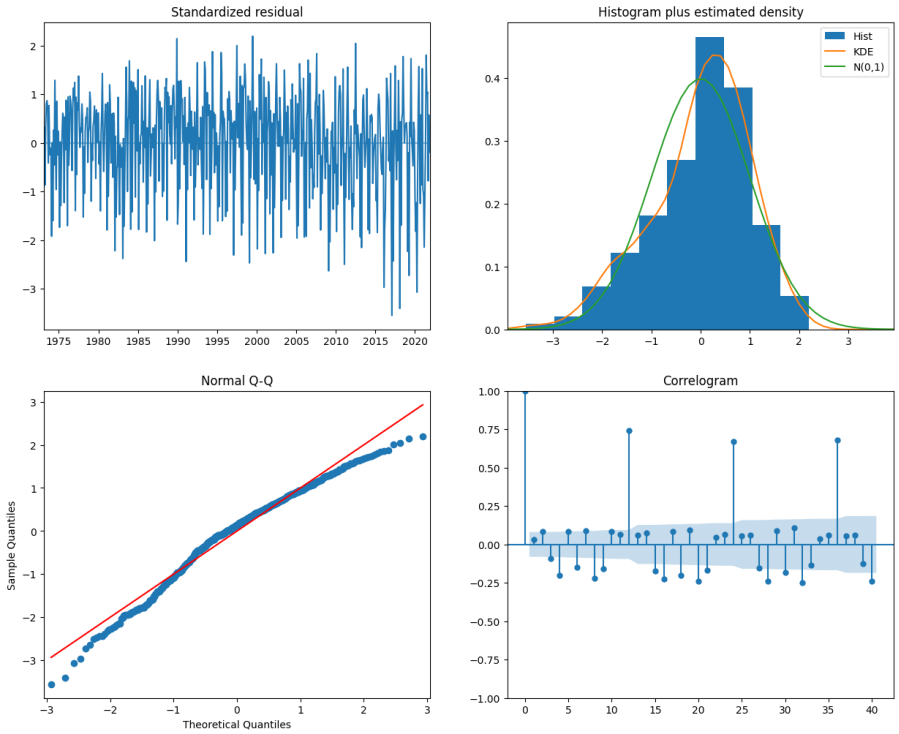

Figure 3.5: Diagnostic plot for ARIMA(3, 1, 1)

Adding the seasonal effect again, we first test a simple model $ARIMA(0,1,0) \times (0,1,0)_{12}$. It turns out to have a BIC of -1676.48, much smaller than the best ARIMA model, as we proposed earlier. We further demonstrated the necessity of seasonality. The next step is to find optimal SAR and SMA models by modifying the values of P and Q.

3.2.1 Finding SAR

The BIC of the ARIMA(0, 1, 0) model is -991.22. As the ACF and PACF of seasonality plots when D=1 appear to be chaotic, we cannot tell if the seasonality comes from SAR or SMA. Therefore, we first hold D as zero and increase P by 1, which gives $SARIMA(0,1,0) \times (1,0,0)_{12}$, with BIC equals to -1733.35, decreasing 742 from the ARIMA model's BIC. The p-value of ar.S.L12 is much smaller than 0.05, indicating that SAR is a feasible direction. If we increase P by 1 again, getting $SARIMA(0,1,0) \times (2,0,0)_{12}$. The BIC decreases by 68 to -1801.30, which is not much change. If considering $SARIMA(0,1,0) \times (1,1,0)$, BIC decreases to -1768.85, which is a smaller change. Therefore, we do not increase the value of D for now and continue with $SARIMA(0,1,0) \times (2,0,0)_{12}$. Then we try in three directions: increasing the value of P, p, and q, respectively. The effect on BIC is as the following:

Table 3: Start from $SARIMA(0,1,0) \times (2,0,0)_{12}$

model	modification	BIC
$SARIMA(0,1,0) \times (3,0,0)_{12}$	P=3	-1850.16
$SARIMA(1,1,0) \times (2,0,0)_{12}$	p=1	-1831.84
$SARIMA(0,1,1) \times (2,0,0)_{12}$	q=1	-1870.06

Starting from $SARIMA(0,1,0) \times (2,0,0)_{12}$, we choose the model that decreases BIC the most, which is $SARIMA(0,1,1) \times (2,0,0)_{12}$, and continue with this one.

Table 4: Start from $SARIMA(0,1,1) \times (2,0,0)_{12}$

model	modification	BIC
$SARIMA(0,1,1) \times (3,0,0)_{12}$	P=3	-1918.31
$SARIMA(1,1,1) \times (2,0,0)_{12}$	p=1	-1920.19
$SARIMA(0,1,2) \times (2,0,0)_{12}$	q=2	-1904.36

After the second test, we proceed with $SARIMA(1,1,1) \times (2,0,0)_{12}$, since its decrease in BIC is the largest.

Table 5: Start from $SARIMA(1,1,1) \times (2,0,0)_{12}$

model	modification	BIC
$SARIMA(1,1,1) \times (3,0,0)_{12}$	P=3	-1968.17
$SARIMA(2,1,1) \times (2,0,0)_{12}$	p=2	-1907.68
$SARIMA(1,1,2) \times (2,0,0)_{12}$	q=2	-1914.96

From the third test, we observe the largest change in BIC is -48, while the other two have increased BIC. Since the difference is smaller than 50, the third test is not so meaningful, and thus, we conclude with the second test's result: $SARIMA(1,1,1) \times (2,0,0)_{12}$.

3.2.2 Finding SMA

Trying out SMA models is similar to the procedures in SAR. We first set D=0 and Q=1, which is $SARIMA(0,1,0) \times (0,0,1)_{12}$, and gives a BIC of -1340.58. If D=1 and Q=1, $SARIMA(0,1,0) \times (0,1,1)_{12}$, the BIC is -1928.94, which is a significant decrease. Therefore, we continue with D=1 and increase the values of Q, q and p, respectively. After every test, we continue the model that brings the biggest decrease to BIC.

Table 6: Start from $SARIMA(0,1,0) \times (0,1,1)_{12}$

model	modification	BIC
$SARIMA(0,1,0) \times (0,1,2)_{12}$	Q=2	-1926.10
$SARIMA(0,1,1) \times (0,1,1)_{12}$	q=1	-1985.142
$SARIMA(1,1,0) \times (0,1,1)_{12}$	p=1	-1955.836

From the first test, we observe that increasing Q brings up the value of BIC, so we only increase q and p in the next test.

Table 7: Start from $SARIMA(0,1,1) \times (0,1,1)_{12}$

model	modification	BIC
$SARIMA(0,1,2) \times (0,1,1)_{12}$	q=2	-2020.32
$SARIMA(1,1,1) \times (0,1,1)_{12}$	p=1	-2032.43

The change of BIC after adding 1 to p is smaller than 50; therefore, we disregard this test and conclude with $SARIMA(0,1,1) \times (0,1,1)_{12}$.

3.2.3 Finding SARMA

We start with $SARIMA(0,1,0) \times (1,0,1)_{12}$, which gives a BIC equal to -1944.43. Same as the above procedures, we increase P, Q, p, and q by 1, respectively. The results show that

unit increments to P and Q bring up BIC, but BIC decreases more than 50 to -1996.48 when q goes up by 1. Then we continue to $SARIMA(0,1,1) \times (0,1,1)_{12}$. By comparing the effect of increasing p and q, we find that $SARIMA(1,1,1) \times (1,0,1)_{12}$ gives BIC=-2046.41. Further changes to p and q based on this model result in a higher BIC.

In summary, the above three directions generate three potential models:
model 1, $SARIMA(1,1,1) \times (2,0,0)_{12}$;
model 2, $SARIMA(0,1,1) \times (0,1,1)_{12}$;
model 3, $SARIMA(1,1,1) \times (1,0,1)_{12}$.

3.3 Model Diagnostics

For model 1,$SARIMA(1,1,1) \times (2,0,0)_{12}$, the four parameters have p-values smaller than 0.001, so they are statistically significant. The residuals are likely to be white noise, as the Ljung-box test has a p-value of 0.65, greater than 0.05. In the diagnostic plots, the residuals have a mean of zero; the histogram plus KDE approximates a normal distribution, and the Q-Q plot displays a good fit. However, the ACF plot has several lags outside Bartlett's approximation boundaries, with lag 24 going significantly beyond the border.

SARIMAX Results

Dep. Variable:	y	**No. Observations:**	588
Model:	SARIMAX(1, 1, 1)x(2, 0, [], 12)	**Log Likelihood**	976.031
Date:	Tue, 06 Dec 2022	**AIC**	-1942.063
Time:	20:14:13	**BIC**	-1920.188
Sample:	01-01-1973	**HQIC**	-1933.539
	- 12-01-2021		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	0.5569	0.036	15.630	0.000	0.487	0.627
ma.L1	-0.9751	0.010	-101.989	0.000	-0.994	-0.956
ar.S.L12	0.5809	0.037	15.716	0.000	0.508	0.653
ar.S.L24	0.3362	0.038	8.830	0.000	0.262	0.411
sigma2	0.0020	0.000	16.176	0.000	0.002	0.002

Ljung-Box (L1) (Q):	0.20	**Jarque-Bera (JB):**	0.82
Prob(Q):	0.65	**Prob(JB):**	0.66
Heteroskedasticity (H):	1.40	**Skew:**	-0.04
Prob(H) (two-sided):	0.02	**Kurtosis:**	2.84

Figure 3.6: Fitness test for model 1

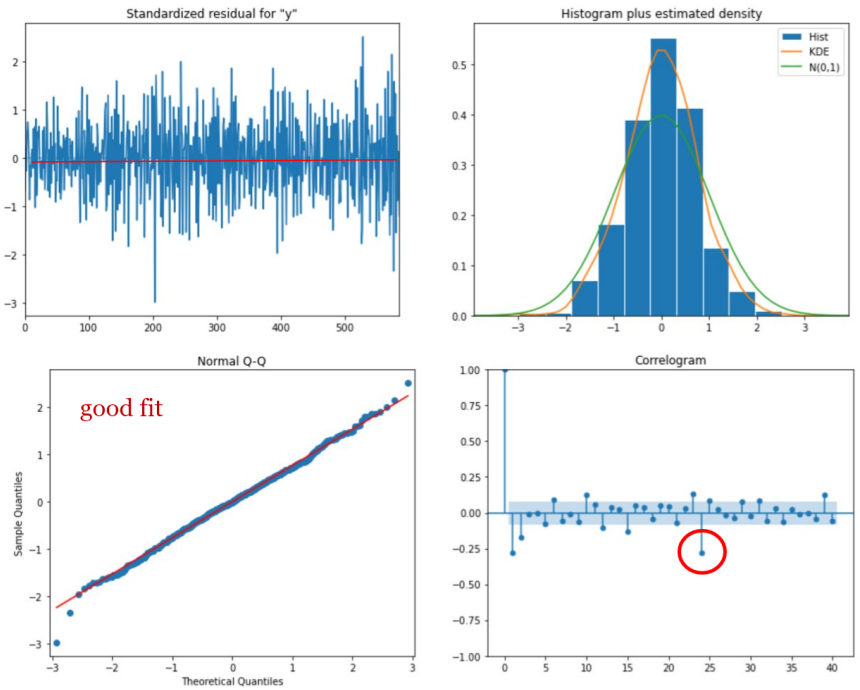

Figure 3.7: Diagnostic plot for model 1

For model 2, $SARIMA(0,1,1) \times (0,1,1)_{12}$, the four parameters have p-values smaller than 0.001, so they are statistically significant. The residuals are likely white noise, as the Ljung-box test has a p-value of 0.23, greater than 0.05. In the diagnostic plots, the residuals have a mean of zero; the histogram plus KDE approximates a normal distribution, and the Q-Q plot displays a good fit. However, the ACF plot has several lags outside Bartlett's approximation boundaries, with lag 2 to lag 5 all exceeding the border, indicating that the fitness is not enough.

SARIMAX Results

Dep. Variable:	y	**No. Observations:**	588
Model:	SARIMAX(1, 1, 1)x(3, 0, [], 12)	**Log Likelihood**	1003.203
Date:	Tue, 06 Dec 2022	**AIC**	-1994.406
Time:	20:14:10	**BIC**	-1968.155
Sample:	01-01-1973	**HQIC**	-1984.177
	- 12-01-2021		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	0.5214	0.040	13.184	0.000	0.444	0.599
ma.L1	-0.9504	0.015	-62.390	0.000	-0.980	-0.921
ar.S.L12	0.4654	0.038	12.302	0.000	0.391	0.540
ar.S.L24	0.1606	0.042	3.828	0.000	0.078	0.243
ar.S.L36	0.3175	0.041	7.704	0.000	0.237	0.398
sigma2	0.0018	0.000	16.786	0.000	0.002	0.002

Ljung-Box (L1) (Q):	0.02	**Jarque-Bera (JB):**	0.10
Prob(Q):	0.88	**Prob(JB):**	0.95
Heteroskedasticity (H):	1.44	**Skew:**	-0.02
Prob(H) (two-sided):	0.01	**Kurtosis:**	3.05

Figure 3.8: Fitness test for model 2

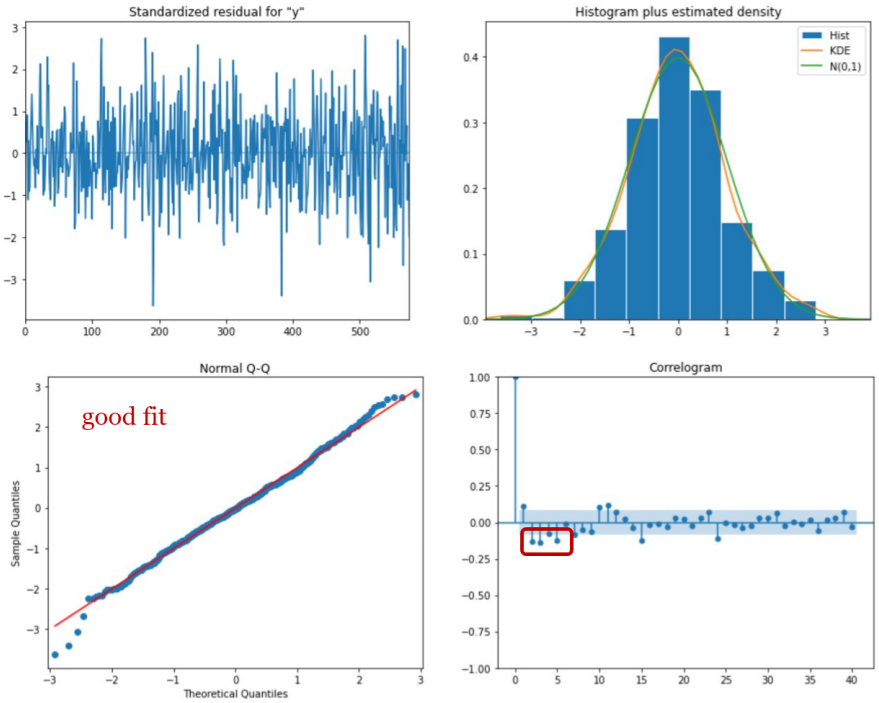

Figure 3.9: Diagnostic plot for model 2

For model 3, $SARIMA(1,1,1) \times (1,0,1)_{12}$, the four parameters have p-values smaller than 0.001, so they are statistically significant. The residuals are very likely to be white noise, as the Ljung-box test has a p-value of 0.82, much greater than 0.05. In the diagnostic plots, the residuals have a mean of zero; the histogram plus KDE approximates a normal distribution, and the Q-Q plot displays a good fit. However, almost all lags in the ACF plot are within Bartlett's approximation boundaries. A few exceptions at lag 10 and lag 15 may be the consequences of random effects.

SARIMAX Results

Dep. Variable:	y	**No. Observations:**	588
Model:	SARIMAX(0, 1, 2)x(0, 1, [1], 12)	**Log Likelihood**	1022.867
Date:	Tue, 06 Dec 2022	**AIC**	-2037.734
Time:	20:14:15	**BIC**	-2020.316
Sample:	01-01-1973	**HQIC**	-2030.940
	- 12-01-2021		
Covariance Type:	opg		

	coef	**std err**	**z**	**P>|z|**	**[0.025**	**0.975]**
ma.L1	-0.4512	0.035	-12.927	0.000	-0.520	-0.383
ma.L2	-0.2807	0.039	-7.280	0.000	-0.356	-0.205
ma.S.L12	-0.8214	0.028	-28.993	0.000	-0.877	-0.766
sigma2	0.0016	8.64e-05	18.827	0.000	0.001	0.002

Ljung-Box (L1) (Q):	0.90	**Jarque-Bera (JB):**	6.26
Prob(Q):	0.34	**Prob(JB):**	0.04
Heteroskedasticity (H):	1.43	**Skew:**	0.04
Prob(H) (two-sided):	0.01	**Kurtosis:**	3.51

Figure 3.10: Fitness test for model 3

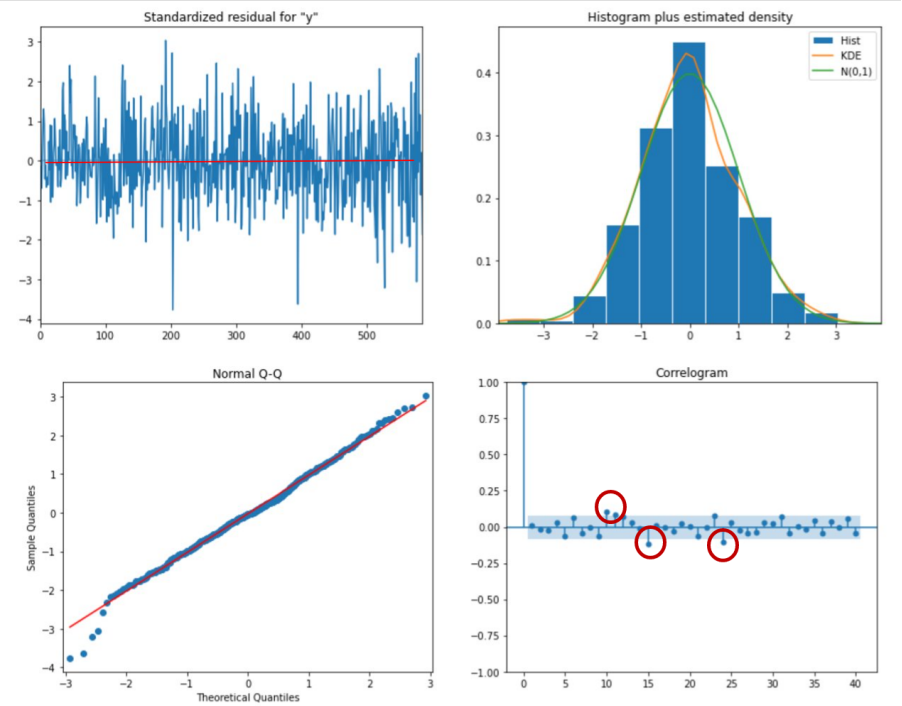

Figure 3.11: Diagnostic plot for model 3

In conclusion, Model 3 is the best fit. It has the lowest BIC value. Its Ljung-box test has the biggest p-value, so the residuals are most likely to be white noises among the three models. Lastly, its ACF plot has the best performance. Therefore, we conclude the model selection with $SARIMA(1,1,1) \times (1,0,1)_{12}$.

3.4 Parameter Estimation

We estimated the parameters for the transformed series based on the best model we selected for thermal electricity data $SARIMA(1,1,1)\times(1,0,1)_{12}$ and new energy data $SARIMA(1,1,1)\times(1,0,1)_{12}$. For both datasets, the p-value of every parameter was smaller than 0.05. We also calculated the parameters for the original time series model.

Note that the coefficient of ∇Z_{t-1} in the new energy model is larger than that of the thermal electricity model. Hence, the net generation of new energy grows faster than thermal electricity.

Model for Thermal electricity time series data:

$$
\begin{aligned}
(1-\phi B)(1-\Psi B^{12})\nabla Z_t =& (1-\theta B)(1-\Theta B^{12})a_t \\
\nabla Z_t =& \phi\nabla Z_{t-1} + \Psi\nabla Z_{t-12} - \phi\Psi\nabla Z_{t-13} + a_t - \theta a_{t-1} - \Theta a_{t-12} + \theta\Theta a_{t-13} \\
\nabla Z_t =& 0.5274\nabla Z_{t-1} + 0.9950\nabla Z_{t-12} - 0.5248\nabla Z_{t-13} \\
& + a_t - 0.9191a_{t-1} - 0.8174a_{t-12} + 0.7513a_{t-13} \\
Z_t =& 1.5274Z_{t-1} - 0.7412Z_{t-2} + 0.9950Z_{t-12} - 1.5198Z_{t-13} + 0.5248Z_{t-14} \\
& + a_t - 0.9191a_{t-1} - 0.8174a_{t-12} + 0.7513a_{t-13}
\end{aligned}
\tag{3.1}
$$

where $\{a_t\} \sim WN(0, 0.0017)$.

Model for New energy time series data:

$$\begin{aligned}(1-\phi B)(1-\Psi B^{12})\nabla Z_t =&(1-\theta B)(1-\Theta B^{12})a_t\\ \nabla Z_t =&\phi\nabla Z_{t-1}+\Psi\nabla Z_{t-12}-\phi\Psi\nabla Z_{t-13}+a_t-\theta a_{t-1}-\Theta a_{t-12}+\theta\Theta a_{t-13}\\ \nabla Z_t =&0.7412\nabla Z_{t-1}+0.9913\nabla Z_{t-12}-0.7348\nabla Z_{t-13}\\ &+a_t-0.9421a_{t-1}-0.7980a_{t-12}+0.7518a_{t-13}\\ Z_t =&1.7412Z_{t-1}-0.7412Z_{t-2}+0.9913Z_{t-12}-1.7261Z_{t-13}+0.7348Z_{t-14}\\ &+a_t-0.9412a_{t-1}-0.7980a_{t-12}+0.7518a_{t-13}\end{aligned} \tag{3.2}$$

where $\{a_t\} \sim WN(0, 0.0012)$.

4 Prediction

4.1 Thermal electricity

The table below shows the actual value (log-transformed) and the prediction with a 95% confidence interval for thermal electricity usage from January 2022 to July 2022.

Table 8: Actual value (log-transformed) and prediction of thermal electricity

number	Actual value	Predicted value	Lower bound	Upper bound
1	12.339637	12.215872	12.13807	12.293674
2	12.148517	12.108351	12.018075	12.198627
3	12.073038	12.085103	11.99004	12.180166
4	12.002312	11.99345	11.895923	12.090977
5	12.169353	12.103748	12.004613	12.202883
6	12.350256	12.289804	12.189421	12.390188
7	12.537017	12.464019	12.362553	12.565486

We can conclude from the figure that our model's prediction is fairly accurate. The blue curve is the actual value (log-transformed); the green curve is the prediction; the grey region is the 95% CI of prediction.

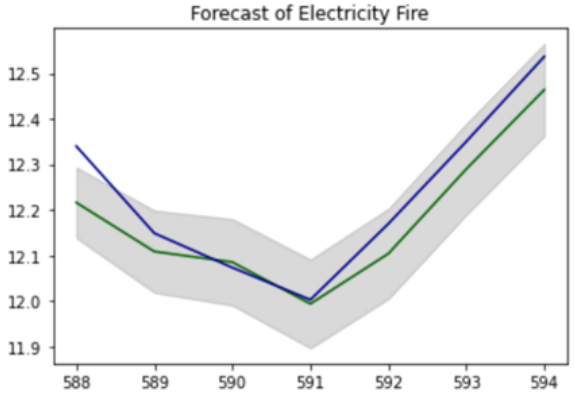

Figure 4.1: Comparison of the actual value (logged) and prediction of thermal electricity

The table below shows the actual value (original) and the prediction for thermal electricity usage from January 2022 to July 2022. We calculated the predicted value by $E[exp(X)] = e^{\mu+\frac{\sigma^2}{2}}$.

Table 9: Actual value (original) and prediction of thermal electricity

number	Actual value	Predicted value
1	228579.033	202128.6879
2	188813.956	181572.7582
3	175086.978	177420.7305
4	163131.504	161892.5241
5	192789.178	180778.3767
6	231019.109	217752.757
7	278456.534	259200.9759

The figure below is the visualization of the data. The blue curve is the actual value (original); the green curve is the prediction; the grey region is the 95% CI of prediction.

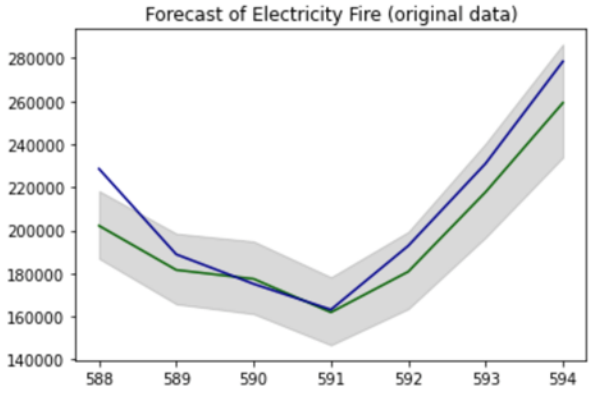

Figure 4.2: Comparison of the actual value (original) and prediction of thermal electricity

4.2 New energy

The table below shows the actual value (log-transformed) and the prediction with 95% confidence interval for new energy usage from January 2022 to July 2022.

Table 10: Actual value (log-transformed) and prediction of new energy

number	Actual value	Predicted value	Lower bound	Upper bound
1	11.914184	11.934303	11.866696	12.001911
2	11.835459	11.82875	11.742284	11.915216
3	11.917679	11.883975	11.787066	11.980883
4	11.851495	11.831995	11.728529	11.935462
5	11.916788	11.891902	11.783979	11.999825
6	11.903021	11.887474	11.776326	11.998621
7	11.886192	11.86854	11.754928	11.982151

We can conclude from the figure below that our model's prediction is fairly accurate. The blue curve is the actual value (log-transformed); the green curve is the prediction; the grey region is the 95% CI of prediction.

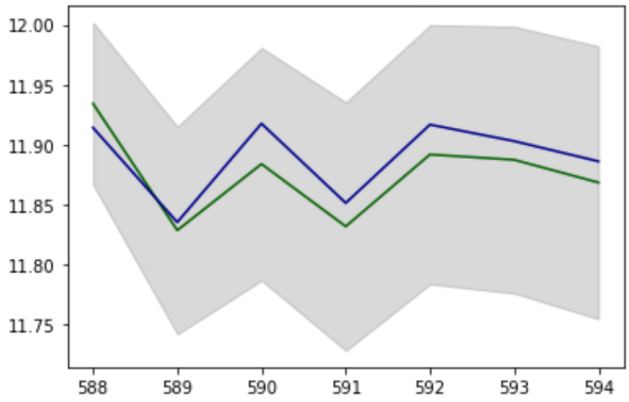

Figure 4.3: Comparison of the actual value (logged) and prediction of new energy

The table below shows the actual value (original) and the prediction for new energy usage from January 2022 to July 2022. We calculated the predicted value by $E[exp(X)] = e^{\mu+\frac{\sigma^2}{2}}$.

Table 11: Actual value (original) and prediction of new energy

number	Actual value	Predicted value
1	149370.395	152496.6509
2	138062.17	137272.4725
3	149893.343	145102.7365
4	140293.897	137776.537
5	149759.79	146300.4925
6	147712.159	145667.496
7	145247.132	142945.6774

The figure below is the visualization of the data. The blue curve is the actual value (original); the green curve is the prediction; the grey region is the 95% CI of prediction.

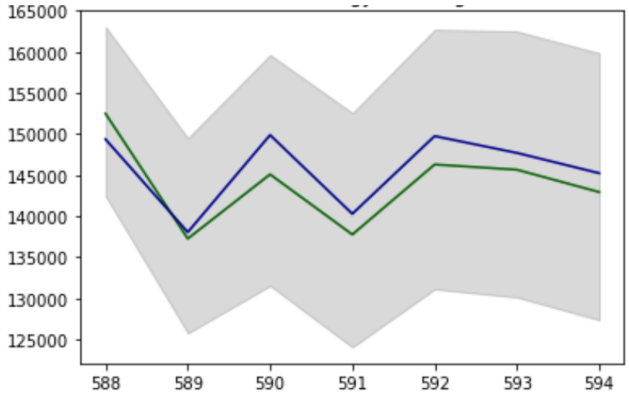

Figure 4.4: Comparison of the actual value (original) and prediction of new energy

5 Further Discussion I: Machine Learning Approach

In the previous sections, we have already built our SARIMA model, which performs well in forecasting. In this section, we further explored the machine learning approach. In Section 5.1, we will discuss short-term forecasting based on long short-term memory (LSTM). And in

Section 5.2, we proposed an LSTM-based encoder-decoder model with an attention mechanism to make longer-term forecasting.

5.1 LSTM for short-term forecasting

Long short-term memory (LSTM) [2] is one of the many variations of recurrent neural network (RNN) architecture, widely applied in time series. LSTM is designed to address some difficulties of capturing long-term dependencies with classic RNN due to the gradient vanishing problem. We will apply LSTM to do short-term forecasting over our data set and investigate the performance.

5.1.1 LSTM and Time Series

A long short-term memory is defined as follows [3]. At the time t, it receives an input vector $xt \in \mathbb{R}^m$ of observations. Here in our uni-variable time series, we have $m = 1$. Note that this can be easily generalized to multi-variable time series. Additionally, an input vector $h_{t-1} \in \mathbb{R}^k$ represents the previous hidden state, and a cell state $c_{t-1} \in \mathbb{R}^k$ from the previous time step will also be received as the input vector. It will compute three gates i_t, f_t, and o_t controlling, respectively, which part of the information to insert, how much information to forget, and which information should be output. It also computes a new value for the cell state c_t and a new hidden state. The information flow and computation procedure are as follows.

$$i_t = \sigma(W_i[x_t, h_{t-1}] + b_i) \tag{5.1}$$

$$f_t = \sigma(W_f[x_t, h_{t-1}] + b_f) \tag{5.2}$$

$$o_t = \sigma(W_o[x_t, h_{t-1}] + b_o) \tag{5.3}$$

$$\tilde{c_t} = \tanh\left(W_c[x_t, h_{t-1}] + b_c\right) \tag{5.4}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c_t} \tag{5.5}$$

$$h_t = o_t \odot \tanh c_t \tag{5.6}$$

where σ is the element-wise logistic sigmoid function and $\odot$ designate the Hadamard product. We chose the base values h_0 and c_0 uniformly from $-0.08to0.08$. Finally, a new output is computed as

$$y_t = f(Wh_t + b) \tag{5.7}$$

where f is element-wise nonlinearity.

Before applying it, we wonder why it works well for the general time series data. The idea is it is similar to the classical time-series model in philosophy. The ARMA model shown below is the most extensive collection of applicable linear stationary time series models we have learned. Let's focus on exactly what we did in this model.

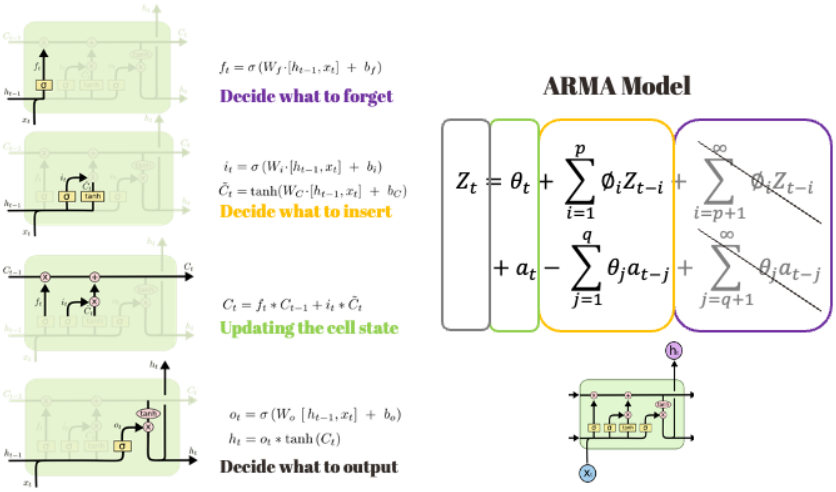

Figure 5.1: LSTM and ARMA

As shown in Figure 5.1, ARMA model truncates the past information at some lags; similarly, we have a forgetting gate f_t in LSTMs to decide which part should be forgotten in long-term memory and how much of the remaining memory to retrain. Secondly, we capture the past information to make the prediction; similarly, in LSTMs, we use the inserting gate i_t to insert the information from the observed input. And next, the ARMA model consists of white noise or simultaneous information; similarly, in LSTMs, we have the cell state c_t to update the most recent information. Finally, the model will give output or prediction, which is again consistent between ARMA and LSTMs.

5.1.2 Results

We trained our model over 420 months from January 1975 to December 2009. And set twelve years from 2010 to 2022 as the test set. The result is shown below.

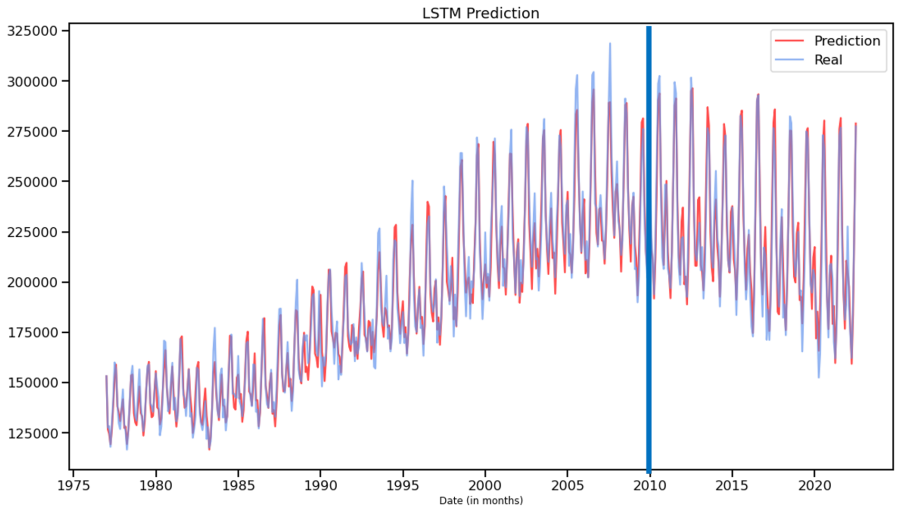

Figure 5.2: Result for LSTM based on past 48 months' data

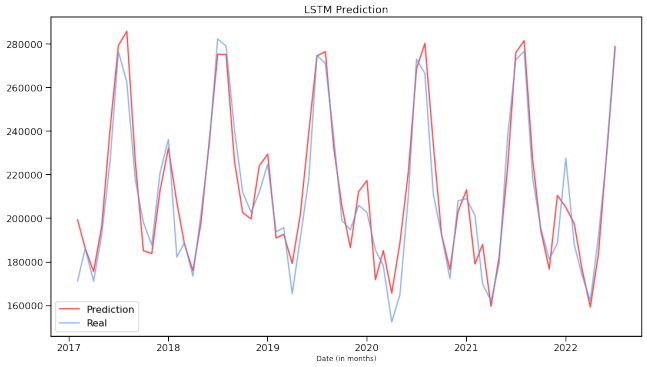

Figure 5.3: Result for LSTM based on past 48 months data (2017-2022)

We can see that the model fits well on the train set and gives us excellent forecasting over the test set. However, our 2-layer LSTM model has an exploding number of parameters compared to the SARIMA model. The next question is, is it an efficient model? Do LSTMs really have long-term memory? Do we need this long-term memory in our data set?

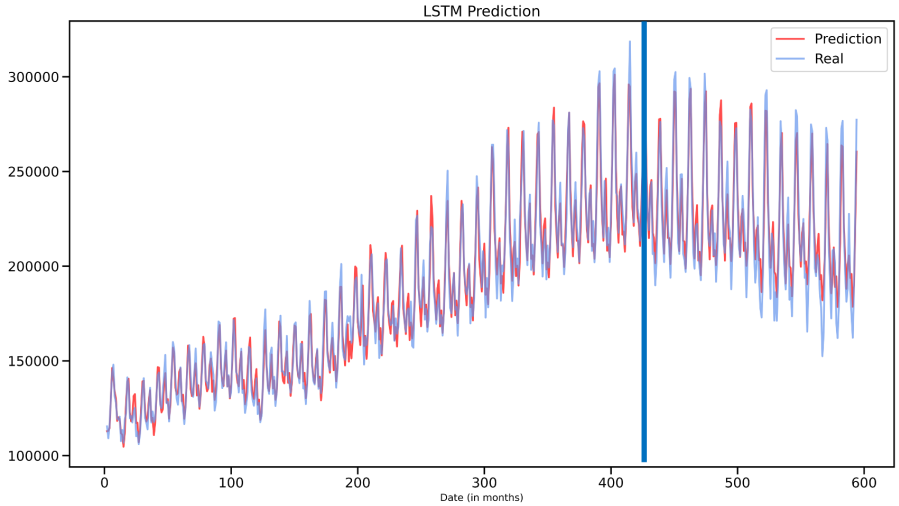

Figure 5.4: Result for LSTM based on past two months' data)

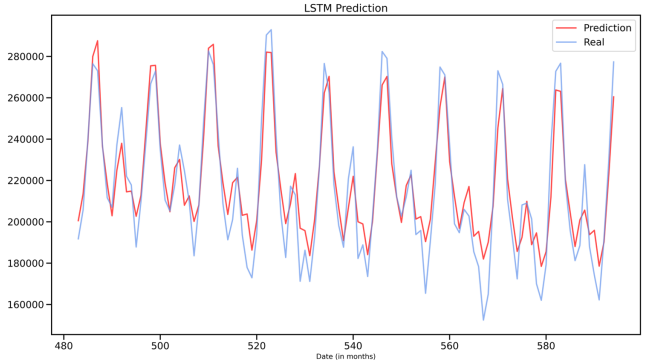

Figure 5.5: Result for LSTM based on past two months' data (2017-2022)

As shown above, Figures 5.4 and 5.5 are the results for LSTM prediction based only on the past two months. The red line gives the rolling two-lag prediction. It also gives relatively accurate forecasting. Interestingly, we don't need the long-term part significantly in this data

set in terms of a short-term prediction. Another interesting observation is that we can use this pre-trained model to do rolling forecasting without training the model again and again. Therefore, the nature of thermal electricity generation data is more stable than we assumed. Note that the ARMA model is only for short-term forecasting. Then we consider further that, based on the ideas in the theory of time series, how to construct a new architecture in the machine learning approach to make a longer-term prediction.

5.2 LSTM-Based Encoder-Decoder Model with Attention

In this section, we tried to design an architecture for time series forecasting based on the nature of the time series, i.e., linear autoregressive property, seasonality, and nonlinearity.

5.2.1 Model Construction

We proposed the architecture, as shown in Figure 5.6. It is generally an encoder-decoder model, where the encoder and decoder are distinct n-layer LSTMs and connected by the attention mechanism. Note that our model can be easily generalized to multi-variable time series prediction.

The input to the encoder is the past time series realization $\{z_t : z_t \in \mathbb{R}^m\}$. Here in our thermal electricity generation data set, we have $m = 1$. The sequence of the encoder outputs $\{y_t : y_t \in \mathbb{R}^k\}$ with hidden dimension k will be pushed into the attention mechanism. The attention mechanism we applied here will be explained in the next section. The result of the attention block will be the input to the decoder, which is another n-layer LSTMs to generate the outputs. The information flow is shown below.

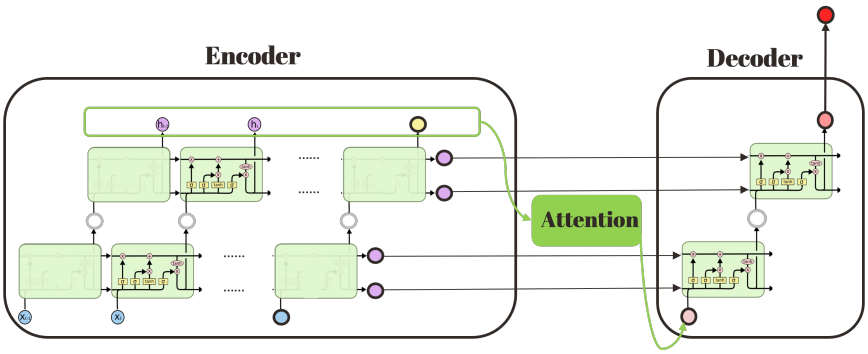

Figure 5.6: LSTM-based encoder-decoder with attention mechanism

Similar to section 5.1, before applying it, let's focus on why it may work for time series. LSTM is to simulate the philosophy of the ARMA model as explained in the previous section. Attention here tells the model which part of the past sequence should be paid more attention to; hopefully, it helps to capture the seasonality. Particularly, when we do the forecasting for the next January, we hope the model focus more on January in the previous years by the 12-months seasonality. And there is also a nonlinear part provided by the nonlinear activation functions.

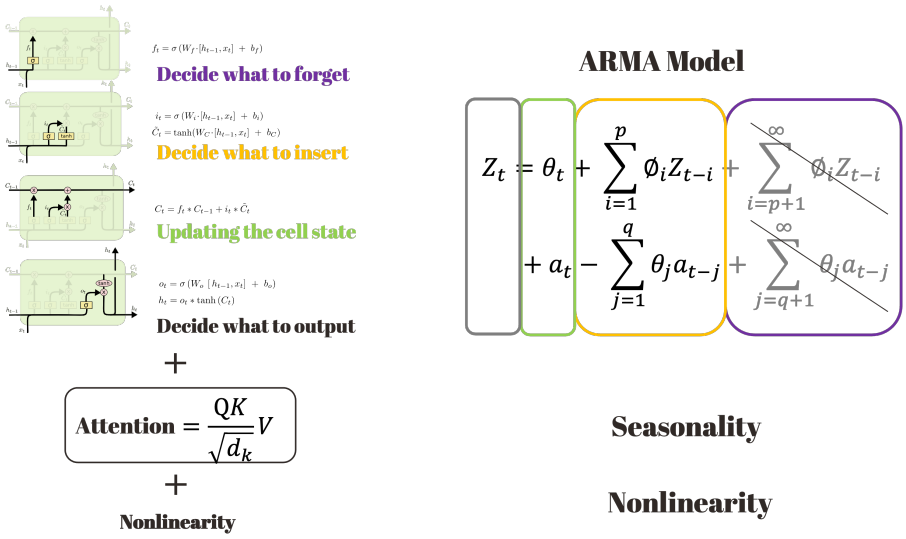

Figure 5.7: Relationship with time series

5.2.2 Attention Mechanism

The development of attention enables the decoder to attend to the whole sequence and thus use the information of the entire sequence during the decoding step. Our attention block maps a query and a set of key-value pairs to an output, where the query, keys, values, and output are all vectors. The output is computed as a weighted sum of the values, where the weight assigned to each value intuitively should be decided by the similarity of the query with the corresponding key.

In our model, the query should be a vector used to forecast in the decoder, which is a transformation of the last output of the encoder. Key-value pairs should be the transformation of the sequence of outputs of the encoder. Intuitively, before we do the forecasting by the particular input vector of the decoder, we use it as a query and ask each output of the encoder: how similar they are. If our query is more similar to some keys, we should be assigned a higher weight for the corresponding values in the output. Then the weighted output, which contains the weighted information from the whole past sequence, is the new input to the decoder to do the forecasting. Hence, the next problem is, given a query Q and a sequence of N keys K, how to determine the similarity between the query and key. We initially considered Mahalanobis distance as a measure of similarity, which is defined as follows

$$d_M(\vec{Q}, K) = \sqrt{(\vec{Q} - \vec{\mu})^\mathsf{T} S^{-1} (\vec{Q} - \vec{\mu})} \tag{5.8}$$

where $\vec{\mu} = (\mu_1, \mu_2, \mu_3, \ldots, \mu_N)^\mathsf{T}$ is the mean of K and S is a positive-definite covariance matrix. Then by the spectral theorem, we have

$$S^{-1} = W^T W \tag{5.9}$$

for some real matrix $W \in \mathbb{R}^N$. Alternatively, we considered to use W_1 and W_2 to approximate S^{-1} numerically. Let $x = \vec{Q} - \vec{\mu}$. We can see that

$$\begin{aligned} x^T S^{-1} x &= x^T W^T W x \\ &\approx x^T W_1^T W_2 x \\ &= (W_1 x)^T (W_2 x) \end{aligned} \tag{5.10}$$

Therefore, given the sequence of encoder outputs E_o, we compute query Q, key K, and value V as follows.

$$\begin{aligned} Q &= W_q E_o \\ K &= W_k E_o \\ V &= E_o \end{aligned} \tag{5.11}$$

It follows that the attention weight is

$$AttentionWeight = Softmax(\frac{Q \odot K}{\sqrt{d_k}}) \tag{5.12}$$

where $\odot$ is Hadamard product and d_k is the dimension of key. The denominator $\sqrt{d_k}$ is an empirical result from industry to avoid the weight becoming too large when the dimension is high. Consequently, for one-head attention, we have

$$output = AttentionWeight * V \tag{5.13}$$

If we consider the encoder as a compressor that sends the input sequence from the higher dimension to the compressed vector in a lower dimension. We may revisit this model in information theory. Note that the efficiency of one-head attention can be proved by maximizing the rate reduction via projected gradient flow [4]. The partial differentiation is exactly the residual of autoregression. This is another reason we tried to use the attention mechanism in the time series. And we hope that the relationship between attention and time series will be proven entirely in the near future.

5.2.3 Result and future work

The result is shown below in Figure 5.8. Unfortunately, because of the limited time and because we don't have GPUs to do the computation, we only trained a few steps. Although with a little training process, it can forecast the trend for the future years. And our next step is to modify the normalization of data to avoid truncation error caused by the float representation in the computer and use multi-head attention instead of one-head to enhance our capture of seasonality.

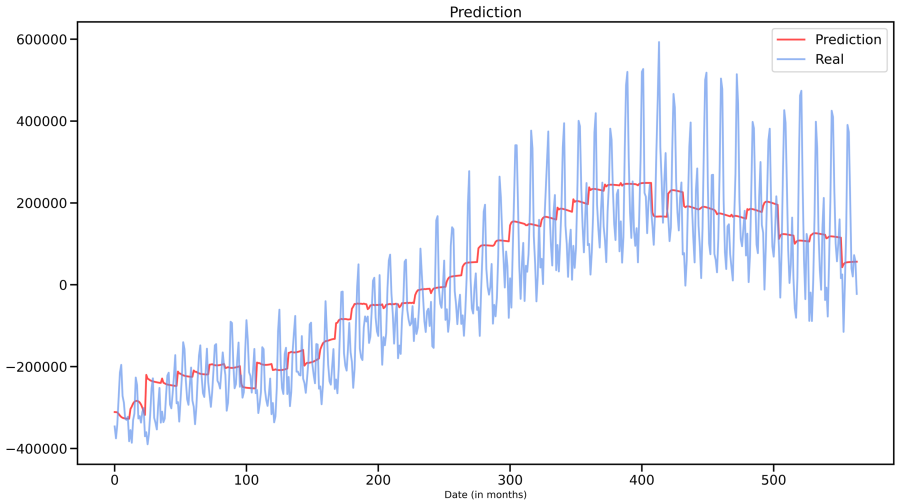

Figure 5.8: Result for LSTM-based encoder-decoder with attention model

5.3 Comments on Machine Learning Approach

The general ARMA model was described in the 1951 thesis of Peter Whittle, and it was popularized by George E. P. Box and Gwilym Jenkins about 50 years ago. In short-term forecasting, our SARIMA model has performed well, as shown in the previous section. Hence, the classical ARMA model has a long history but still has a very strong application today. Nowadays, with the development of computers, we have more arithmetic power, and more and more scientists devote themselves to machine learning and deep learning, building all kinds of complex models and tuning parameters night and day to train the models. But the principle of effective models comes from the nature of time series. Looking back at the classical time series models, they are still instructive for building new models today, just like the comparison between LSTM and the ARMA model mentioned in this section. Last but not least, the take-home message of this section is the beauty of the classical ARMA model and the nature of the time series itself.

Hopefully, we will have more robust and efficient models for time series forecasting in the near future. And that model is not obtained through millions of days and nights of parameter tuning but an efficient white-box model that can be analyzed.

6 Further Discussion II: ARCH-type Model

Previously, we assumed that the residual plots of the fitted SARIMA models should follow the White Noise Process, which indicates that the variance of all the residual terms should be the same. However, it is only sometimes the case in real-world situations, as the variance could fluctuate regarding different events. Hence, to improve the predicted confidence intervals' reasonableness, ARCH-type models are constructed in this section.

6.1 Heteroskedasticity detection

We first would like to revisit the residual plots of fitted thermal electricity generation and new energy generation to see whether they have particular patterns. The two residual plots are shown below:

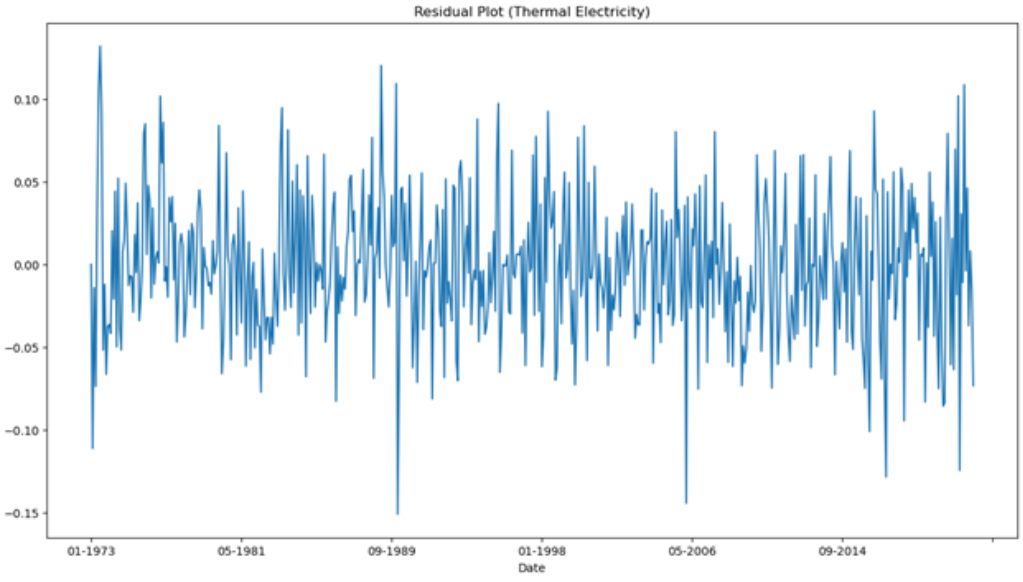

Figure 6.1: Thermal Electricity Residual plot

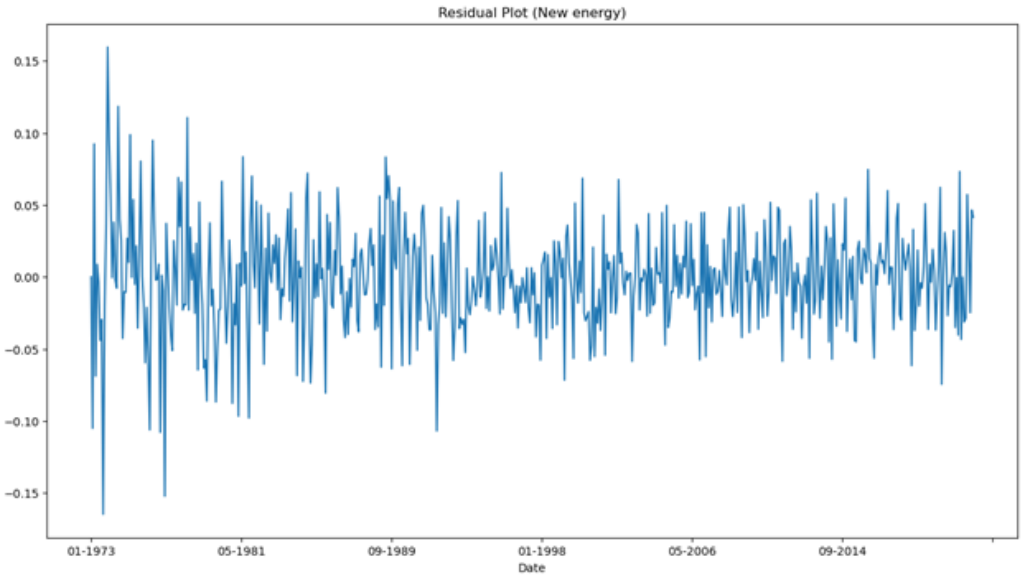

Figure 6.2: New Energy Residual plot

For the residual plot of Thermal Electricity, although there is some time that the variance is outstanding, the residual trend is somehow stable; hence, the variance might be constant. But for the residual plot of New energy, there is a clear pattern that the variance is shrinking from 1973 to 2021, which means that this trend is not likely to be a White Noise process.

Next step, different ARCH-type models to detect the heteroskedastic will be fitted to these two data sets. Before doing so, the ARCH-Test will be made to check whether the ARCH effect exists:

```
LM_pvalue = het_arch(electricity_fire_series_sarimaresidual, ddof = 4)[1]
print('LM-test-Pvalue:', '{:.5f}'.format(LM_pvalue))
```

```
LM-test-Pvalue: 0.04278
```

Figure 6.3: ARCH-Test Result for Thermal Electricity

```
LM_pvalue = het_arch(electricity_nonfire_series_sarimaresidual, ddof = 4)[1]
print('LM-test-Pvalue:', '{:.5f}'.format(LM_pvalue))
```

```
LM-test-Pvalue: 0.00000
```

Figure 6.4: ARCH-Test Result for New Energy

After checking these two results, since they are both less than 0.05, the inference can be continued further.

6.2 ARCH-type model fitting

6.2.1 ARCH-type model introduction

To apply these models for application, they are introduced as follows:

Consider an ARIMA-GARCH model, where r_t term no-longer follows WN(0,σ_t^2) with constant variance:

$$Z_t = \varphi_t + \sum_{i=1}^{p} \varphi_i Z_{t-i} + a_t - \sum_{i=1}^{q} \theta_j a_{t-j} + r_t$$

The GARCH (Generalized Autoregressive Conditional Heteroscedastic) formulation for r_t:

$$r_t = \sigma_t \varepsilon_t; \sigma_{t^2} = \omega + \sum_{j=1}^{s} \beta_j \sigma_{t-j}^2 + \sum_{j=1}^{m} \alpha_j a_{t-j}^2$$

It is noted that the terms $\sum_{j=1}^{s} \beta_j \sigma_{t-j}^2$ is similar to the "AR part" $\sum_{i=1}^{p} \varphi_i Z_{t-i}$ in the ARIMA model, which means that the current value of variance depends on the previous value of variance. The terms $\sum_{j=1}^{m} \alpha_j a_{t-j}^2$ is similar to the "MA part" $\sum_{i=1}^{q} \theta_i a_{t-i} + r_t$ as constructed in the ARIMA model. There is one constraint to ensure that the unconditional variance in the original ARIMA model is finite and constant, which is to meet the requirement of stationarity:

$$0 < \sum_{j=1}^{s} \beta_j + \sum_{j=1}^{m} \alpha_j < 1$$

There is another model called E-GARCH (Exponential Generalized Autoregressive Conditional Heteroscedastic), which has the formulation regarding σ_t^2 as:

$$ln(\sigma_t^2) = \omega + \sum_{j=1}^{s} \beta_j ln(\sigma_{t-j}^2) + \sum_{j=1}^{m} [\alpha_j (|\varepsilon_{t-j}| - E(|\varepsilon_{t-j}|)) a_{t-j}^2 + \gamma_i \varepsilon_{t-j}]$$

The function uses log transformation to capture the exponential increase or decrease trend regarding variance. Note that this model has no restrictions on parameters as $ln(\sigma_t^2)$ may be negative.

6.2.2 Model fitting and Selection

During the model fitting procedure, we still use BIC & p-value as the criteria consistent with the previous analysis. Then, the best choice for the residual data of Thermal power is ARCH (1), and the residual data of New energy is E-GARCH (1, 0, 1). The summaries are shown below:

Constant Mean - ARCH Model Results

Dep. Variable:	None	**R-squared:**	0.000
Mean Model:	Constant Mean	**Adj. R-squared:**	0.000
Vol Model:	ARCH	**Log-Likelihood:**	1041.44
Distribution:	Normal	**AIC:**	-2076.88
Method:	Maximum Likelihood	**BIC:**	-2063.75
		No. Observations:	588
Date:	Fri, Dec 02 2022	**Df Residuals:**	587
Time:	14:58:35	**Df Model:**	1

Mean Model

	coef	std err	t	P>|t|	95.0% Conf. Int.
mu	-1.2874e-03	1.683e-03	-0.765	0.444	[-4.587e-03,2.012e-03]

Volatility Model

	coef	std err	t	P>|t|	95.0% Conf. Int.
omega	1.5230e-03	1.135e-04	13.424	4.368e-41	[1.301e-03,1.745e-03]
alpha[1]	0.1115	4.955e-02	2.250	2.448e-02	[1.435e-02, 0.209]

Constant Mean - EGARCH Model Results

Dep. Variable:	None	**R-squared:**	0.000
Mean Model:	Constant Mean	**Adj. R-squared:**	0.000
Vol Model:	EGARCH	**Log-Likelihood:**	1159.36
Distribution:	Normal	**AIC:**	-2310.73
Method:	Maximum Likelihood	**BIC:**	-2293.22
		No. Observations:	588
Date:	Fri, Dec 02 2022	**Df Residuals:**	587
Time:	18:04:25	**Df Model:**	1

Mean Model

	coef	std err	t	P>|t|	95.0% Conf. Int.
mu	4.3589e-04	7.135e-03	6.109e-02	0.951	[-1.355e-02,1.442e-02]

Volatility Model

	coef	std err	t	P>|t|	95.0% Conf. Int.
omega	-0.0604	1.609e-04	-375.705	0.000	[-6.076e-02,-6.013e-02]
alpha[1]	-0.0466	8.873e-03	-5.251	1.516e-07	[-6.398e-02,-2.920e-02]
beta[1]	0.9916	1.272e-09	7.796e+08	0.000	[0.992, 0.992]

Figure 6.5: Model Summaries for Thermal Electricity (Left) and New energy (Right)

Reviewing the result of thermal power data first, the p-value of the mean is 0.444 (< 0.05), which indicates that the residual plots have a mean of 0. And considering the p-value of the parameters, they are all non-zero at 5% significance level. Moreover, the estimated $\alpha = 0.1115 < 1$, which is consistent with the constraint of GARCH. It can be concluded that the model is not overfitted. Nevertheless, it can be found that the BIC is -2063.75, which is lower than its SARIMA model without ARCH fitting (-2046.413), although the decrease is slight. It is also reasonable after we visualize the conditional volatility diagram:

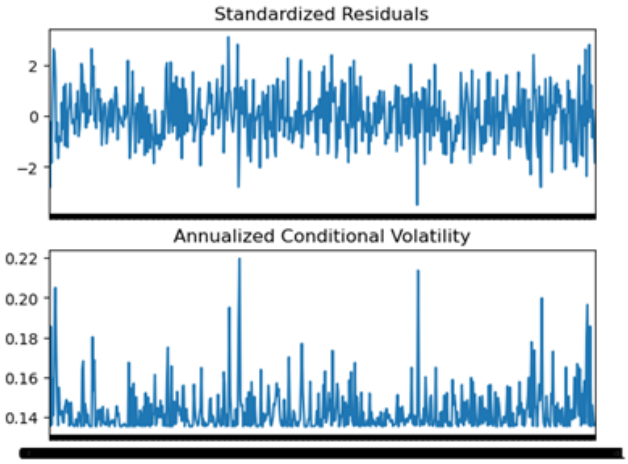

Figure 6.6: Thermal Electricity Annualized Conditional Volatility diagram

There are some extreme values that occurred in the 50 years time zone, but there are reasonable chances that these values were driven by specific events instead of structural changes in variance. Therefore, we can treat these cases as unexpected random factors and ignore them, which means that it is acceptable just to treat the residual plots of thermal power as White Noise (Choose the more straightforward result).

For the New energy power data, it is also the case that the mean can be inferred as 0, and all the parameters are statistically significant at 5%. In contrast, its BIC is -2293.22, a remarkable decline compared to -2231.710 in the SARIMA model. The conditional volatility diagram is as follows:

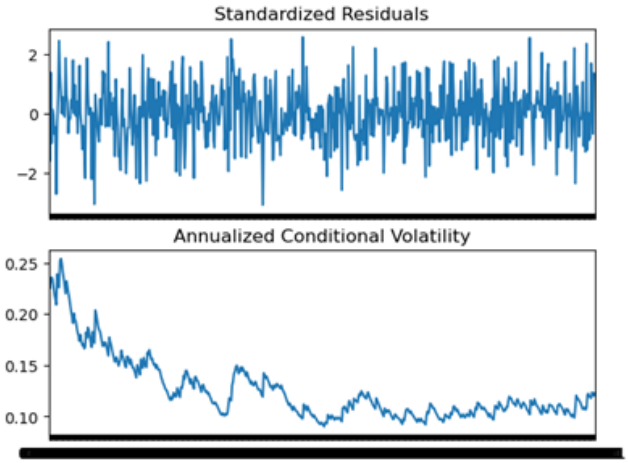

Figure 6.7: New Energy Annualized Conditional Volatility diagram

The annualized conditional volatility shows a clear exponential decreasing trend from around 0.23 to 0.13, which is the case of fitting the E-GARCH model. Therefore, we would like to consider the E-GARCH (1, 0, 1) to be the process of New energy data's residual:

$$ln(\sigma_{t^2}) = -0.0604 + 0.9916ln(\sigma_{t-1}^2) - 0.0466(|\varepsilon_{t-1}| - E(|\varepsilon_{t-1}|))a_{t-1}^2$$

By noticing the Standardized Residual plots, we can see that the variance shrinkage trend is eliminated. ACF and PACF plots also show that almost every lag's values are within the required boundary, with only some out lags that can be treated as randomly arise. The Q-Q plot and histogram are also consistent with the normal distribution with constant mean and variance. These all indicate that the model fitting is reasonable.

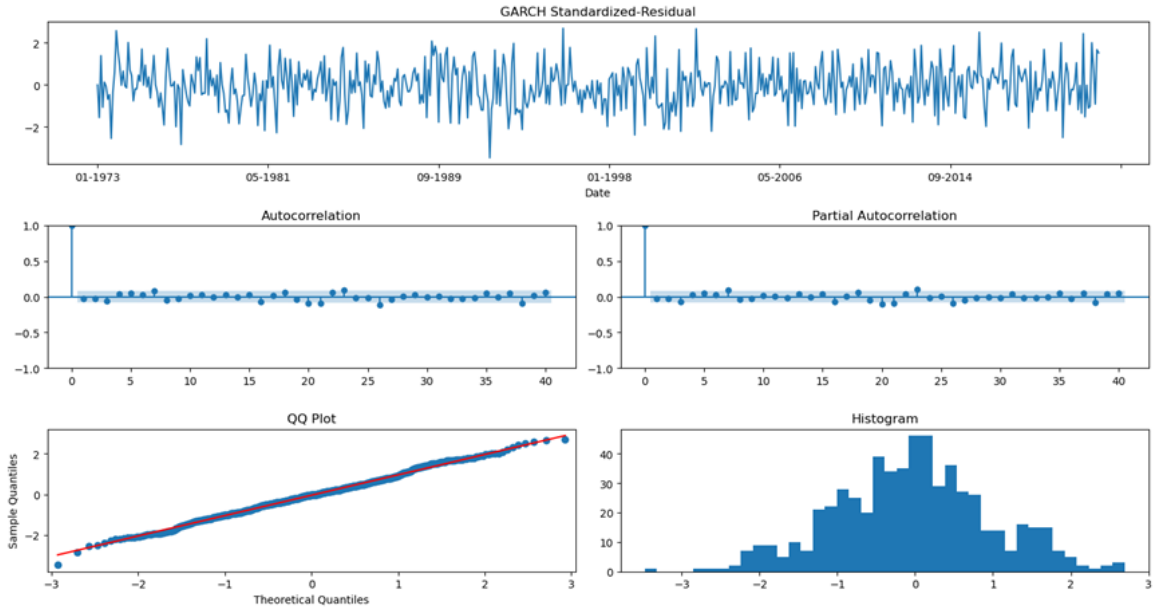

Figure 6.8: E-GARCH model checking for New energy

6.3 E-GARCH prediction and comparison

Eventually, we would like to predict the variance based on the E-GARCH model obtained above for the new energy data. Since python doesn't support the "analytical" method for E-GARCH model prediction, we will choose the "simulation" method and the "bootstrap" method to make a prediction. The former is generalized by repetition, while the latter is based on resampling. The prediction plots are shown below: (the solid line is the prediction result)

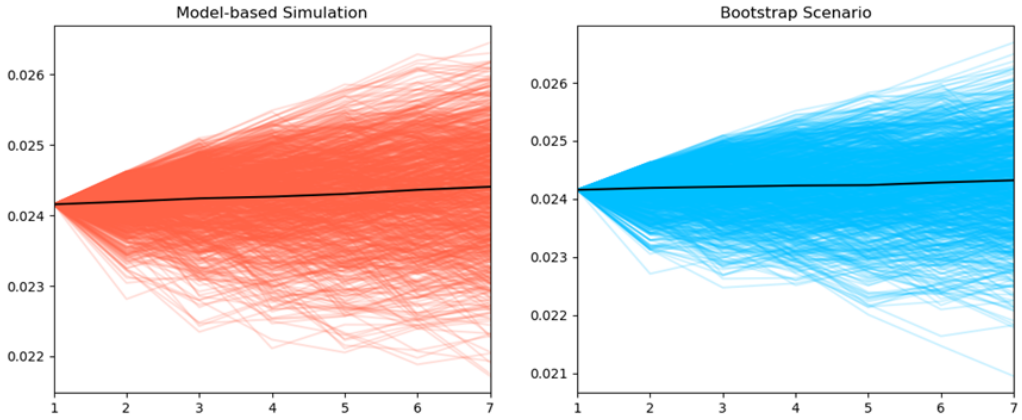

Figure 6.9: E-GARCH prediction based on Simulation (Left) and Bootstrap (Right) Methods

We can also compare the result to the MSE we obtained in the prediction part:

Table 12: Predicted MSE comparison

Lags	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Original	0.03444	0.04408	0.04944	0.05282	0.05513	0.05680	0.05809
E-GARCH, Simulation	0.02416	0.02420	0202424	0.02427	0.02431	0.02436	0.02441
E-GARCH, Bootstrap	0.02416	0.02419	0202421	0.02423	0.02424	0.02429	0.02432

Notice that the E-GARCH prediction variances by Simulation and Bootstrap are both less than the previous estimation, which is somehow more accurate as it doesn't consider much about the variance long ago.

Eventually, the log-transformed New energy data follows $SARAIMA(1,0,1) \times (1,1,1)_{12} - E-GARCH(1,0,1)$. This model also gives a better insight into variance prediction that could help in other residual analyses.

7 Conclusion

This study conducted a time series analysis of Thermal Electricity and New energy net generation for a total of 588 monthly periods from 1973-2021 in the US. The two data series were finally fitted with a $SARIMA(1,1,1) \times (1,0,1)_{12}$ model, using BIC as an indicator and the aid of images. Based on the parameter estimates fitted to the model, we forecast Thermal Electricity and New energy net generation for a total of seven months from January to July 2022, construct confidence intervals, and compare them to the real data to confirm the validity of our model. In order to better improve the prediction effect of the model, we have further discussion from two perspectives of prediction long-term prediction accuracy and prediction confidence interval. considering that the original SARIMA model is not suitable for long-term prediction, we fit the data again by LSTM deep learning model, which captures the long term trend to some extent. Moreover, after testing the possibility of fitting ARCH-type models based on data heteroskedasticity, we finally provide E-GARCH considerations for the residuals of New energy net generation and use Simulation and Bootstrap methods to make more accurate forecasts of MSE for the next 7 months. The above two methods provide a new perspective for future in-depth studies.

References

[1] C. F. Kutscher, J. B. Milford, and F. Kreith., *Principles of Sustainable Energy Systems, Third Edition (3rd edition.).* CRC Press, an imprint of Taylor and Francis, 2018. 1

[2] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997. 5.1

[3] C. Dyer, "Notes on lstms." 5.1.1

[4] K. H. R. Chan, Y. Yu, C. You, H. Qi, J. Wright, and Y. Ma, "Redunet: A white-box deep network from the principle of maximizing rate reduction," *Journal of Machine Learning Research*, vol. 23, no. 114, pp. 1–103, 2022. 5.2.2