2022-12-16

TEMPERATURE FORECAST

DATS 6313 Time Series Analysis and Modeling

Instructor: Reza Jafari

Author: Yuan Dang

TABLE OF CONTENTS

1. ABSTRACT

We always look at the weather forecast with cell-phone APP when we want to go out somewhere, as we need to decide what to wear. Therefore, it is important to know the next day or even next hour temperature.

2. DATA DESCRIPTION

A. Introduction

The original data contains several .csv files, one for each feature. I combined the datasets into one file, and thus we can extract weather information given any city name we want. In this case, I'll use weather information from 'Vancouver', but this project works for any given city.

The temperature data is based on hours, the goal of this project is to predict temperature of next few hours.

- **Source:** Kaggle(https://www.kaggle.com/datasets/selfishgene/historical-hourly-weather-data/discussion/56293?select=temperature.csv)
- **Timestamp:** Hourly measured from *2012-10-01 12:00* to *2017-11-30 00:00*
- **Size:** 45253 rows x 7 columns
- **Features:**
    - **Temperature(K):** target
    - Humidity(%)
    - Pressure(hPa)
    - Weather Description: 37 categories: [clear, light rain, clouds, mist, …]
    - Wind Direction(°)
    - Wind Speed(m/s)
    - 36 City: 27 US cities, 3 Canadian cities, 6 Israeli cities.(Vancouver in this case)
- **Goal:** Forecast temperature for future hours

Temperature is our dependent variable, which is the hourly recorded temperature in Vancouver in Kelvin, but we will convert it into Fahrenheit and use Fahrenheit as our unit in the whole project for convenience.

In multiple linear regression, our independent variables are "humidity", "pressure", "weather description", "wind direction", "wind speed".

In time series models, our independent variable is just time.

B. Pre-processing

i. Find Missing Values

I designed a heatmap to help us indicating the position of missing values, so we know exactly where on the timeline the data is missing.

Figure 2-1 missing values

Missing values:

	var	number of missing values
0	Vancouver_temp	795
1	humidity	1826
2	pressure	4234
3	weather	793
4	wind_direction	795
5	wind_speed	795

Figure 2-2 missing values table

The brown color is where the .isna() function returns True. Along with the table, we see that a lot of values are missing at the middle of timeline in 'pressure' and 'humidity'. And we miss the record of information at the end of timeline.

ii. Impute Missing Values

As for the non-recorded data, we can simply truncate them and this will not affect our dataset. But for the middle missing values, we can not simply drop them since we need a continuous time series data.

We can impute those missing values using backward interpolation, by which we impute the missing data with the next-hour's data. We are not doing a forward interpolation because there are missing values at the beginning of timeline.

C. Visualize Raw Data

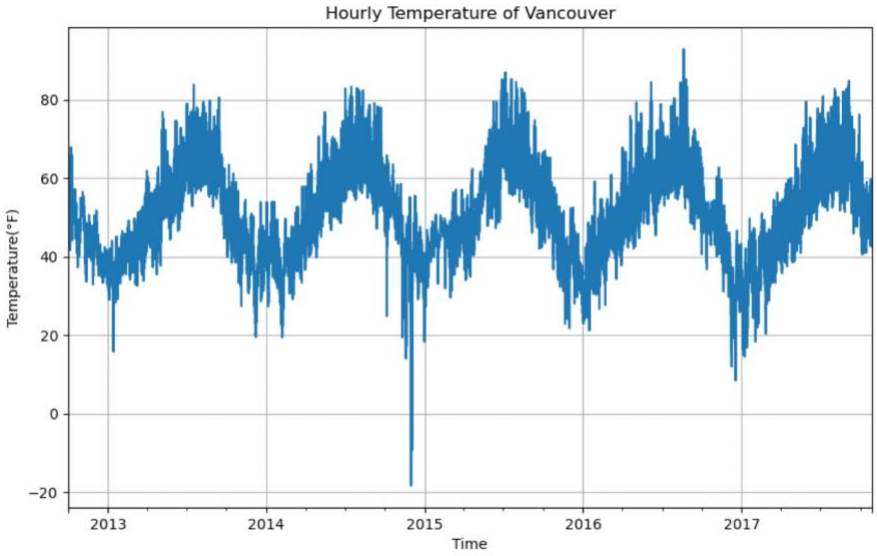


Figure 2-3 raw data

The raw data shows a strong yearly seasonality. Notice that for a hourly based dataset, the data is likely to be daily cycled as well. So we have multiple seasonality to deal with in this case.

Our data seems to be stationary, but we need to test on it.

D. ACF/PACF

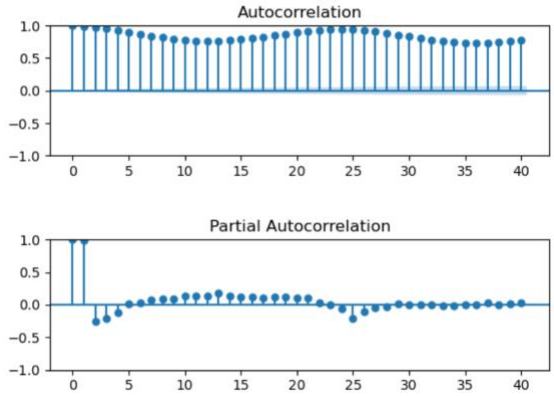


Figure 2-4 ACF/PACF of raw data

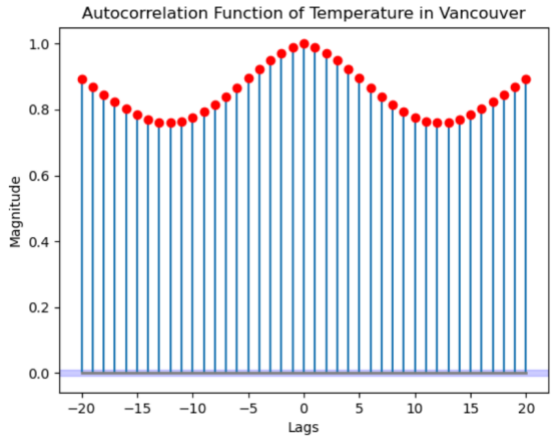


Figure 2-5 ACF of raw data

The ACF plot shows that temperature is not stationary since there is some seasonal patterns in the ACF.

E. Correlation Matrix

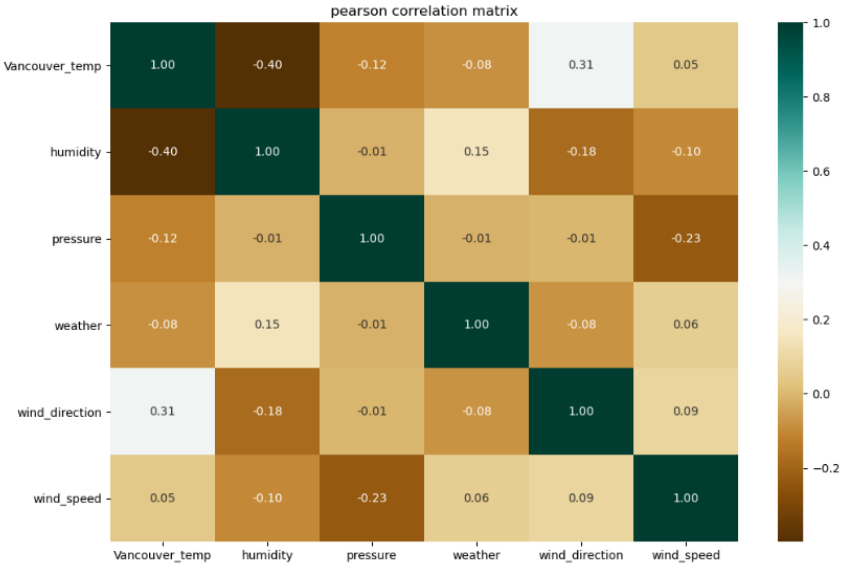


Figure 2-6 correlation matrix of features

A correlation matrix of all features in this dataset shows that features are not really correlated to each other. Although “humidity” has a -40% correlation with temperature, this value is acceptable.

3. STATIONARY

A. Check Stationarity

i. kpss-test

```
Results of KPSS Test:
Test Statistic            1.85279
p-value                   0.01000
LagsUsed                109.00000
Critical Value (10%)      0.34700
Critical Value (5%)       0.46300
Critical Value (2.5%)     0.57400
Critical Value (1%)       0.73900
dtype: float64
```

Figure 3-1 kpss test original data

The KPSS test result indicate temperature is non-stationary.

The KPSS test assumes data is stationary as null hypothesis. For a 95% confidence interval, the p-value 0.01 < 0.05, thus we reject the null hypothesis, and data is non-stationary.

ii. ADF-test

```
ADF Statistic: -5.510879
p-value: 0.000002
Critical Values:
    1%: -3.431
    5%: -2.862
    10%: -2.567
```

Figure 3-2 ADF test original data

It is interesting that the data passes the ADF test. ADF test assumes the data is non-stationary as null hypothesis, and the small p-value here means we reject the null hypothesis and temperature is stationary.

But in order to conclude stationarity, the data need to pass both tests. So our conclusion is the original data is non-stationary.

iii. Rolling mean and variance

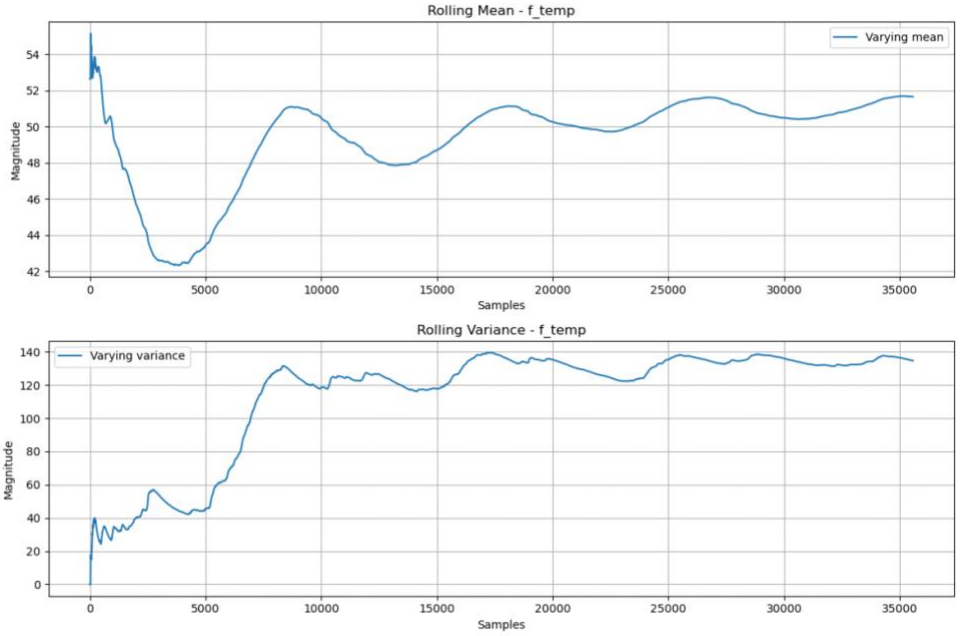


Figure 3-3 rolling mean and variance original data

The rolling mean is not quite constant, probably that's why we got different results from two tests.

B. 1st Order Seasonal Differencing

Remember our raw data is seasonal but not trended, so the non-stationarity may be caused by seasonality. So we can try a seasonal differencing with seasonal-period = 24 for hourly temperature.

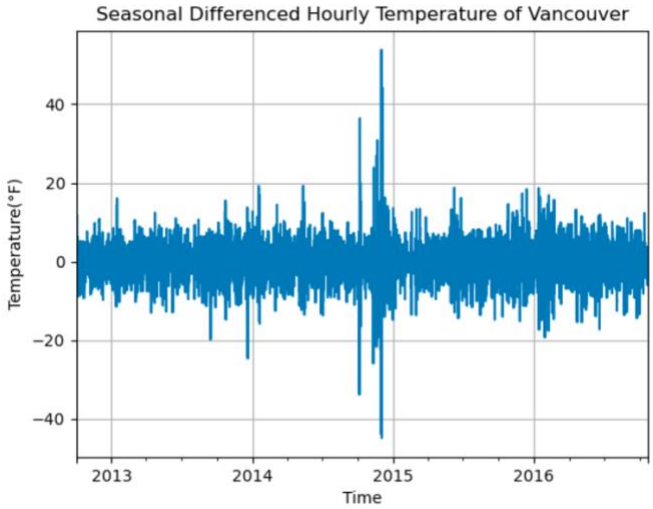


Figure 3-4 seasonal differenced data

This is how the seasonal differenced data looks like. We will now do some stationarity test on it.

i. kpss-test

```
Results of KPSS Test:
Test Statistic            0.012732
p-value                   0.100000
LagsUsed                 98.000000
Critical Value (10%)      0.347000
Critical Value (5%)       0.463000
Critical Value (2.5%)     0.574000
Critical Value (1%)       0.739000
dtype: float64
```

Figure 3-5 kpss test seasonal differenced data

Great, the data passes KPSS test this time.

ii. ADF-test

```
ADF Statistic: -26.072101
p-value: 0.000000
Critical Values:
    1%: -3.431
    5%: -2.862
    10%: -2.567
```

Figure 3-6 ADF test seasonal differenced data

It also passes the ADF test.

We now say that our seasonal differenced data is stationary.

iii. Rolling mean and variance

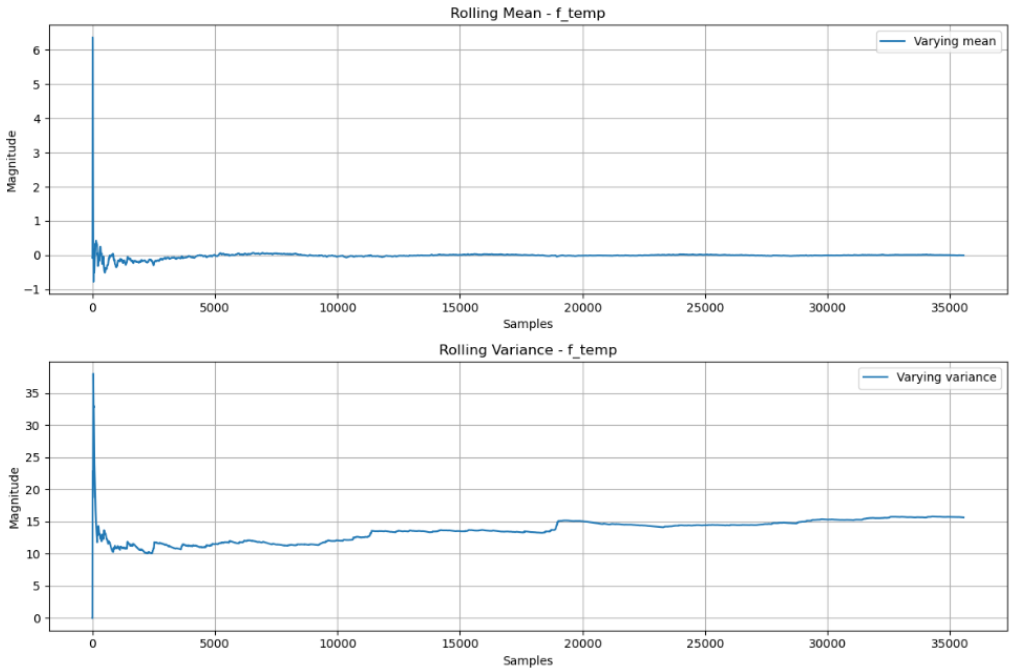


Figure 3-7 rolling mean and variance seasonal differenced data

The rolling mean and variance is obviously constant this time.

4. ECOMPOSITION

A. STL Decomposition

i. Original Data Decomposition

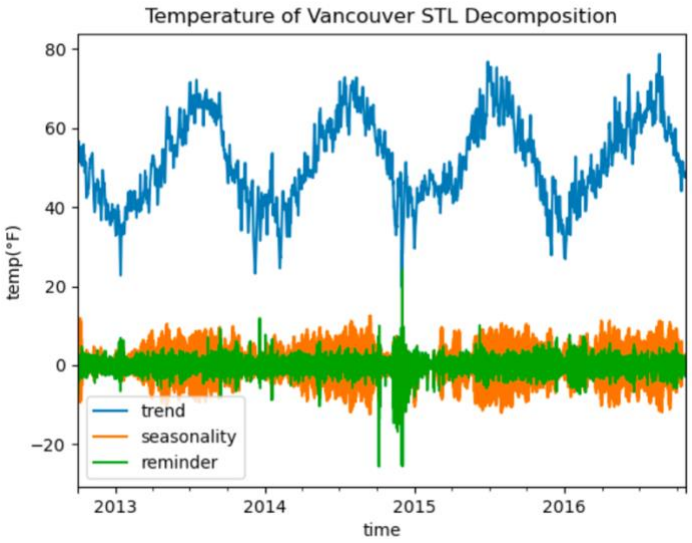


Figure 4-1 STL decomposition original data

1. *Strength of Trend and Seasonality*

```
The strength of trend for this dataset is 98.28%
The strength of seasonality for this dataset is 88.06%
```

The decomposition on original data gives a strong trend and seasonality. From the STL graph, we see that this trend is caused by seasonality.

ii. Seasonal Differenced Data Decomposition

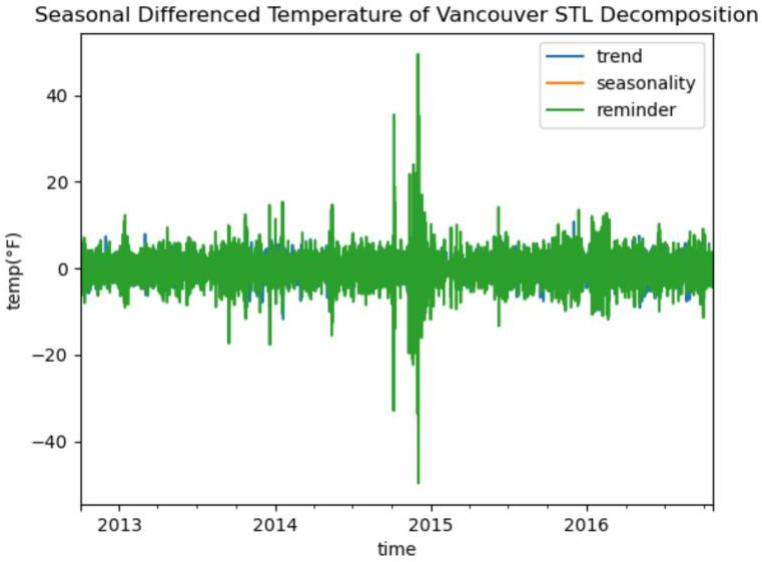


Figure 4-2 STL decomposition seasonal differenced data

1. *Strength of Trend and Seasonality*

```
strength of trend and seasonality after differencing
The strength of trend for this dataset is 62.52%
The strength of seasonality for this dataset is 11.08%
```

The seasonal differenced reduced both strength of trend and seasonality. But strength of trend is still strong. From STL decomposed graph, we see there is no seasonality in the data now. So we will do a non-seasonal 1st order differencing on the seasonal differenced data.

iii. Non-seasonal Differenced follow Seasonal Differenced Data

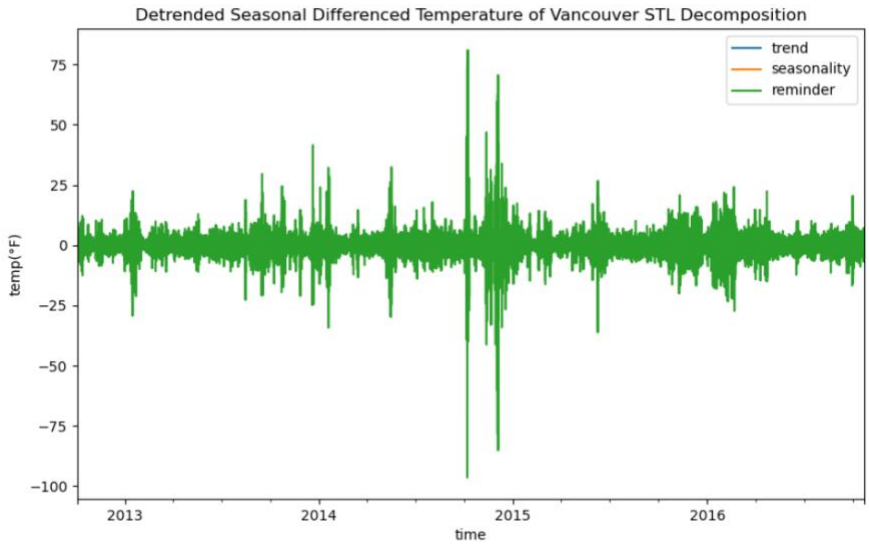


Figure 4-3 STL decomposition seasonal and non-seasonal differenced data

1. *Strength of Trend and Seasonality*

```
strength of trend and seasonality after detrended seasonal differencing
The strength of trend for this dataset is 1.45%
The strength of seasonality for this dataset is 3.79%
```

Now we have a clean data that has seasonality and trend removed by an 1st order seasonal differencing and 1st order non-seasonal differencing. Our final temperature data looks like:

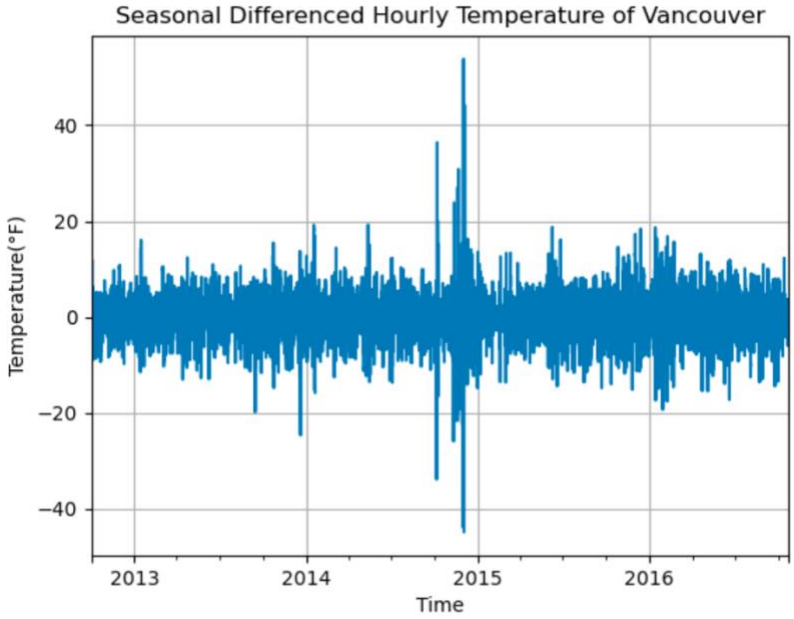


Figure 4-4 seasonal and non-seasonal differenced data

5. MODELING

Now we are ready for modeling, we will use 3:1 train test split on our dataset and has 80% train data and 20% test data. We have our original data, 1st order seasonal differenced data, 1st order non-seasonal follow seasonal differenced data. We will use different data to fit into different models depends on model features.

6. HOLT-WINTER METHOD

The holt-winter method is able to catch the trend and seasonality, it has two ways of decomposition: additive and multiplicative. So we will use original data to fit into this model and see if it can catch seasonality for our data.

A. Fit Train Set

We use the additive decomposition to fit our model since our temperature has negative values and multiplicative decomposition does not accept negative values to fit in.

B. One step Prediction

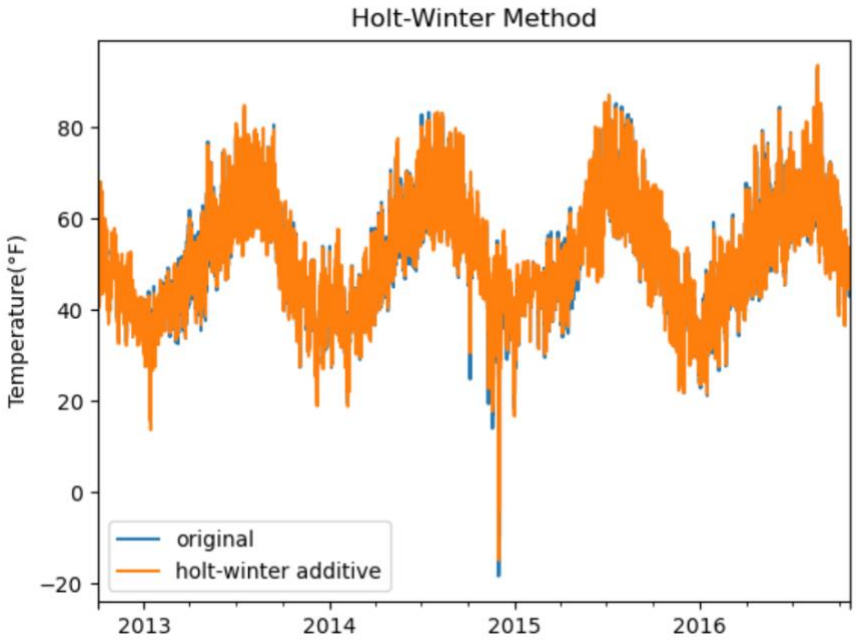


Figure 6-1 additive Holt-Winter training

It seems the model predict well on the train set.

C. H-step Forecasting

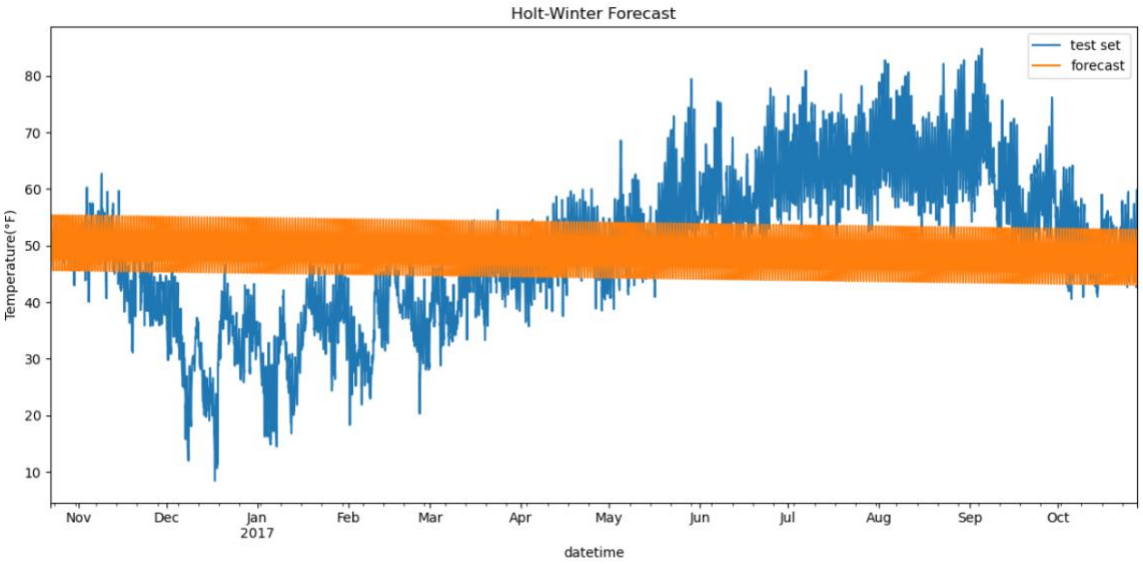


Figure 6-2 Holt-Winter forecasting

However, for the prediction on test set, the holt-winter method performs badly, as it does not catch the year seasonality. But our purpose is to predict the next few hours temperature, we do not need to look at the long term forecasting. Let's see how this model predicts for next 1000 hours.

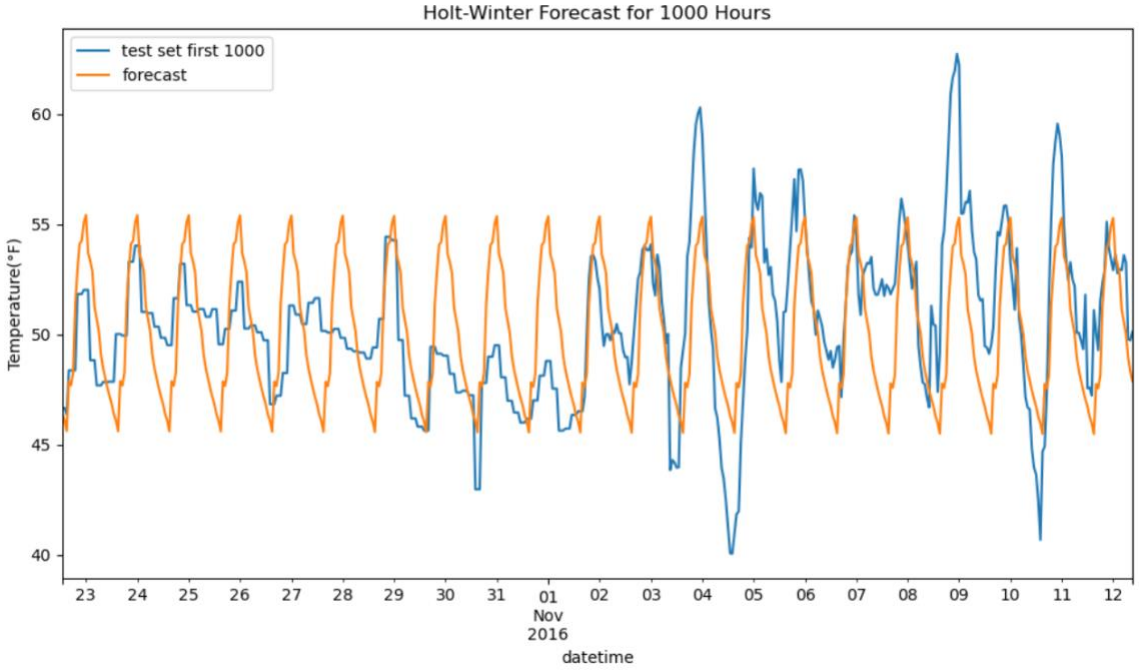


Figure 6-3 Holt-Winter forecasting first 1000

It seems that holt-winter method does catch the daily seasonality.

D. MSE

```
Mean square error of Holt-Winter method: 174.1898
```

The mean square error from forecast error is quite large. So this is not a good model.

7. BASE MODELS

In case that we are interested in how base model performs, such as average method, naïve method, drift method, SES method:

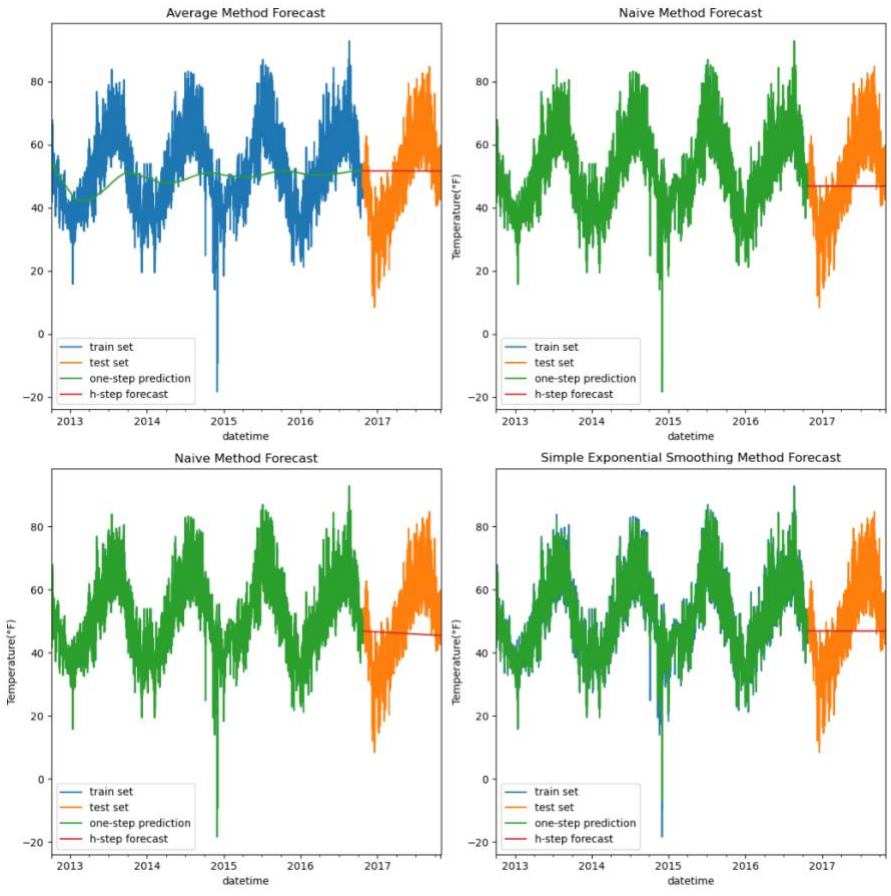


Figure 7-1 base models

8. MULTIPLE LINEAR REGRESSION

A. Feature Selection

i. Backwards Stepwise Selection

The backwards stepwise selection selects features based on their p-value. A p-value larger than 0.05 in a model is considered insignificant and should be removed.

features	humidity	pressure	weather	Wind direction	Wind speed	AIC	BIC	Adjusted R^2
keep or not	1	1	1	1	1	264296.2	264347.1	0.26328

The backwards selection says that we should keep all the five features. Since every feature is significant.

ii. VIF Selection

The VIF selection selects features based on their vif values. A vif value larger than 3 is considered insignificant and should be removed.

features	humidity	pressure	weather	Wind direction	Wind speed	AIC	BIC	Adjusted R^2
keep or not	1	1	1	1	1	264296.2	264347.1	0.26328

The VIF selection gives the exactly same result as backwards selection.

iii. SVD

```
singular values= [48186.18296387 45549.15227913 30872.40345426 28468.26262597
 24643.99867678]
```

The singular value from SVD decomposition shows that none of the five features should be removed since there is no singular value that is close to zero.

iv. Condition Number

```
condition number= 1.3983171180045006
```

Similarly, a condition number close to 1 shows that there is no collinearity between features.

v. PCA

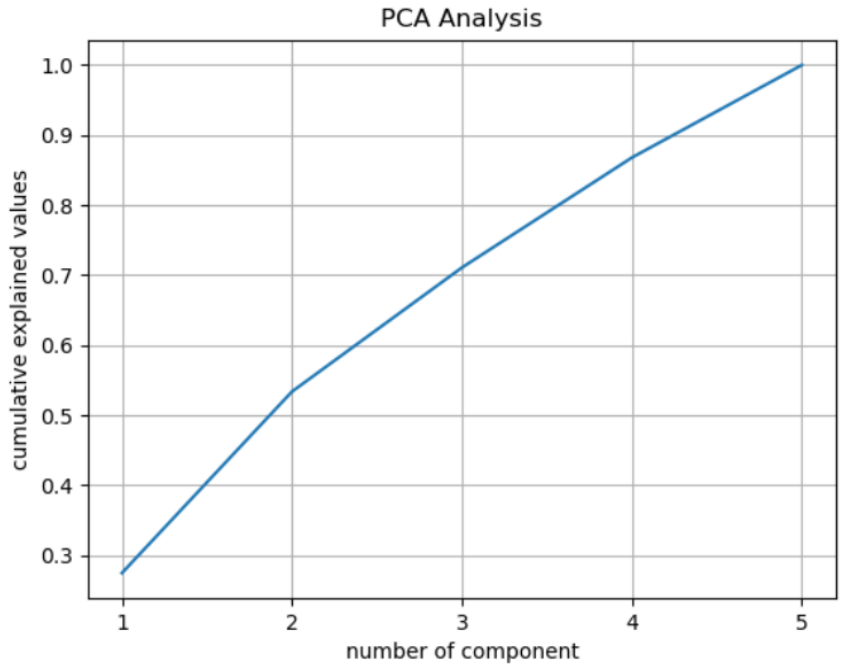


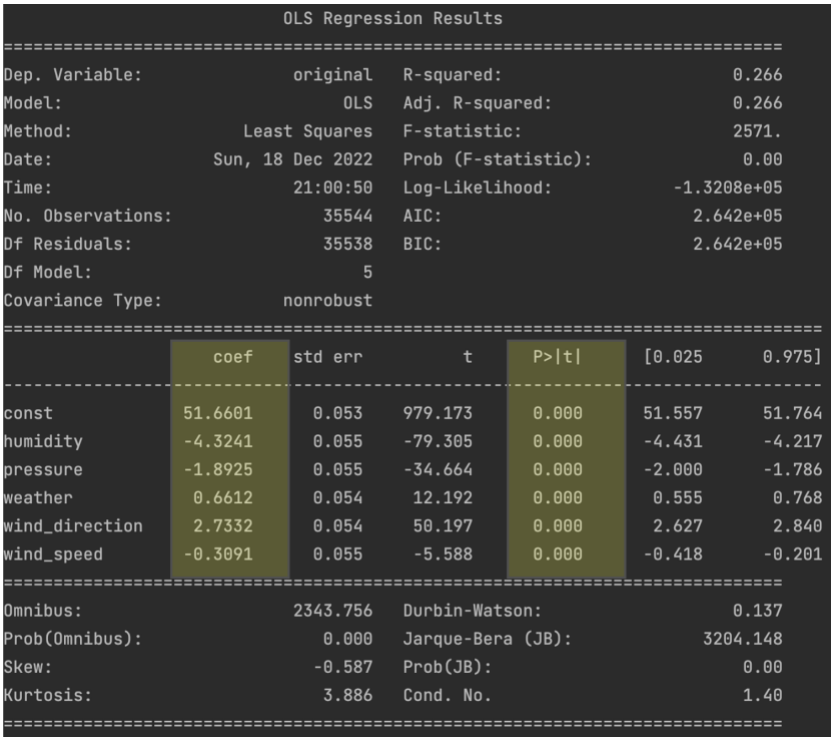
Figure 8-1 PCA

The PCA plot shows that in order to reach 90% of data explanation, we need to include 5 features.

vi. Conclusion

We need all the five features for our linear regression model.

B. Coefficients



OLS Regression Results

Dep. Variable:	original	R-squared:	0.266
Model:	OLS	Adj. R-squared:	0.266
Method:	Least Squares	F-statistic:	2571.
Date:	Sun, 18 Dec 2022	Prob (F-statistic):	0.00
Time:	21:00:50	Log-Likelihood:	-1.3208e+05
No. Observations:	35544	AIC:	2.642e+05
Df Residuals:	35538	BIC:	2.642e+05
Df Model:	5		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	51.6601	0.053	979.173	0.000	51.557	51.764
humidity	-4.3241	0.055	-79.305	0.000	-4.431	-4.217
pressure	-1.8925	0.055	-34.664	0.000	-2.000	-1.786
weather	0.6612	0.054	12.192	0.000	0.555	0.768
wind_direction	2.7332	0.054	50.197	0.000	2.627	2.840
wind_speed	-0.3091	0.055	-5.588	0.000	-0.418	-0.201

Omnibus:	2343.756	Durbin-Watson:	0.137
Prob(Omnibus):	0.000	Jarque-Bera (JB):	3204.148
Skew:	-0.587	Prob(JB):	0.00
Kurtosis:	3.886	Cond. No.	1.40

Figure 8-2 linear regression summary

The regression analysis gives us the coefficients:

Temp = 51.66 - 4.32humidity - 1.89pressure + 0.66weather + 2.73windDirection - 0.31windSpeed

C. Analysis

i. T test

```
t-test result for each coefficients:

const             0.0
humidity          0.0
pressure          0.0
weather           0.0
wind_direction    0.0
wind_speed        0.0
dtype: float64
```

As we see above, the t test results shows that all coefficients are significant with extremely small p-values.

ii. F test

```
f-test result for model:

(F-statistics: 2570.7331, p-value:0.0000)
```

The F test with a small p-value shows that, this is a significant model, it is significant to include all the features.

iii. Criterion Values

R-squared:	0.266
Adj. R-squared:	0.266
F-statistic:	2571.
Prob (F-statistic):	0.00
Log-Likelihood:	-1.3208e+05
AIC:	2.642e+05
BIC:	2.642e+05

Notice that the AIC/BIC values are extremely high, which may be a sign of overfitting.

The R square and adjusted R square is quite low, indicating that we only have 26.6% of variance explained by this linear regression model. This may not be a good model.

iv. Forecast

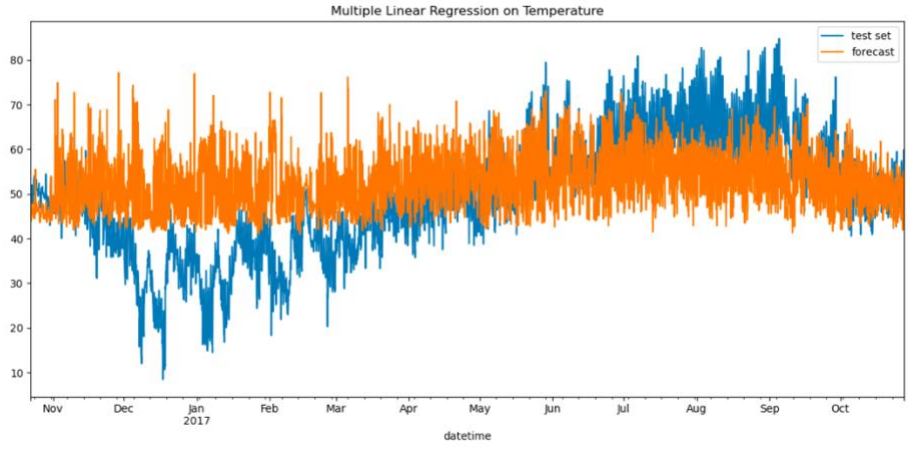


Figure 8-3 linear regression forecast

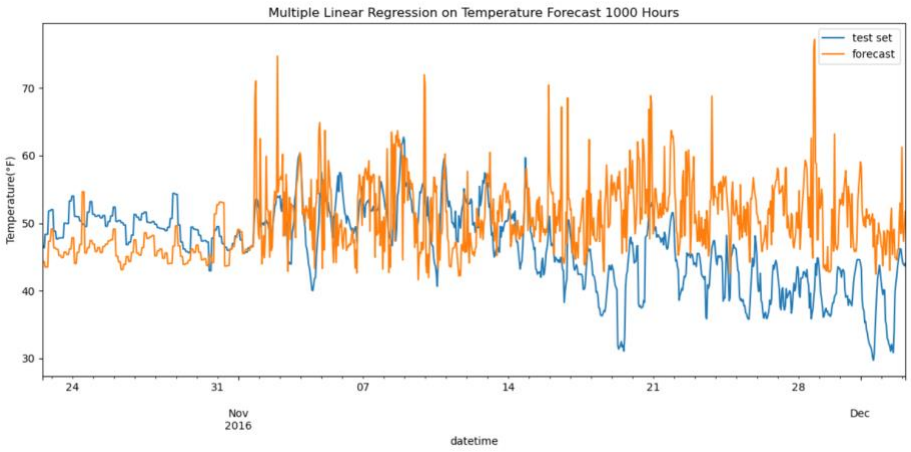


Figure 8-4 linear regression forecast first 1000

```
Mean squared error for linear regression = 152.34
```

The mean square error is smaller than MSE of holt-winter method, which is good. We can see that linear regression works better than Holt-Winter method in the long term. In the short term, from the graph, although this model catches the pattern in first few hours, the differences still exist.

v. ACF/PACF of Residual Error

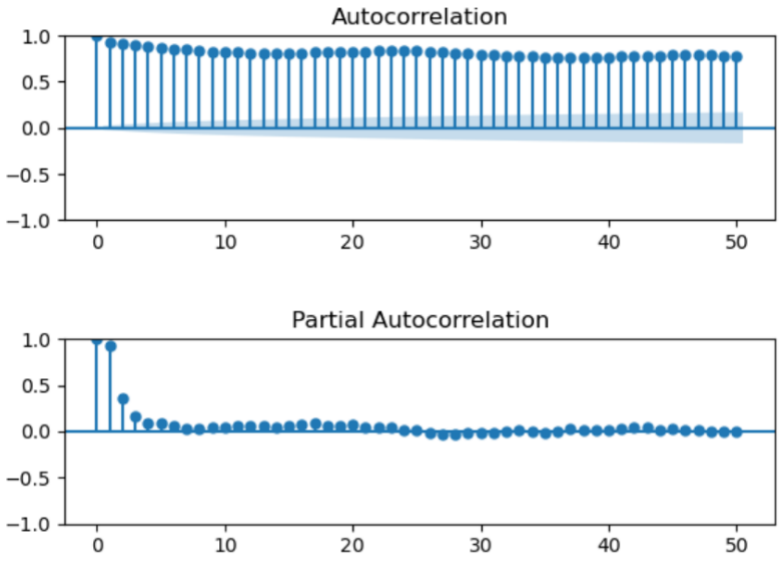


Figure 8-5 linear regression forecast error

The residual error is not white. Linear regression is not a good enough model for our data.

vi. Q value & Chi-square test

```
Q value of residual error with linear regression is 5383.57
```

```
Chi-square test on residual error from linear regression:
The error is not white
```

The Chi-square test re-confirms that our residual is not white.

vii. Variance and Mean of Residuals

```
mean of linear regression residual error is -3.11,
variance of linear regression residual error is 142.64
```

The mean of residual is ok, but the variance is quite big.

9. ARMA MODEL

We can now try some time series model like ARMA. Since ARMA model works only for stationary data, we will use our seasonal and non-seasonal differenced data for ARMA.

A. Order Determination

i. GPAC Table

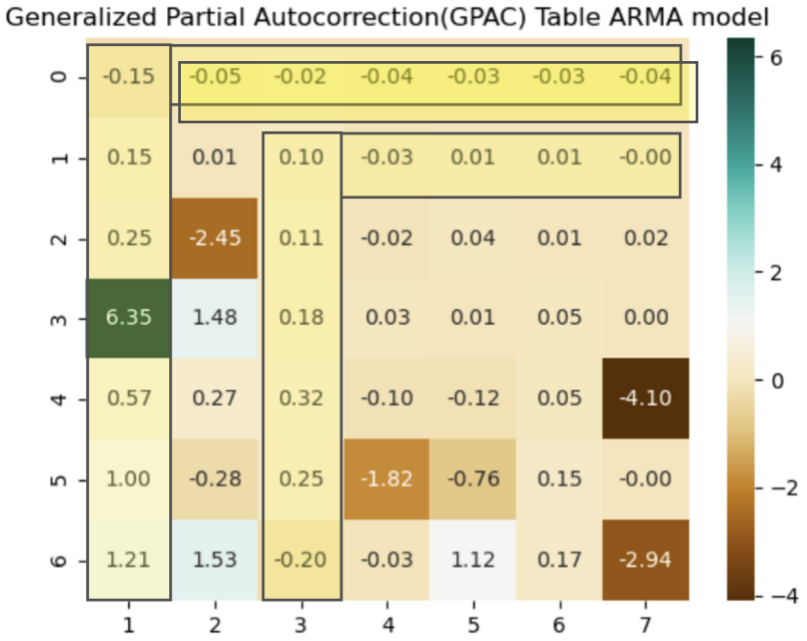


Figure 9-1 ARMA GPAC table

The GPAC table gives us some possible combination of AR and MA order na, nb.

Here we pick two combinations: (na=1, nb=0) and (na=3, nb=1)

ii. ACF/PACF

We now use PACF plot to re-confirm our order.

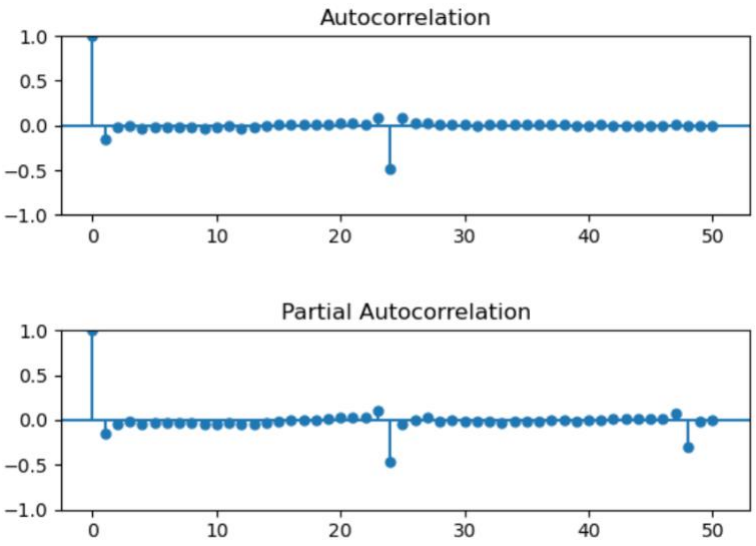


Figure 9-2 ARMA ACF/PACF

We notice that there is a multiplicative model pattern in the ACF/PACF, but for now, we will try a pure ARMA model first.

So for the non-seasonal part in ACF/PACF we could see a non-obvious cut-off in ACF and tail-off in PACF. The cut-off happens at lag=1. And this is a pattern of MA(1).

Therefore, we now have three possible combinations of na, nb. We will try estimate the parameter coefficients with all three combinations and compare the results.

B. Parameter Estimation

i. AR(1)

```
                           SARIMAX Results
==============================================================================
Dep. Variable:            differenced   No. Observations:                35544
Model:               SARIMAX(1, 0, 0)   Log Likelihood              -73227.946
Date:                Sun, 18 Dec 2022   AIC                         146459.892
Time:                        21:00:58   BIC                         146476.849
Sample:                  10-02-2012   HQIC                        146465.289
                       - 10-22-2016
Covariance Type:                 opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1         -0.1486      0.001   -135.208      0.000      -0.151      -0.146
sigma2         3.6058      0.005    720.719      0.000       3.596       3.616
===================================================================================
Ljung-Box (L1) (Q):                   1.65   Jarque-Bera (JB):           7024455.40
Prob(Q):                              0.20   Prob(JB):                         0.00
Heteroskedasticity (H):               1.01   Skew:                            -0.24
Prob(H) (two-sided):                  0.43   Kurtosis:                        71.87
===================================================================================
```

Figure 9-3 AR(1) estimation

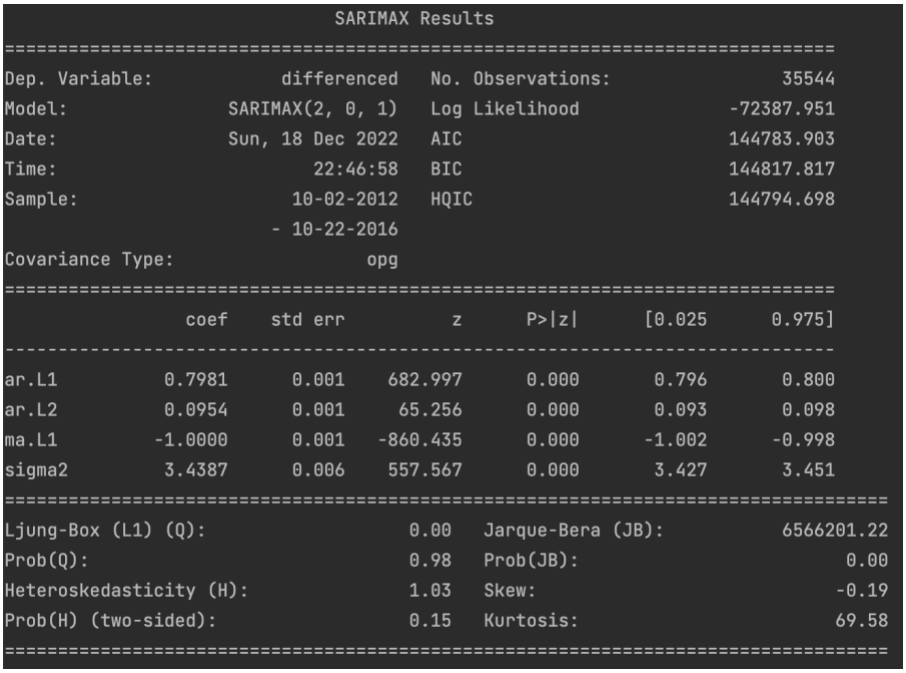
The parameter estimation from python package gives us model y(t) = -0.14y(t-1) + e(t), the confidence interval does not include zero, and it has extremely small p-value.

ii. ARMA(3, 1)/ARMA(2, 1)

```
                           SARIMAX Results
==============================================================================
Dep. Variable:            differenced   No. Observations:                35544
Model:               SARIMAX(3, 0, 1)   Log Likelihood              -72387.921
Date:                Sun, 18 Dec 2022   AIC                         144785.843
Time:                        22:23:06   BIC                         144828.235
Sample:                  10-02-2012   HQIC                        144799.336
                       - 10-22-2016
Covariance Type:                 opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.7982      0.001    683.227      0.000       0.796       0.800
ar.L2          0.0958      0.003     37.038      0.000       0.091       0.101
ar.L3         -0.0006      0.002     -0.239      0.811      -0.005       0.004
ma.L1         -1.0000      0.002   -568.908      0.000      -1.003      -0.997
sigma2         3.4386      0.008    457.384      0.000       3.424       3.453
===================================================================================
Ljung-Box (L1) (Q):                   0.00   Jarque-Bera (JB):           6569536.94
Prob(Q):                              1.00   Prob(JB):                         0.00
Heteroskedasticity (H):               1.03   Skew:                            -0.19
Prob(H) (two-sided):                  0.15   Kurtosis:                        69.60
===================================================================================
```

Figure 9-4 ARMA(3, 1) estimation

The parameter estimation shows a non-significant parameter at lag3, which has really large p-value. We need to reduce the AR order by one and rerun the estimation.

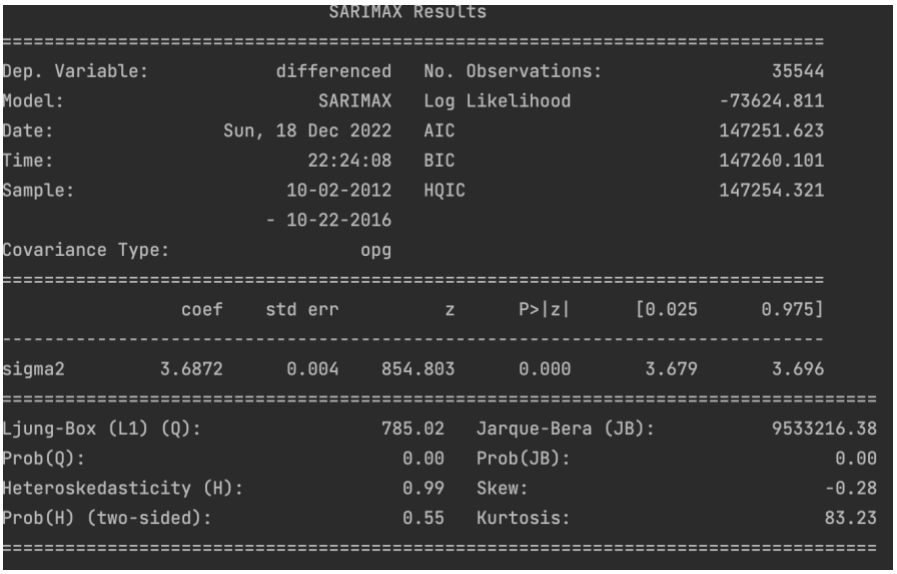


```
                           SARIMAX Results
==============================================================================
Dep. Variable:           differenced   No. Observations:                35544
Model:               SARIMAX(2, 0, 1)   Log Likelihood             -72387.951
Date:                Sun, 18 Dec 2022   AIC                         144783.903
Time:                        22:46:58   BIC                         144817.817
Sample:                  10-02-2012   HQIC                        144794.698
                       - 10-22-2016
Covariance Type:                 opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.7981      0.001    682.997      0.000       0.796       0.800
ar.L2          0.0954      0.001     65.256      0.000       0.093       0.098
ma.L1         -1.0000      0.001   -860.435      0.000      -1.002      -0.998
sigma2         3.4387      0.006    557.567      0.000       3.427       3.451
===================================================================================
Ljung-Box (L1) (Q):                   0.00   Jarque-Bera (JB):           6566201.22
Prob(Q):                              0.98   Prob(JB):                         0.00
Heteroskedasticity (H):               1.03   Skew:                            -0.19
Prob(H) (two-sided):                  0.15   Kurtosis:                        69.58
===================================================================================
```

Figure 9-5 ARMA(2, 1) estimation

Now we have ARMA(2, 1), and the coefficients seems good. The confidence interval now does not include zero, and it has extremely small p-value.

iii. MA(1)



```
                           SARIMAX Results
==============================================================================
Dep. Variable:           differenced   No. Observations:                35544
Model:                       SARIMAX   Log Likelihood             -73624.811
Date:                Sun, 18 Dec 2022   AIC                         147251.623
Time:                        22:24:08   BIC                         147260.101
Sample:                  10-02-2012   HQIC                        147254.321
                       - 10-22-2016
Covariance Type:                 opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
sigma2         3.6872      0.004    854.803      0.000       3.679       3.696
===================================================================================
Ljung-Box (L1) (Q):                 785.02   Jarque-Bera (JB):           9533216.38
Prob(Q):                              0.00   Prob(JB):                         0.00
Heteroskedasticity (H):               0.99   Skew:                            -0.28
Prob(H) (two-sided):                  0.55   Kurtosis:                        83.23
===================================================================================
```

Figure 9-6 MA(1) estimation

The parameter estimation gives us a coefficient of sigma2, and its p-value is extremely small.

iv. Conclusion

Since the coefficients are all significant now, we want to choose model that has the lowest order and minimum AIC/BIC values. So our final model is AR(1): y(t) = -0.14y(t-1) + e(t).

C. One-step Prediction

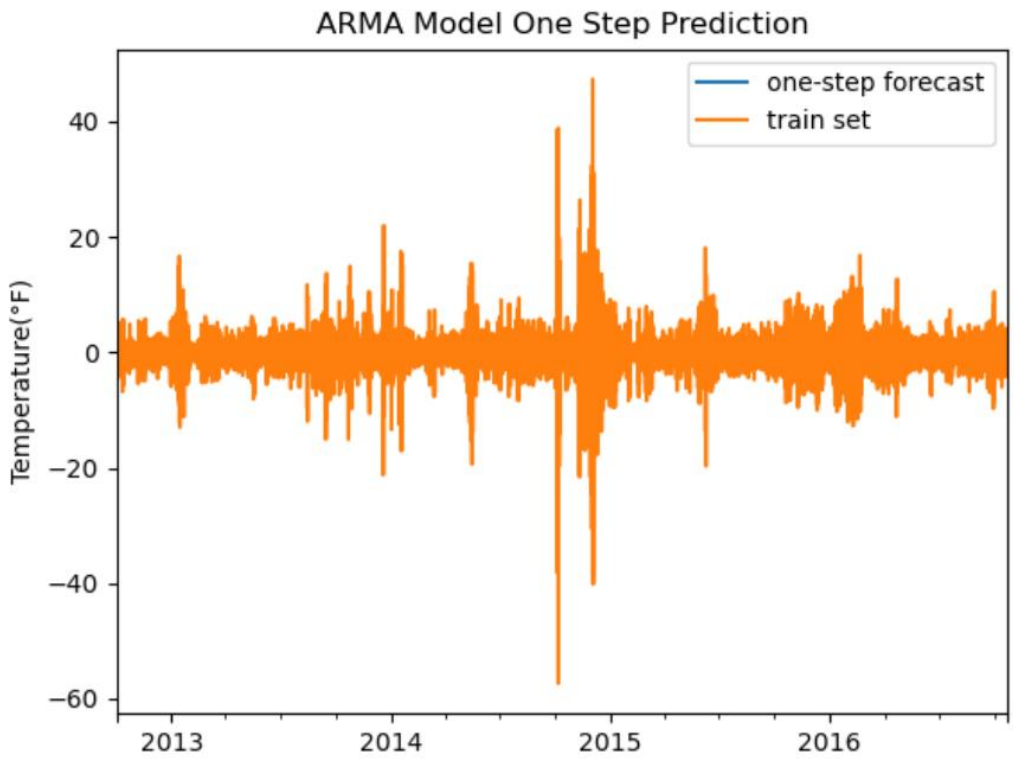


Figure 9-7 ARMA 1-step prediction

i. Residual Error ACF

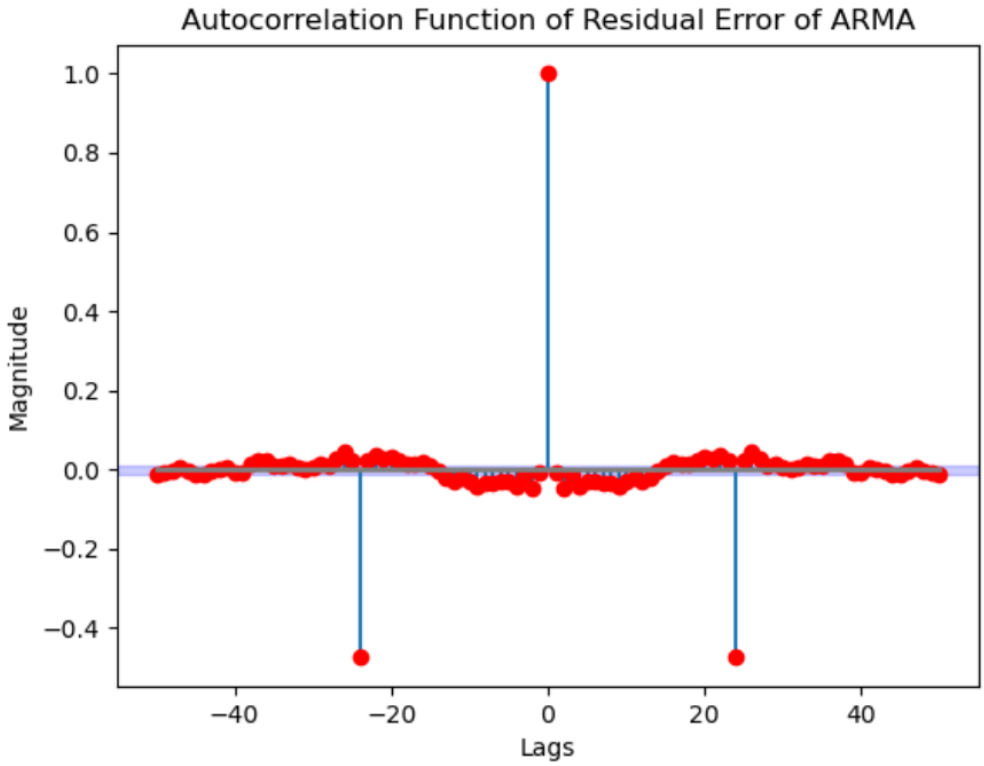


Figure 9-8 ARMA prediction error

The residual error is mostly white, except there exists additional seasonal pattern at lag24.

ii. Residual Error Q-value& Chi-square test

```
Q value for ARMA residual error: 7.201358950935142
************************************************
Chi-square test on residual error from ARMA:
The error is white
```

The test tells us the residual error is white.

D. H-step Prediction

Since we apply ARMA model on the differenced data, we need to invert the differencing on our h-step forecasted data.

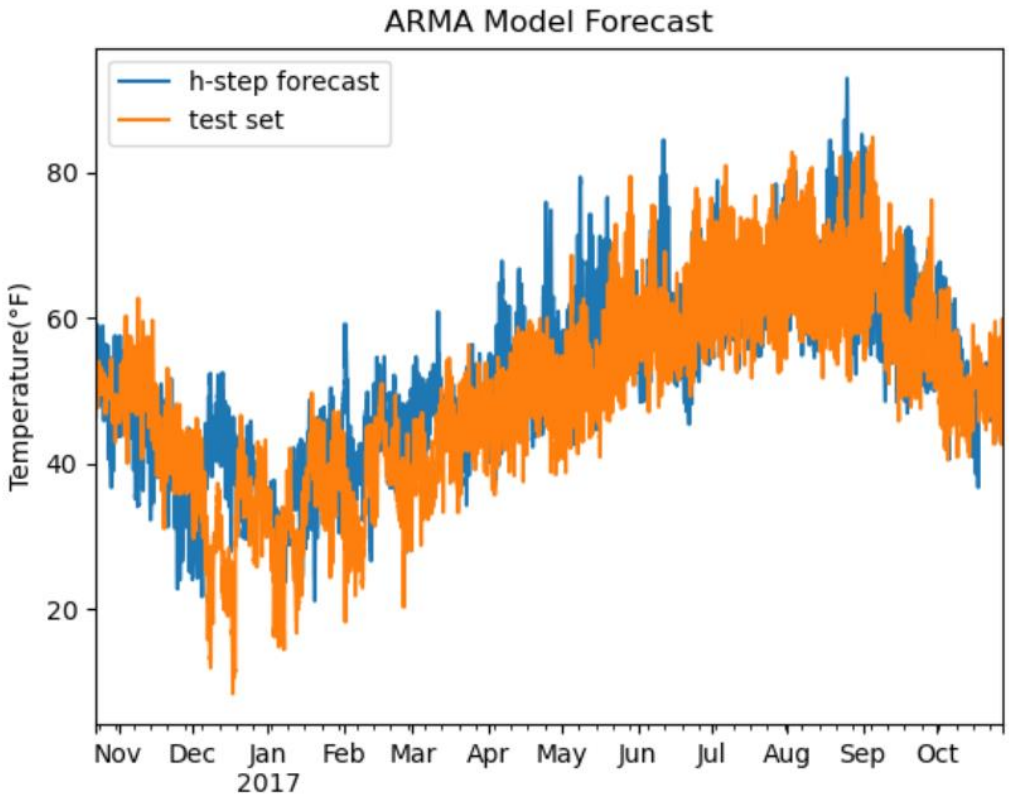


Figure 9-9 ARMA h-step prediction

We can see that the ARMA model forecasting performs better than the previous models. However, we still need to test the forecast error.

i. Forecast Error ACF

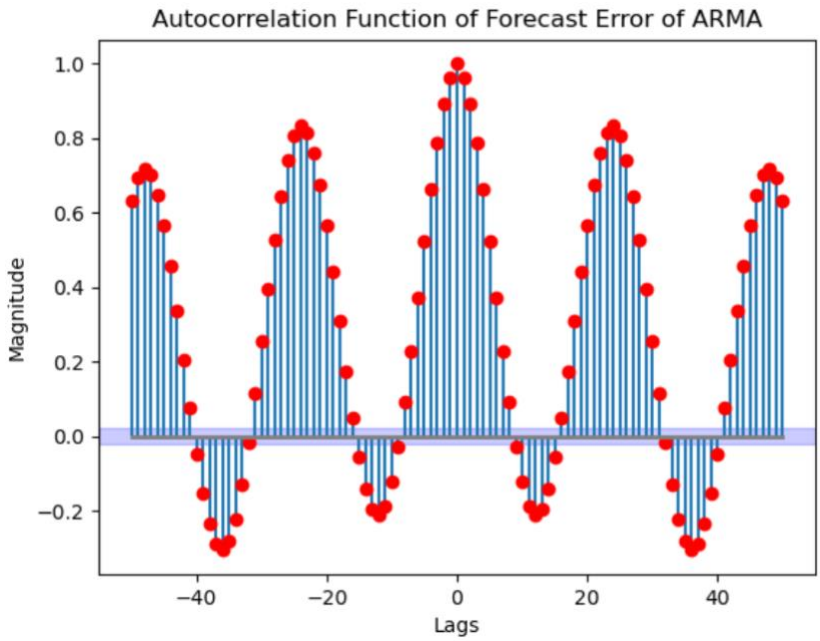


Figure 9-10 ARMA forecast error

Clearly, the forecast error is not white, there is some seasonal pattern that is not catched by ARMA model, this is the case where we need a SARIMA model.

ii. Forecast Error Q-value& Chi-square test

```
**************************************************
Q value for ARMA forecast error: 3540.29783627881
**************************************************
Chi-square test on forcast error from ARMA:
The error is not white
```

iii. MSE

```
Mean squared error = 119.33
```

10. SARIMA MODEL

A. Order Determination

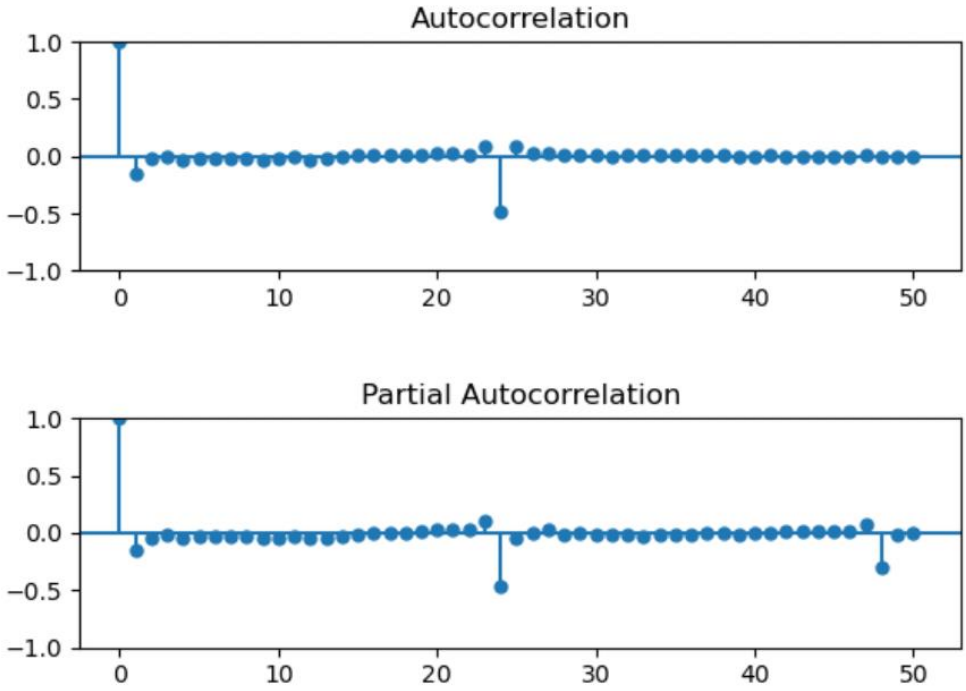


Figure 10-1 SARIMA ACF/PACF

We use the previous ACF/PACF to determine the SARIMA order. We can see a seasonal cut-off in ACF at lag24 and seasonal tail-off in PACF at lag24/48. This indicates we have seasonal MA(1)$_{24}$. For son-seasonal part, we have tail-off in PACF and cut-off in ACF both at lag1. So this gives us MA(1). Therefore we have a multiplicative model: MA(1)XMA(1)$_{24}$.

B. Parameter Estimation

SARIMAX Results

Dep. Variable:	differenced	No. Observations:	35544
Model:	SARIMAX(0, 0, 1)x(0, 0, 1, 24)	Log Likelihood	-63910.481
Date:	Sun, 18 Dec 2022	AIC	127826.963
Time:	23:52:32	BIC	127852.398
Sample:	10-02-2012	HQIC	127835.059
	- 10-22-2016		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ma.L1	-0.0831	0.001	-72.829	0.000	-0.085	-0.081
ma.S.L24	-0.9057	0.001	-729.047	0.000	-0.908	-0.903
sigma2	2.1321	0.003	744.478	0.000	2.126	2.138

Ljung-Box (L1) (Q):	0.06	Jarque-Bera (JB):	8803767.25
Prob(Q):	0.80	Prob(JB):	0.00
Heteroskedasticity (H):	1.07	Skew:	0.25
Prob(H) (two-sided):	0.00	Kurtosis:	80.10

Figure 10-2 SARIMA estimation

From the estimation, we get significant coefficients, and it gives the model y(t) = e(t) - 0.083e(t-1) – 0.91e(t-24).

C. H-step Forecasting

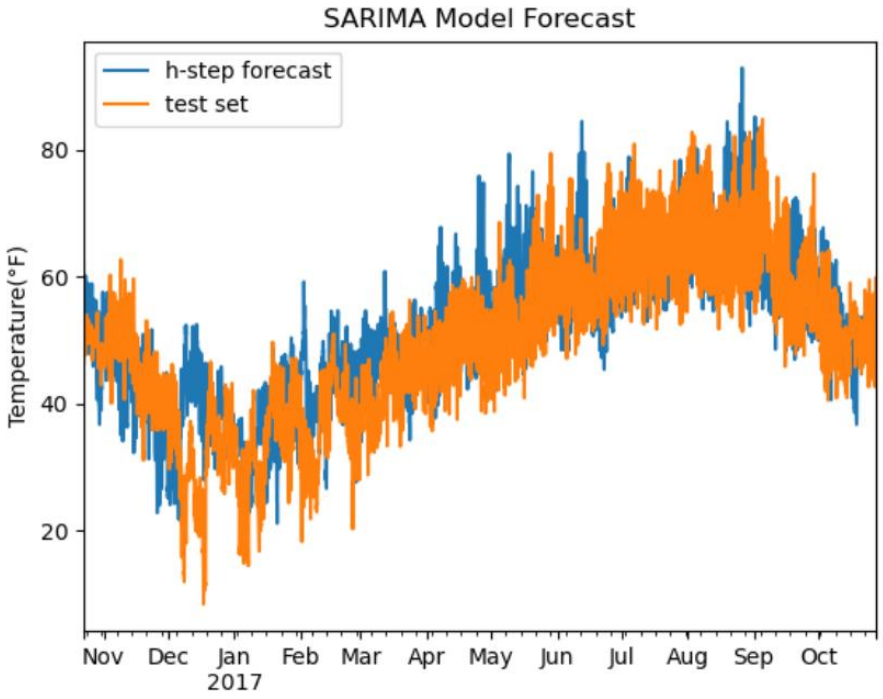


Figure 10-3 SARIMA forecast

We can see that the ARMA model forecasting performs better than the previous models. However, we still need to test the forecast error.

D. Forecast Error

i. Forecast Error ACF

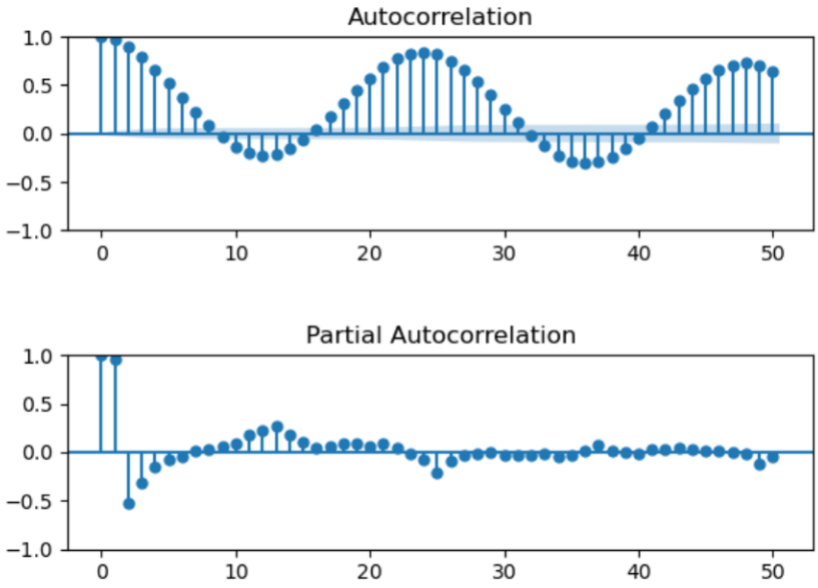


Figure 10-4 SARIMA forecast error

The ACF of forecast error looks exactly the same as previous ARMA model. Since we previous catched all seasonality detected by STL decomposition. We probably need other method to catch complicated seasonality.

ii. Forecast Error Q-value& Chi-square test

```
Q value for SARIMA forecast error: 26646.44602989614
************************************************
Chi-square test on forecast error from SARIMA:
The error is not white
************************************************
```

iii. MSE

```
Mean squared error = 118.71
```

11. MODEL SELECTION

A. Compare MSE/AIC

models	Holt-Winter	Multiple Linear Regression	ARMA	SARIMA
MSE	174.1898	152.34	119.33	118.71
AIC		264,200	146,459	127,852

So we pick the lowest MSE for modeling, and the final model is ARMA model, AR(1). However, we need to add our differencing to the model. Remember we did 1st order seasonal differencing with m=24 and 1st order non-seasonal differencing.

Therefore, our final model should be **ARIMA(1, 1, 0)x SARIMA(0, 1, 0)$_{24}$**

12. CONCLUSION

The final model performs good on training, but really bad on testing, this could be caused by overfitting, which means the model is biased. And we also need to catch the complex seasonality. In our case, we have multiple seasonality: yearly, daily and possibly weekly as well.

13. REFERENCE

Rob J Hyndman and George Athanasopoulos, *FORECASTING: PRINCIPLE AND PRACTICE 2nd 11.1 Complex Seasonality*

https://otexts.com/fpp2/complexseasonality.html

Benjamin Etienne, *Time Series in Python — Part 2: Dealing with seasonal data*

https://towardsdatascience.com/time-series-in-python-part-2-dealing-with-seasonal-data-397a65b74051

Jim Frost, *When Do You Need to Standardize the Variables in a Regression Model?*

https://statisticsbyjim.com/regression/standardize-variables-regression/

Etqad Khan, *Python Code on Holt-Winters Forecasting*

https://medium.com/analytics-vidhya/python-code-on-holt-winters-forecasting-3843808a9873