Time Series: Forecasting Spatial-Temporal Windspeed Project

Ying Gao (yg2804)

1. Introduction

The forecasted windspeed becomes important in the Wind-Energy Marketplace for setting up future energy related deals. The goal of this project is to forecast the windspeed in the next six months for data covered locations based on the historical observations.

Our data includes monthly spatial-temporal observations for windspeed from January,1979 to December, 2018, which is computed by taking the average of the measurements around the last 30 days. The final dataset consists of 480 cases (480 months) of observations in total without missing values. Each column in the dataset represents a single site and contains the windspeed data for that site. These sites are mainly located over Texas, New Mexico and Oklahoma. Here we assume that they are independent of each other, which means they can be separately analyzed and fitted in the model.

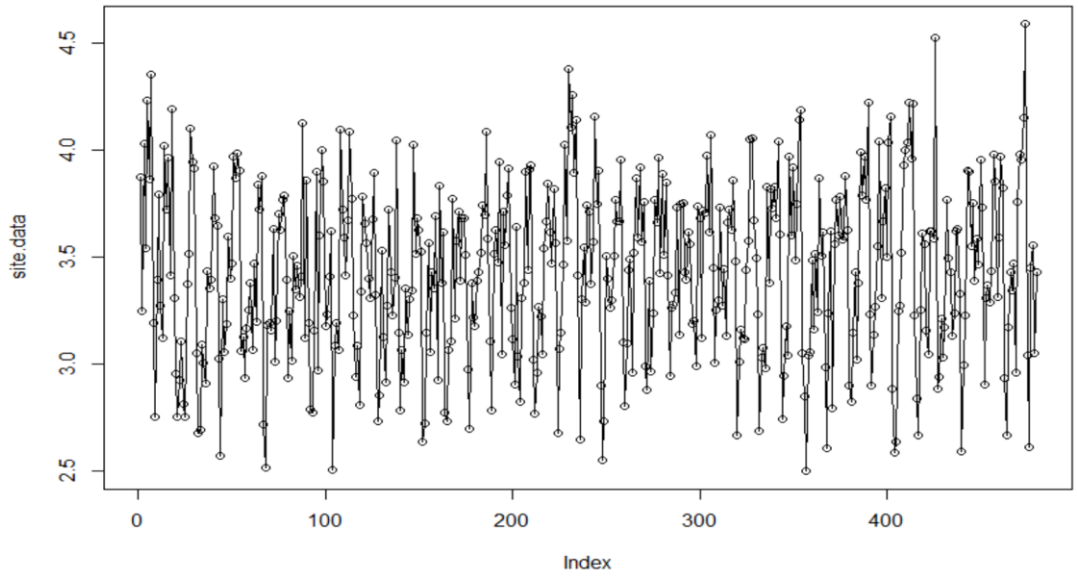

FIGURE 1.1. *Time series plot for site X1 over 480 months as index, presenting monthly average windspeed, connecting with lines.*

Figure 1.1 suggests that windspeed data has some seasonality which needs to be noted when we explore the data and build the model in the future. If we look at more specifically the yearly change and data spread using Figure 1.2., it clearly indicates that the monthly average windspeed follows similar a trend every year. It gently kept increasing from January to June, had a drop during July and August and grew back after September.

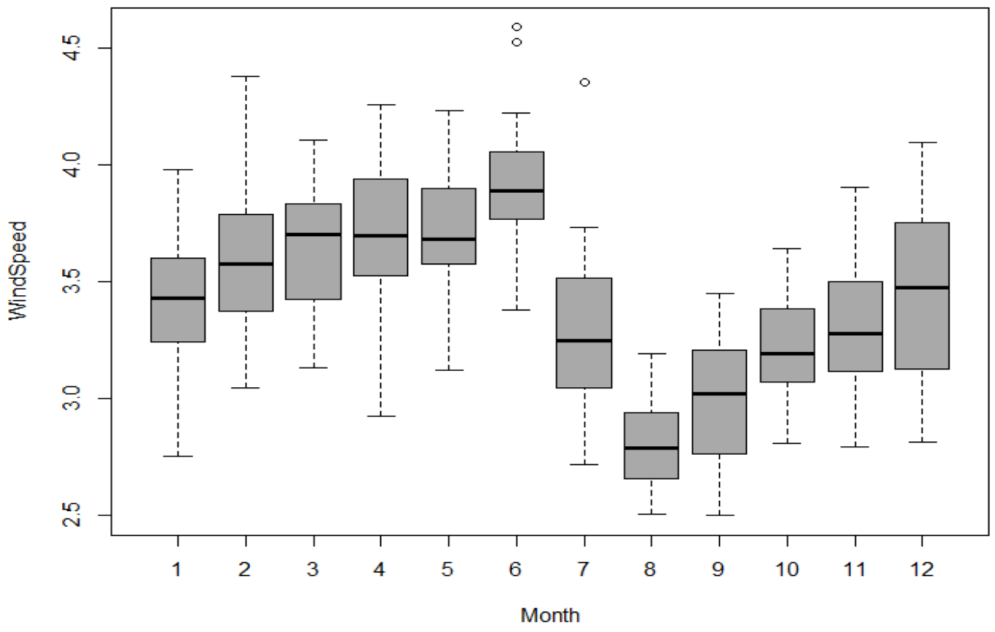

FIGURE 1.2. *Boxplot of average windspeed for the particular month over years. For example, the first boxplot displays the distribution of average windspeed in January from 1979 to 2018.*

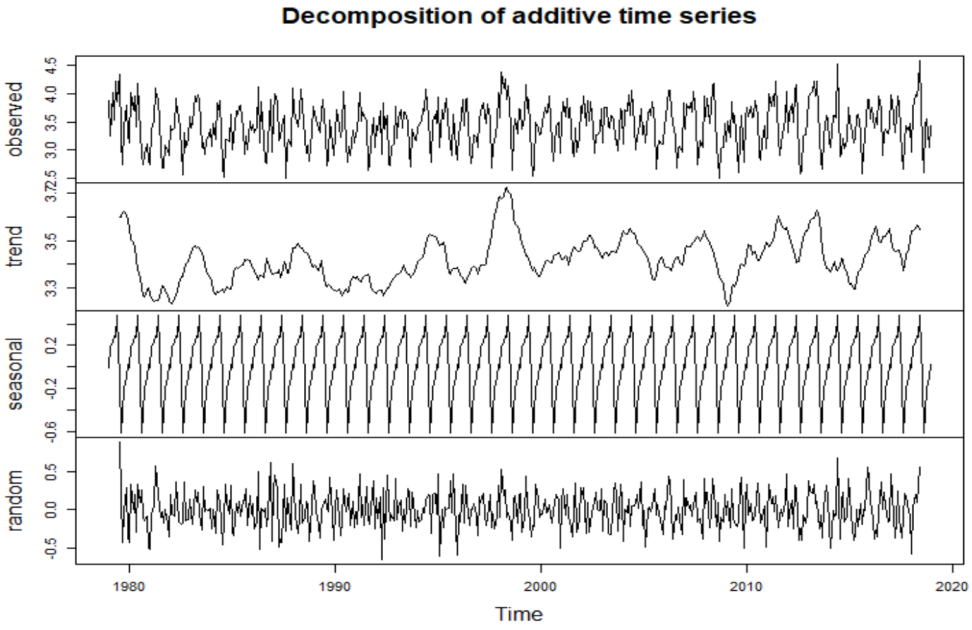

FIGURE 1.3. *Decomposition of additive time series plot, including overserved data plot, trend, seasonality, noise components.*

Time series data may have additive components. Therefore, to further explore the characteristics of the data and prepare for the knowledge before building model, decomposition plot in Figure 1.3. helps provide a better understanding of data trend, seasonality, and residuals.

2. Stationary

Based on Figure 1.1. and Figure 1.3., we can briefly summarize the pattern of the data. However, most time series models depend on the assumption of stationary data which could be indicated by the statistical properties of it not changing over time, so the stationary here need to be tested. The result of Augmented Dickey-Fuller (ADF) test provides p-value is 0.01 < 0.05 and then we can reject the null hypothesis and conclude that there is significant evidence showing that the data is stationary. Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test is also applied. The KPSS level is 0.1626 and p-value is 0.1 > 0.05. This suggests that we would not reject the null hypothesis of trend stationary, which means the data is stationary enough.

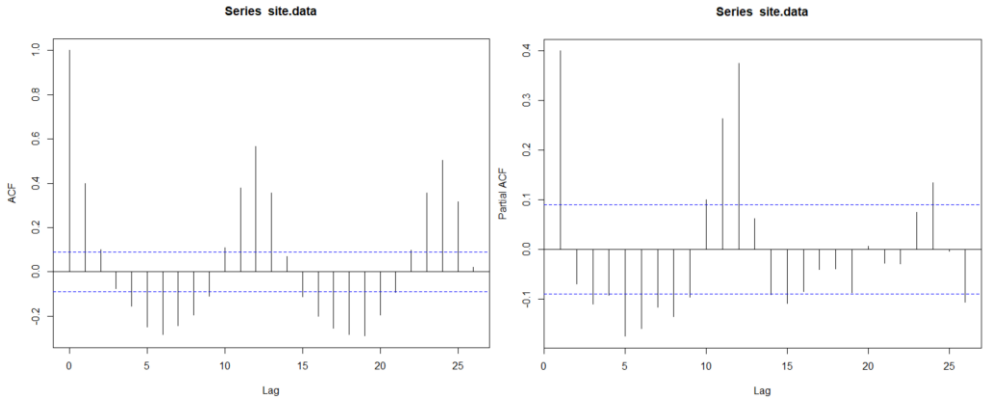

FIGURE 2.1. *The ACF and PACF plots of the original windspeed data of site X1.*

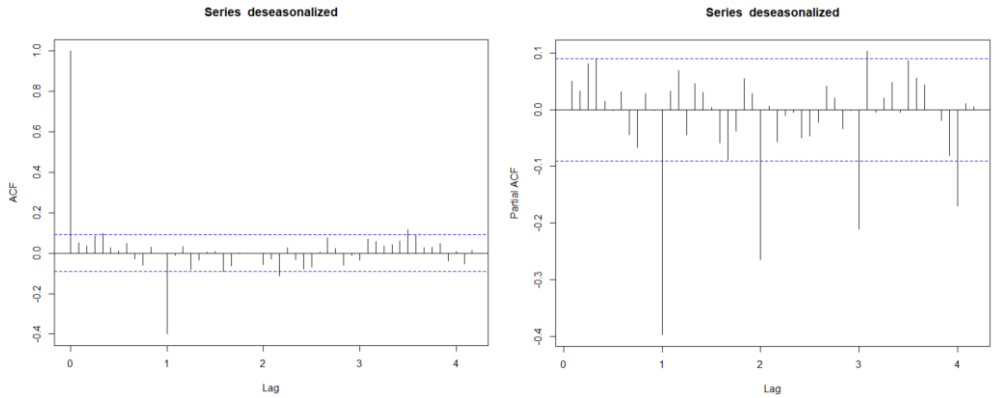

FIGURE 2.2. *The ACF and PACF plots of the de-seasonalized data of site X1.*

The ACF of PACF plots in Figure 2.1. have the observable seasonality on it. After de-seasonalizing (Figure 2.2.), ACF decreases sharply after the lag=1 but PACF does not decrease exponentially, so MA model and ARMA model is not appropriate here. Since PACF does not cut off at some lag=p, AR model is also not sufficient to be used. Hence, ARIMA model would possibly fit.

More specifically, seasonal ARIMA with difference d=0 will be considered because of the data stationary tested and the seasonality shown above. There is no reason to include difference d in ARIMA if the data is stationary.

3. Modeling

The training set is the windspeed over the first 432 months. The test set contains the last 48 months data that has been sampled out. In this case, BIC combined with other statistics would be mainly used to help evaluate and select the time series models.

The auto ARIMA with BIC as information criteria suggests that seasonal ARIMA(1,1,0)(1,1,0)[12] might be a reasonable model for the data. But since the difference d should be set to 0 and its BIC=440.45 which is not small enough, the model still needs to be adjusted by order selection.

By observing the lags and spikes on the ACF and PACF plots for de-seasonalized data in Figures 2.2., the orders can be chosen and tried are 0, 1 for p, 0, 1 for q, 1, 2 for P and 0, 1 for Q for our seasonal ARIMA model. After iterating all the possible models, we will focus on the four models with the lowest BIC values, ARIMA(0,0,0)(1,1,1)[12] with BIC=150.5056, ARIMA(1,0,1)(1,1,1)[12] with BIC=155.3281, ARIMA(1,0,0)(1,1,1)[12] with BIC=155.3179, and ARIMA(0,1,1)(1,1,1)[12] with BIC= 155.5003.

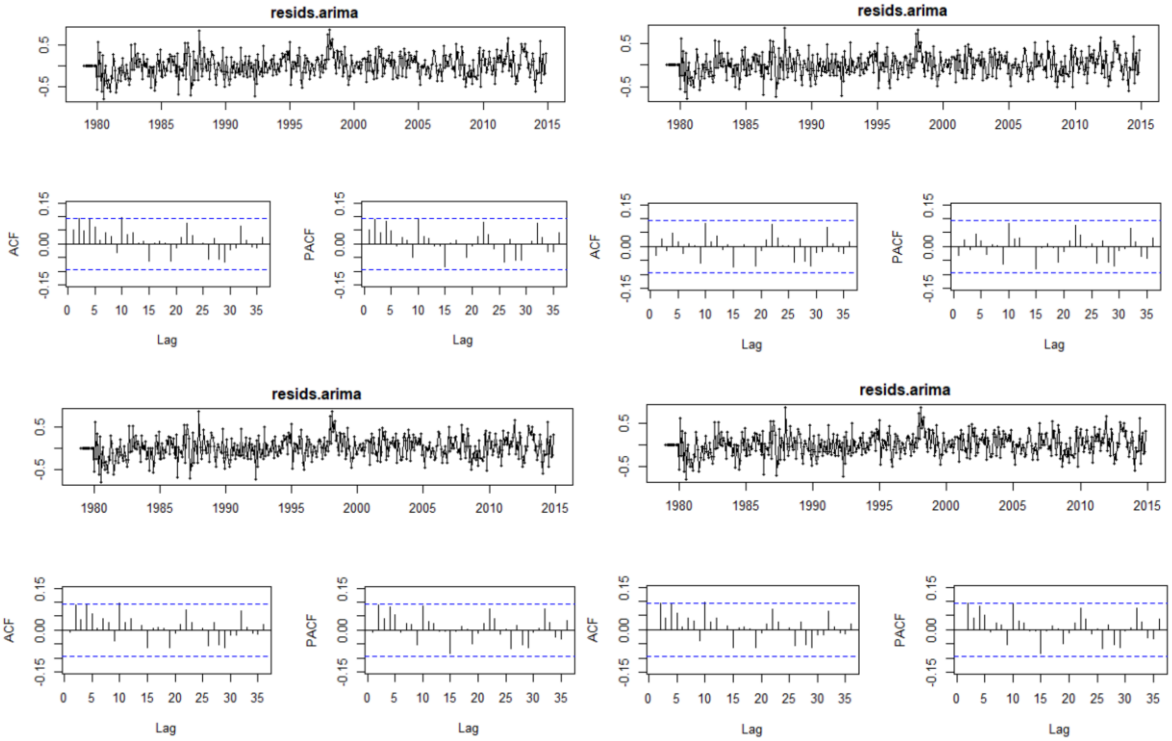

FIGURE 3.1. *The ACF, PACF and residuals plots summary of ARIMA(0,0,0)(1,1,1)[12] (left top), ARIMA(1,0,1)(1,1,1)[12] (right top), ARIMA(1,0,0)(1,1,1)[12] (left bottom), and ARIMA(0,1,1)(1,1,1)[12] (right bottom).*

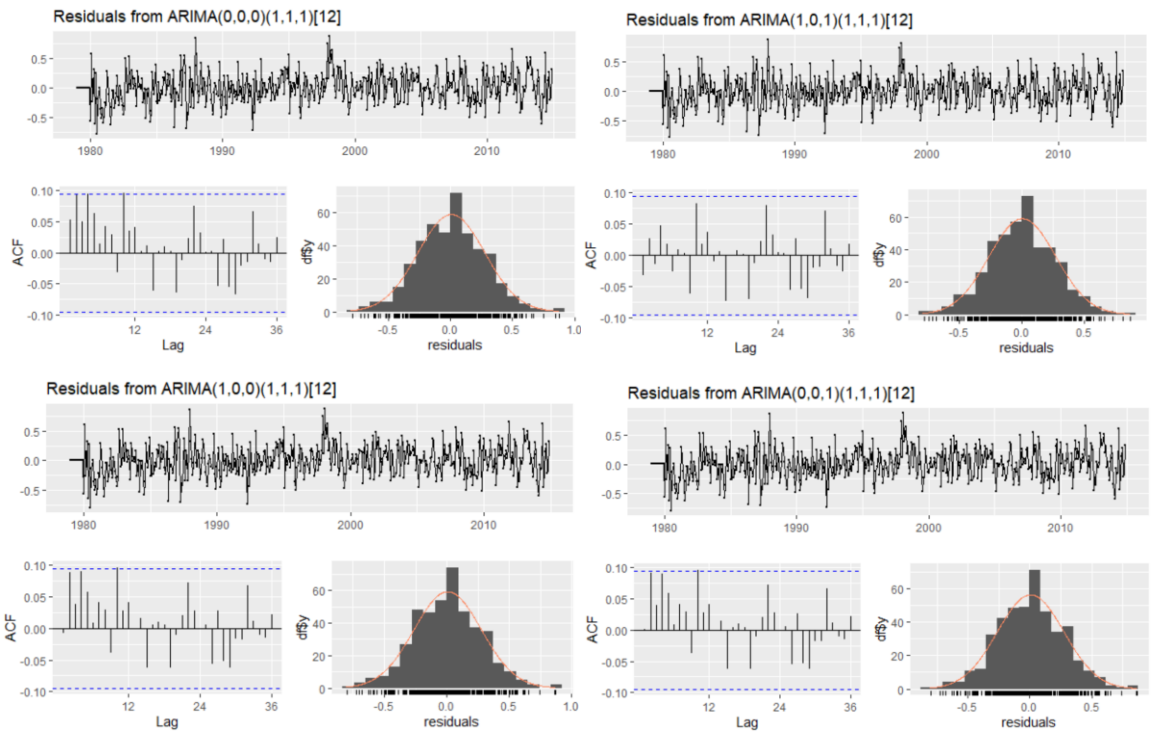

FIGURE 3.2. *The residuals plots summary of four models, including the corresponding ACF and histograms.*

From Figure 3.2., the residuals of these four seasonal ARIMA models approximately follow the normal distribution and almost all the lags for ACF and PACF shown in Figure 3.1. are within the 95% confidence interval.

	model	AIC	AICC	BIC	MAPE	Ljung.Box.p.value
1	ARIMA(0,0,0)(1,1,1)[12]	138.4399	138.4976	150.5606	6.266020	0.09217
2	ARIMA(1,0,1)(1,1,1)[12]	135.1268	135.2718	155.3281	6.130393	0.43980
3	ARIMA(1,0,0)(1,1,1)[12]	139.1569	139.2532	155.3179	6.252261	0.15370
4	ARIMA(0,1,1)(1,1,1)[12]	139.3392	139.4356	155.5003	6.254506	0.13730

FIGURE 3.3. *AIC, AICC, BIC, MAPE and p-value of Ljung-Box test from model summaries.*

Comparing the result values, ARIMA(1,0,1)(1,1,1)[12] has the least AIC, AICC and MAPE. Its p-value for Ljung-Box test is 0.4398 > 0.05, which indicates that we cannot reject the null hypothesis and the model does not show lack of fit.

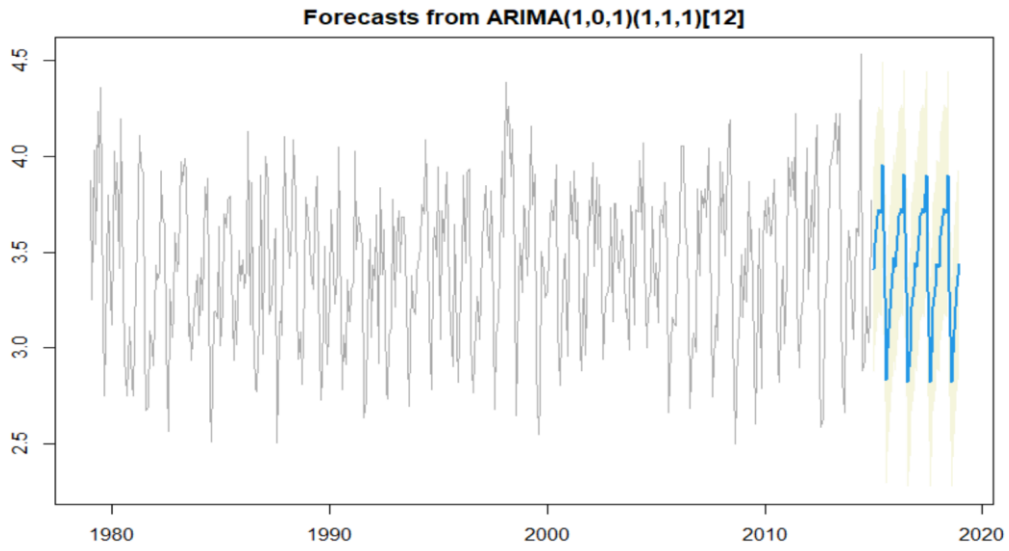

FIGURE 3.4. *Prediction plot for site X1 by ARIMA(1,0,1)(1,1,1)[12] with h=48.*

The prediction results on test data have been shown in Figure 3.4. with RSME=0.2938 and MAPE=6.9353. The plot provides the points forecasted with 95% prediction intervals. Since ARIMA(1,0,1)(1,1,1)[12] performs well in plots, model summaries and prediction, it would be used for fitting other two sites, X562 and X808, that are randomly chosen by sample to further test and support the model selection.

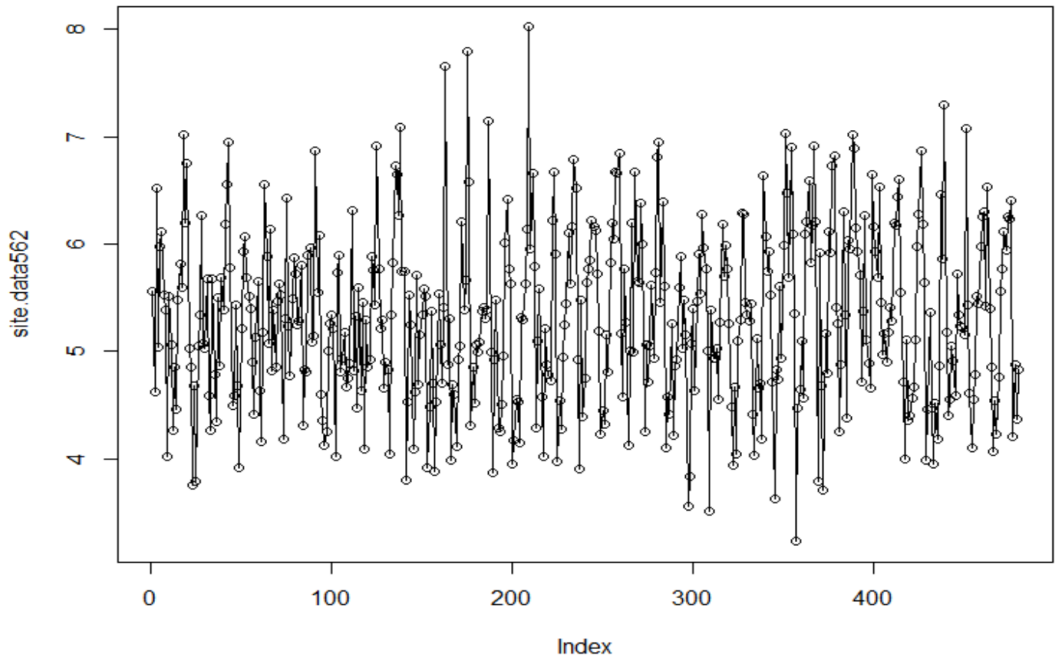

FIGURE 3.5. *Time series plot for site X562 over 480 months as index, presenting monthly average windspeed, connecting with lines.*

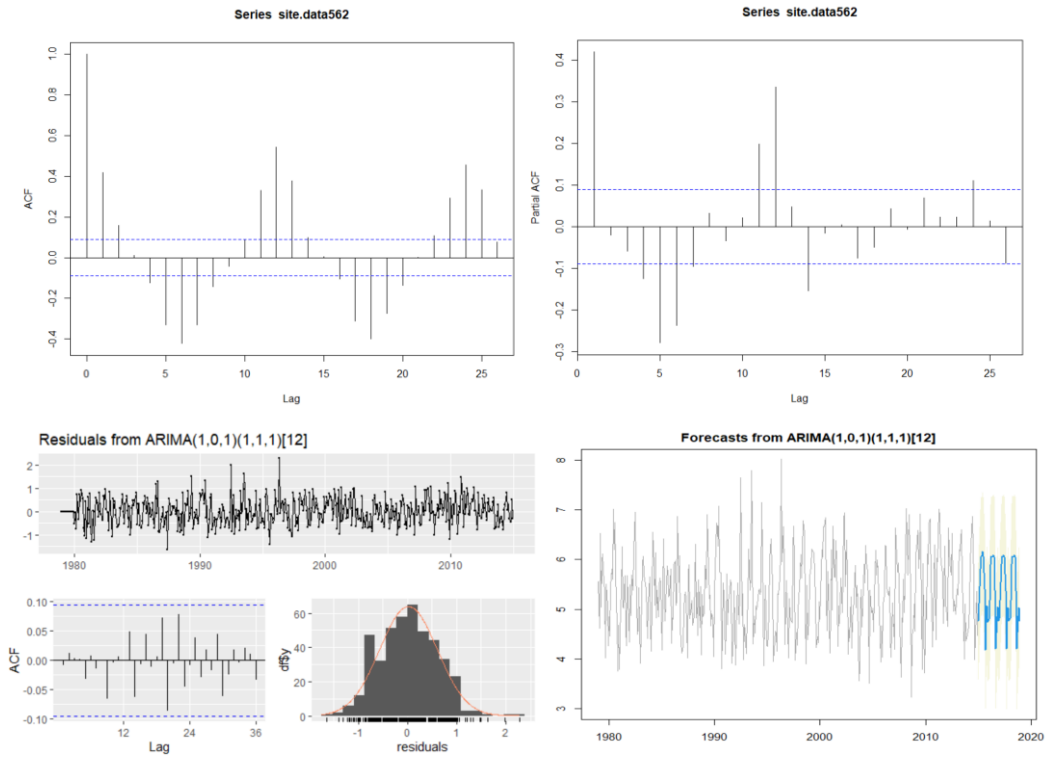

FIGURE 3.6. *The ACF, PACF plots of original data for site X562 (top). Residuals plots summary and prediction plot for site X562 (bottom).*

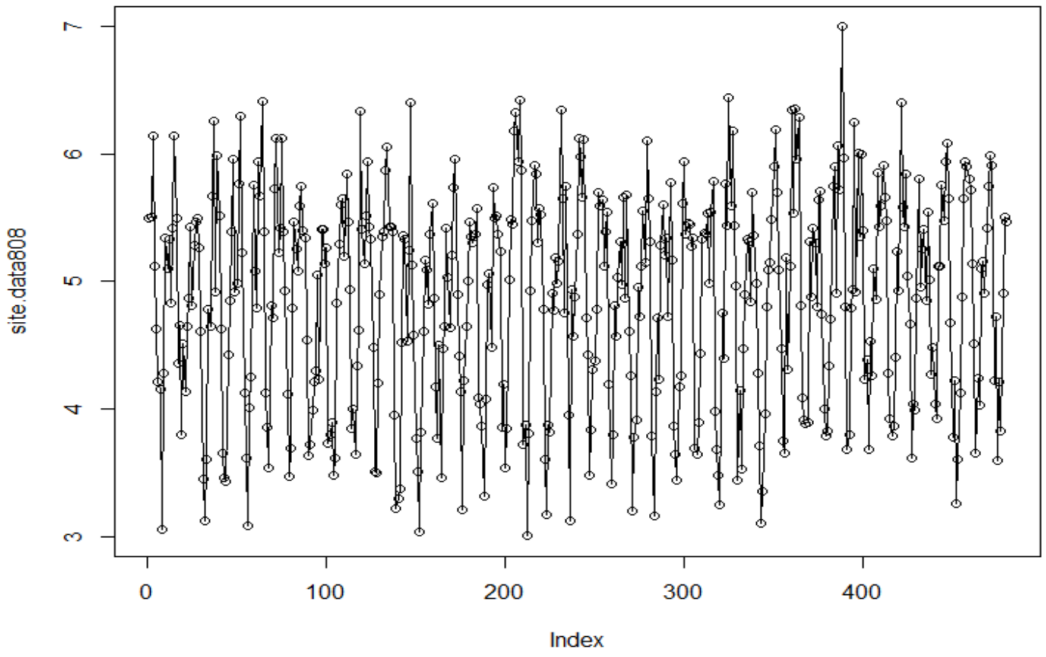

FIGURE 3.7. *Time series plot for site X808 over 480 months as index, presenting monthly average windspeed, connecting with lines.*

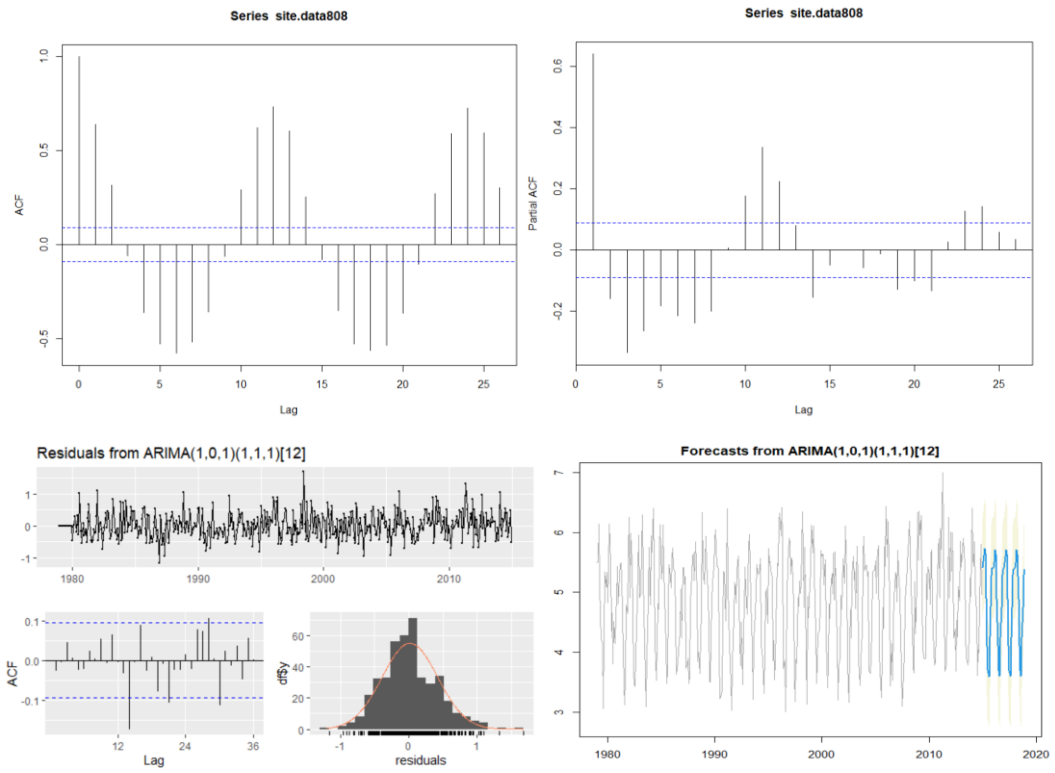

FIGURE 3.8. *The ACF, PACF plots of original data for site X808 (top). Residuals plots summary and prediction plot for site X808 (bottom).*

Based on Figure 3.6. and Figure 3.8., similar seasonality shown in ACF and PACF plots for X562 and X808. The residuals are distributed normally and the lags in corresponding ACF plot after modeling show relatively randomness and they are nearly all located inside the 95% confidence interval.

	location	RMSE	MAE	MPE	MAPE
1	X1	0.2985691	0.2417647	0.5386632	6.930891
2	X562	0.5895186	0.4512869	-2.9337830	8.883560
3	X808	0.3866375	0.3230307	0.7253775	6.964230

FIGURE 3.9. *The prediction accuracy metrics of site X1, X562 and X808.*

In addition, by checking the prediction accuracy metrics of the model prediction (Figure 3.9.), the MAPEs are all less than 10%, indicating that the model works well in fitting and predicting the data. It supports that the seasonal ARIMA with the order selection (1,0,1)(1,1,1)[12] can be expanded.

4. Results

The model ARIMA(1,0,1)(1,1,1)[12] has been selected to be utilized to fit the data of all sites provided and predict the windspeed for the next six months. Although the data of the sites might behave similarly on their correlograms, they still have different characteristics. Some columns are more sensitive to the model, or they might have constraints that the model cannot capture. For example, ARIMA uses "CSS-ML" in modeling by default but site X76 cannot satisfy the constraints in optimization, which would probably cause an error. To tackle this problem, ARIMA(0,0,1)(1,1,1)[12], one of the four models found before with the nice statistical metrics, is used for the columns that do not satisfy the constraints for ARIMA(1,0,1)(1,1,1)[12].

	X37	X591	X725	X775	X841
1	4.48942272536489	6.60759113147572	5.92987773038559	5.63637123068171	5.46187104520474
2	5.41462049482019	6.96965600320892	5.9973580040585	5.71442476351889	5.58149159112704
3	5.89698503028937	7.44390909673565	6.26571730771082	5.88298328769402	5.73167683044954
4	6.61182298777668	7.59342474652256	6.22546530544139	5.86734788272212	5.67907736219639
5	5.93588884320199	7.57280806471637	5.61336086886598	5.15465457637368	4.92466586610643
6	5.44029026056306	7.33240361404716	4.80021480999536	4.35729020633835	4.11670720994792

FIGRUE 4.1. *Prediction results for five sites. The sites are randomly selected.*

Finally, the seasonal ARIMA(1,0,1)(1,1,1)[12] and the alternative ARIMA(0,0,1)(1,1,1)[12] are applied to the data of 916 sites in total to predict the windspeed for the following six months in the future. The table in Figure 4.1. is a five-site sample of the forecasting results.

5. Appendix

Selected Code for final prediction:

```
final.pred <- data.frame(future_months = c(1,2,3,4,5,6))
modelfit <- function(data, i) {
  site.data <- data[, i]
  y <- ts(site.data, start = c(1979, 1), end = c(2018, 12), frequency = 12)
  t <- try(Arima(y=y, order=c(1,0,1), seasonal=c(1,1,1)), silent = TRUE)
  if("try-error" %in% class(t)) {
    arima.fit <- Arima(y=y, order=c(0,0,1), seasonal=c(1,1,1))
  } else {
    arima.fit <- Arima(y=y, order=c(1,0,1), seasonal=c(1,1,1))
  }
  pred <- forecast(arima.fit, h=6)
  return(pred$mean)
}
for (i in 3:918) {
  site.name <- paste("X", as.character(i-2), sep = "")
  pred.value <- modelfit(weather, i)
  final.pred[site.name] <- pred.value
}
# final prediction results are stored in final.pred
```