A REPORT
ON

ANALYSIS OF ANTARCTIC WEATHER DATA

BY

ANIRUDHA K

2015B3A7626P

M.Sc.(Hons.) ECONOMICS AND

B.E. (Hons.) COMPUTER SCIENCE

Prepared in partial fulfillment of the Practice School-I BITS C221

AT

National Centre for Antarctic and Ocean Research, Goa.

A Practice School-I Station of

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI

JUNE, 2017

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI
Practice School Division

Station: National Centre for Antarctic and Ocean Research, Goa
Duration: 55 days
Date of start: 22 May, 2017
Date of submission: 15 June, 2017

Title of the Project: Analysis of Antarctic Data Analysis

Name: ANIRUDHA . K
ID: 2015B3A7626P
Discipline: Economics and Computer Science

Name and designation of Expert: Dr. Sakthivel Samy, Head–ITCD, NCAOR.

Name of the PS faculty: Mr. Amit Setia

Key Words: SARIMA, ACF , PACF , Prediction , Blizzard

Project Area: Python, Time-Series Data Analysis

Abstract: There are two parts. First temperature data has been analyzed with the Seasonal ARIMA model. Then we predict future values with the help of obtained model. Second part deals with the wind speed analysis. Here main objective was to find out the monthly blizzard distribution.

Signature of Student

Date : 19h June, 2017

Signature of PS Faculty

Date : 19th June, 2017

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without mention of the people who had been of great help to me in making this possible.

I express my wholehearted thanks to Dr. M. Ravichandran Director, National Centre for Antarctic and Ocean Research (NCAOR), Goa, for allowing us to undertake this project in this esteemed organization. I also want to thank Dr. Ashoke Kumar Sarkar, Director of Bits-Pilani, Pilani Campus for giving us such a good opportunity to have practical exposure after our 4th semester.

I would like to express our sincere thanks to the PS-1 programme coordinator at NCAOR, Dr. Rahul Mohan for his continued support and guidance. I am highly indebted to my project mentor Dr. Sakthivel Samy V, Head of Information Communication Technology Division for giving me an opportunity to work on this project. He has been a true mentor, motivator and most importantly a remarkable source of inspiration.

I would like to express my special gratitude to our instructor Dr. Amit Setia for guiding, motivating and supporting us at every phase of this period. He made many efforts in resolving our problems throughout our stay. We are extremely fortunate to have him as our PS instructor.

I wish to avail myself of this opportunity, express a deep sense of gratitude and love to my beloved parents and my friends for their support, strength and help.

TABLE OF CONTENTS

SARIMA MODEL TO PREDICT MONTHLY MEAN TEMPERATURES

INTRODUCTION

Autoregressive Integrated Moving Average Model (ARIMA), is a widely used time series analysis model in statistics. ARIMA model was firstly proposed by Box and Jenkins in the early 1970s, which is often termed as Box-Jenkins model or B-J model for simplicity (Stoffer and Dhumway, 2010). ARIMA is a kind of short-term prediction model in time series analysis. Because this method is relatively systematic, flexible and can grasp more original time series information, it is widely used in meteorology, engineering technology, Marine, economic statistics and prediction technology, (Kantz and Schreiber, 2004; Cryer and Chan, 2008). The general ARIMA model is also applicable for non-stationary time series that have some clearly identifiable trends (Stoffer and Dhumway, 2010). We usually denote ARIMA model as ARIMA(p, d, q), where P and q are non-negative integers that correspond to the order of the autoregressive, integrated and moving average parts of the model, respectively. In addition to the general ARIMA model, namely non-seasonal ARIMA(p, d, q) model, we should also consider some periodical time series. The periodicity of periodical time series is usually due to seasonal changes (including monthly, quarterly and degree of weeks change) or some other natural reasons. We can build pure seasona A ARIMA(P,D,Q) model (He, 2004) with the time series date in different cycle and the same phase, the parameters P, D and Q are the relevant seasonal autoregressive parameter, seasonal integrated parameter and seasonal moving average parameter.

Considering the data relation, we can build a multiplication seasonal SARIMA(p, d, q)(P, D, Q)s model, (Wang et al., 2008). The model has been successfully applied in many subjects. In practical applications, the order of model SARIMA is usually not too large (Guo, 2009). If the period of time sereis equals to 12, it can be denoted as SARIMA(p, d, q)(P, D, Q)12. In the adjustment of the season, this is a very convenient, steady model.

In this study, we will take the monthly mean temperature time series as an example to build an seasonal ARIMA model and then forecast the monthly mean temperatures in the next few months. Specifically, in a seasonal ARIMA model, once we have smoothed the data and identified the parameters D and d, other parameters P, Q, P and q can be preliminarily identified from the ACF and PACF of the stationary processing series. Other related technologies were also used in the study.

MATERIALS AND METHOD

METHOD :

SARIMA model - Seasonal Autoregressive Integrated Moving Average

DATA :

IIG MAITRI ,ANTARTICA – DURATION : 2012 JANUARY to 2015 DECEMBER

This data was divided into two parts -

training data – 2012 jan to 2015 april

test data – 2015 may to 2015 december

SOFTWARES :

PYTHON – ver 3.5

The following packages were used for analysis

- NUMPY – basic array functionality
- PANDAS – data reading
- MATPLOTLIB – data plotting
- STATSMODEL – SARIMA analysis

THEORY

ARIMA and SARIMA models are extensions of ARMA class in order to include more realistic dynamics, in particular, respectively, non stationarity in mean and seasonal behaviours.

In practice, many economic time series are nonstationary in mean and they can be modelled only by removing the nonstationary source of variation. Often this is done by differencing the series.

Suppose X_t is nonstationary in mean, the idea is to build an ARMA model on the series w_t, definible as the result of the operation of differencing the series d times (in general d = 1): $w_t = \Delta^d X_t$.

Hence, ARIMA models (where I stays for integrated) are the ARMA models defined on the d-th difference of the original process:

$$\Phi(B)\Delta^d X_t = \theta(B)a_t$$

where $\Phi(B)\Delta^d$ is called generalized autoregressive operator and $\Delta^d X_t$ is a quantity made stationary through the differentiation and can be modelled with an ARMA.

Consider a few examples:

- ARIMA (0,1,1) is $\Delta X_t = a_t - \theta_1 a_{t-1}$ → the first difference of X_t is modelled as MA(1).

- ARIMA (1,1,0) is $(1 - \Phi_1 B)\Delta X_t = a_t$ → the first difference of X_t is modelled as AR(1).

Often time series possess a seasonal component that repeats every s observations. For monthly observations s = 12 (12 in 1 year), for quarterly observations s = 4 (4 in 1 year). In order to deal with seasonality, ARIMA processes have been generalized: SARIMA models have then been formulated.

$$\Phi(B)_s \Phi(B^S)\Delta^D \Delta^d {}_s X_t = \theta(B)_s \Theta(B^S)\alpha_t$$

METHODOLOGY

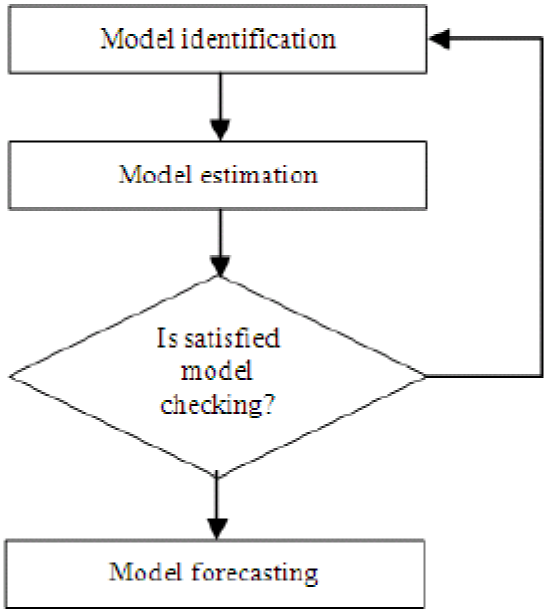


Fig 1.1.methodology flow chart

STEP 1 - DATA VERIFICATION

We need to make sure that the data is STATIONARY.
Data is said to be stationary if :

- Constant mean
- Constant variance
- covariance function depends on time difference

TEST FOR STATIONARITY
Unit Root Test was derived in 1979 by Dickey and Fuller to test the presence of a unit root vs. a stationary process. The unit root process and a stationary process are given by equations (1) and (2) below:

$$\rho_t = \varphi_1 \rho_{t-1} + e_t \quad \text{- (1)}$$

$$\rho_t = \varphi_0 + \varphi_1 \rho_{t-1} + e_t \quad \text{- (2)}$$

If rho=*1* then the series is said to have unit root and is not stationary.

The Unit Root Test as proposed by Kwiatkowski-Phillips-Schmidt-Shin(KPSS), test the hypothesis below:

H_0: Φ_1=series *is level or trend stationary*

H_A: Φ_1=series *is level or trend non-stationary*

If test statistic value of the KPSS test is less than critical value, we accept the null hypothesis that the data is level or trend stationary. Similarly, the Unit Root Test as proposed by Dickey and Fuller (ADF), test the hypothesis below:

H_0: Φ_1=series *has unit root*

H_A: Φ_1=series *has no unit root*

If test statistic of the ADF test is less than critical value we reject the null hypothesis that the data has a unit root.

STEP 2-MODEL IDENTIFICATION

- Assess the stationarity of the process
- If the process is not stationary, difference it (i.e. create an integrated model) as many times as needed to produced a stationary process to be modeled using the mixed autoregressive-moving average process mixed autoregressive-moving average process described above.
- Identify (i.e. determining the order of the process) the resulting the ARMA model.
  - The sample autocorrelation and sample partial autocorrelation .
  - Compare your ACF(Autocorrelation function) and PACF(Partial autocorrelation function) with the theoretical ACF and PACF .

The theoretical ACF and PACF for AR and MA model is shown below.

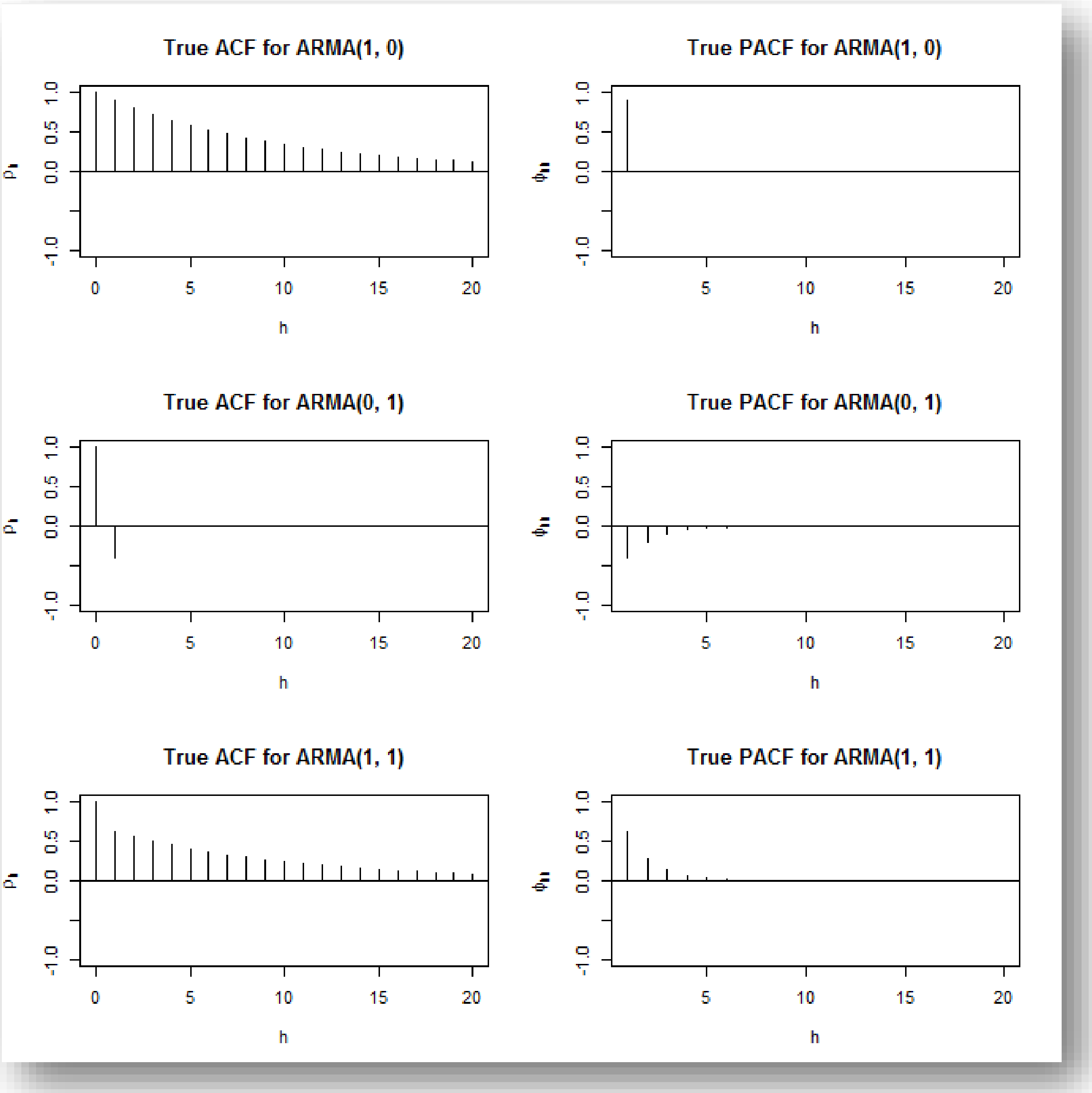


Fig 1.2 . theoretical ACF and PACF

In the above figure

ARMA(1,0) – is AR(1) MODEL

ARMA(0,1) – is MA(1) MODEL

For and AR(2) model ACF is similar to AR(1) but PACF will have 2 spikes rather than one .

If the model is seasonal ARMA their will be spikes in 4 th lag ,12 th lag ,etc as shown below.

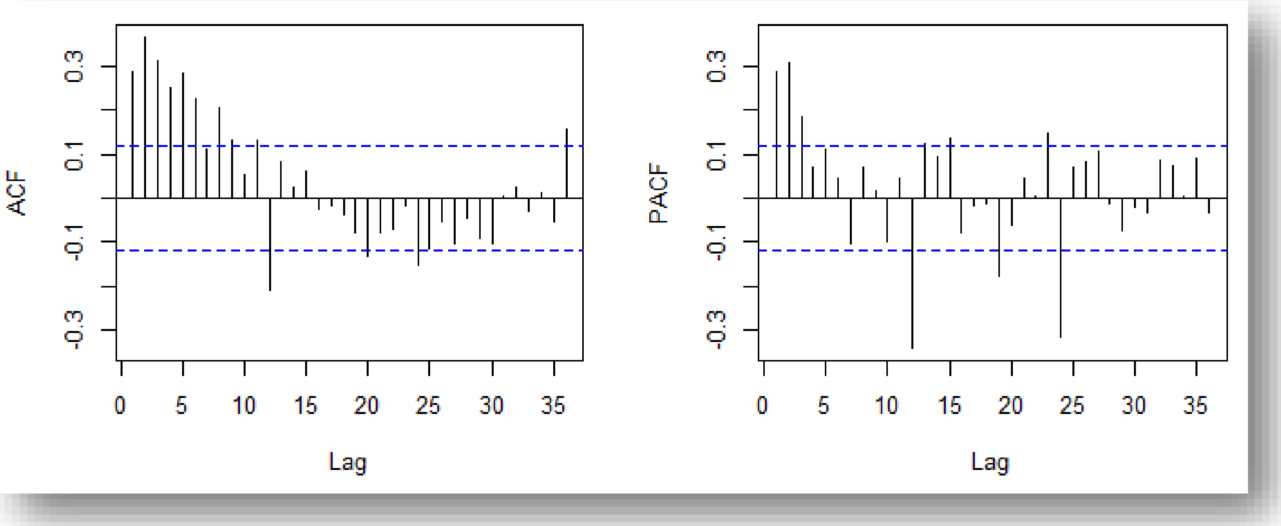


Fig.1.3. seasonal ARIMA example

Now comparing our ACF and PACF with the above graphs we can identify the order SARIMA model.

STEP 3-MODEL VERIFICATION AND PREDICTION

- Conduct visual inspection of the residual plots
- Residuals of a well-specified ARIMA model should mimic *Gaussian white noises*: the residuals should be uncorrelated and distributed approximated normally with mean zero and variance n^{-1}
- Apparent patterns in the standardized residuals and the estimated ACF of the residuals give an indication that the model need to be re-specified
- The *results.plot_diagnostics ()* function conveniently produce several plots to facilitate the investigation.
- The estimation results also come with some statistical tests
- If our model is proper then we can use it to predict future values

RESULTS

1) DATA VERIFICATION

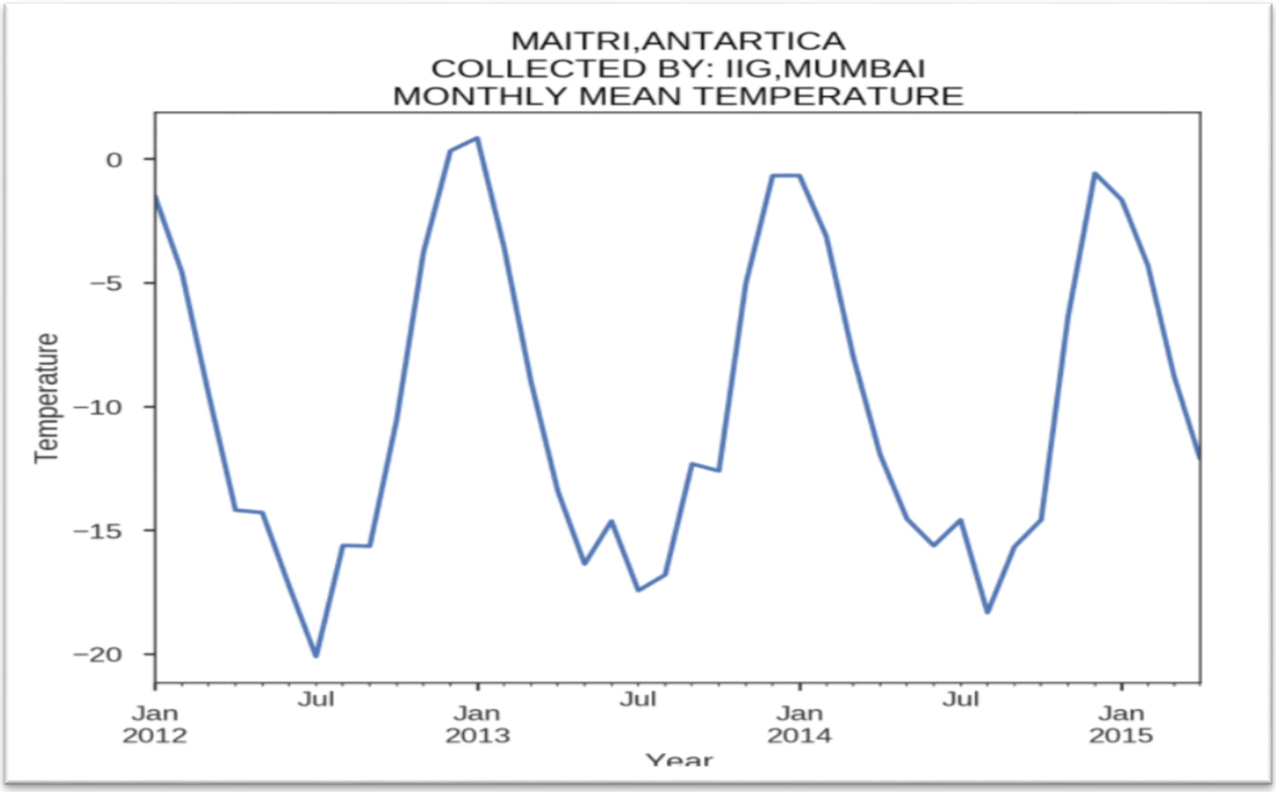
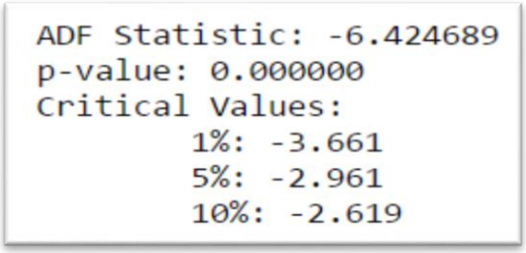


Fig 1.4. monthly mean temperature plot

- Here we have plotted monthly mean temperature with time
- We see that the data has strong seasonal trend
- However the data appears to have constant annual mean over the years
- Also the variance too appears to be constant
- We perform augmented **Dickey–Fuller test** (ADF) **test** confirm if the data is stationary. The results are as below:



```
ADF Statistic: -6.424689
p-value: 0.000000
Critical Values:
        1%: -3.661
        5%: -2.961
        10%: -2.619
```

Fig 1.5 ADF test results

Since ADF statistic is less the critical values, we reject H_0.
Hence we conclude that our data is Stationary.

2) MODEL IDENTIFICATION

We now plot ACF and PACF of our data.

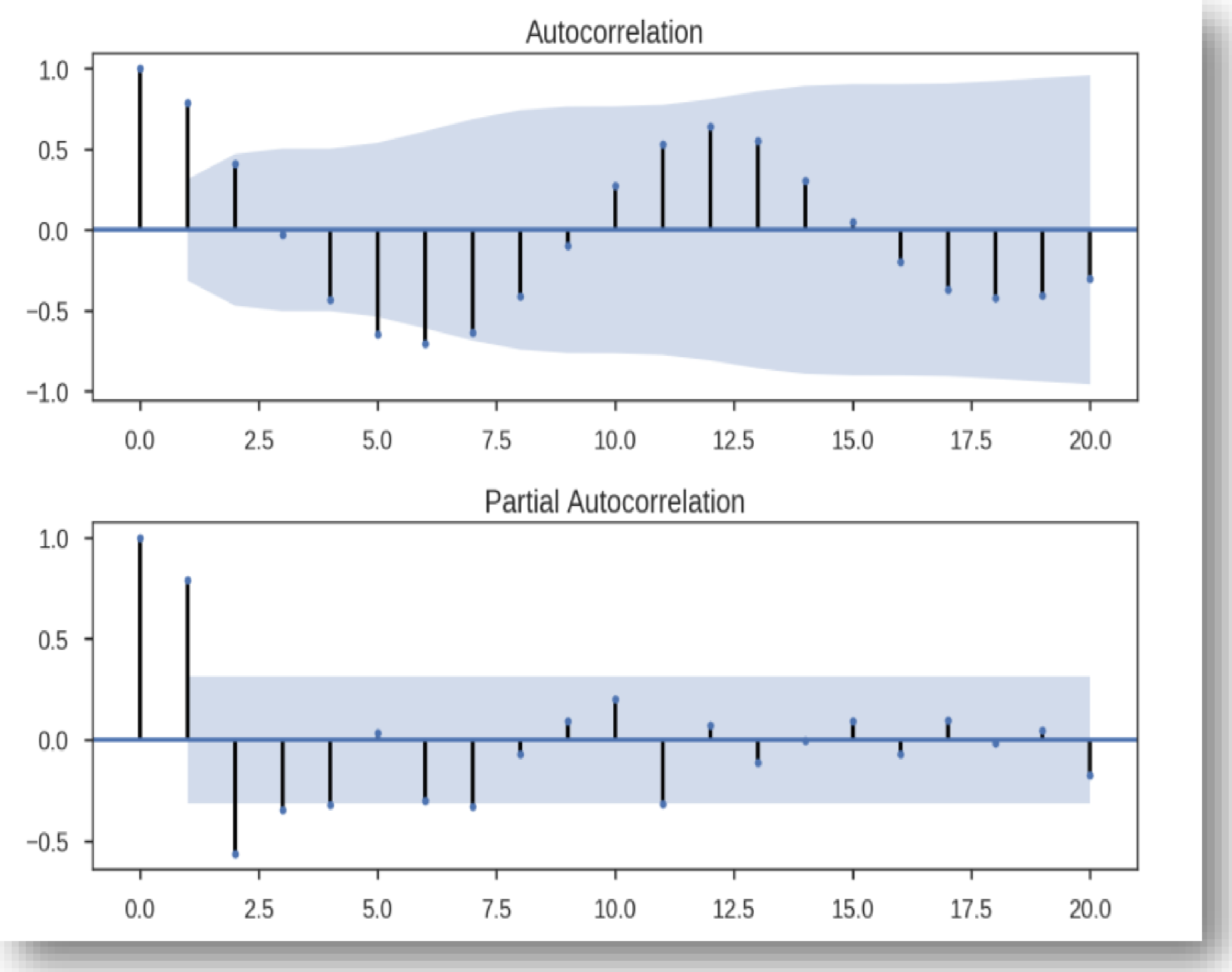
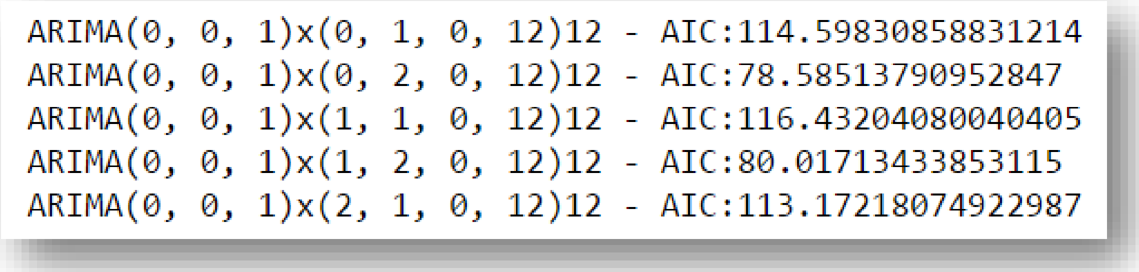


Fig 1.6 ACF AND PACF plots of the data

- On comparing with the theoretical models:
- It has an AR of order 0
- It has an MA of order 1
- It has a seasonal component with lag – 12
- Hence it is an ARIMA(0,0,1) with seasonal of lag 12

- Now we try different orders for seasonal component and select the order with least AIC value

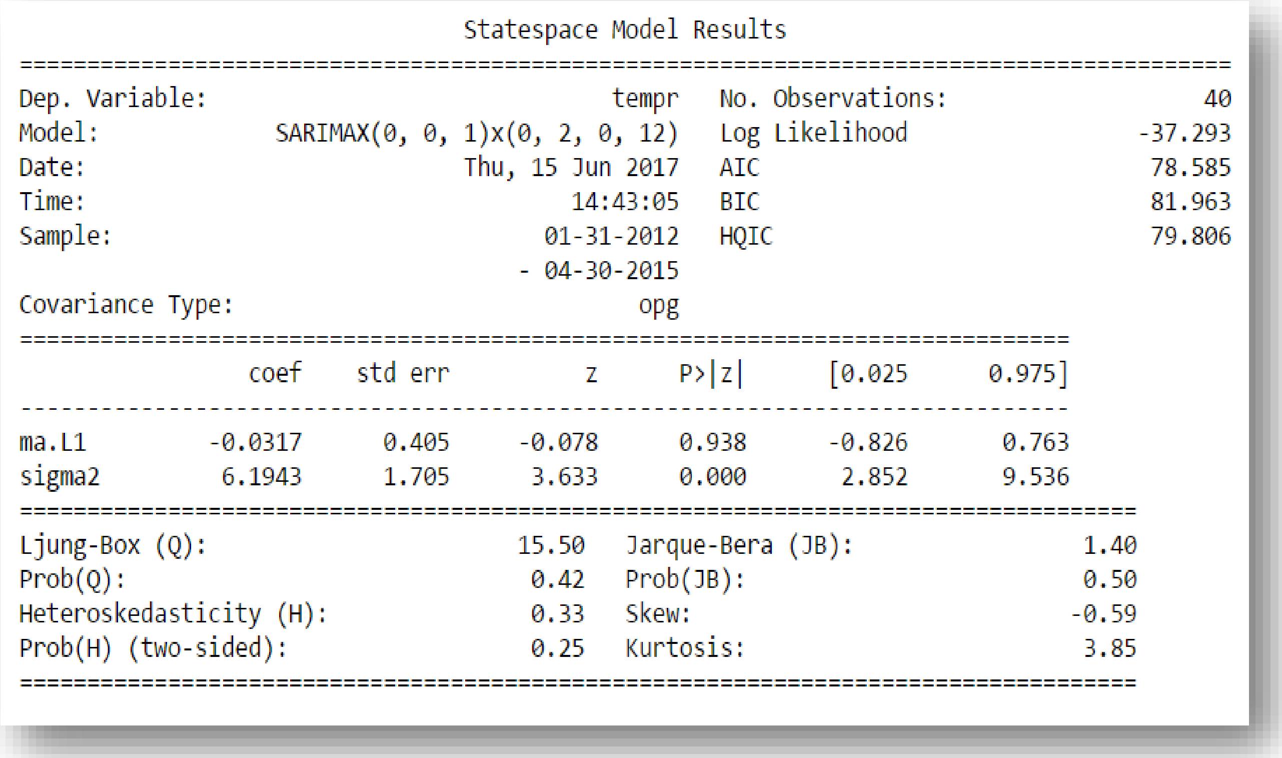


```
ARIMA(0, 0, 1)x(0, 1, 0, 12)12 - AIC:114.59830858831214
ARIMA(0, 0, 1)x(0, 2, 0, 12)12 - AIC:78.58513790952847
ARIMA(0, 0, 1)x(1, 1, 0, 12)12 - AIC:116.43204080040405
ARIMA(0, 0, 1)x(1, 2, 0, 12)12 - AIC:80.01713433853115
ARIMA(0, 0, 1)x(2, 1, 0, 12)12 - AIC:113.17218074922987
```

Fig 1.7 AIC for different SARIMA orders

- **ARIMA(0,0,1) * (0,2,0,12)** is selected as it has the least AIC value.

Results of the above model is shown below.



```
                                 Statespace Model Results
==========================================================================================
Dep. Variable:                             tempr   No. Observations:                    40
Model:             SARIMAX(0, 0, 1)x(0, 2, 0, 12)   Log Likelihood                 -37.293
Date:                           Thu, 15 Jun 2017   AIC                             78.585
Time:                                   14:43:05   BIC                             81.963
Sample:                               01-31-2012   HQIC                            79.806
                                    - 04-30-2015
Covariance Type:                             opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ma.L1         -0.0317      0.405     -0.078      0.938      -0.826       0.763
sigma2         6.1943      1.705      3.633      0.000       2.852       9.536
===================================================================================
Ljung-Box (Q):                       15.50   Jarque-Bera (JB):                  1.40
Prob(Q):                              0.42   Prob(JB):                          0.50
Heteroskedasticity (H):               0.33   Skew:                             -0.59
Prob(H) (two-sided):                  0.25   Kurtosis:                          3.85
===================================================================================
```

Fig 1.8 result of SARIMA analysis

3) MODEL PREDICTION

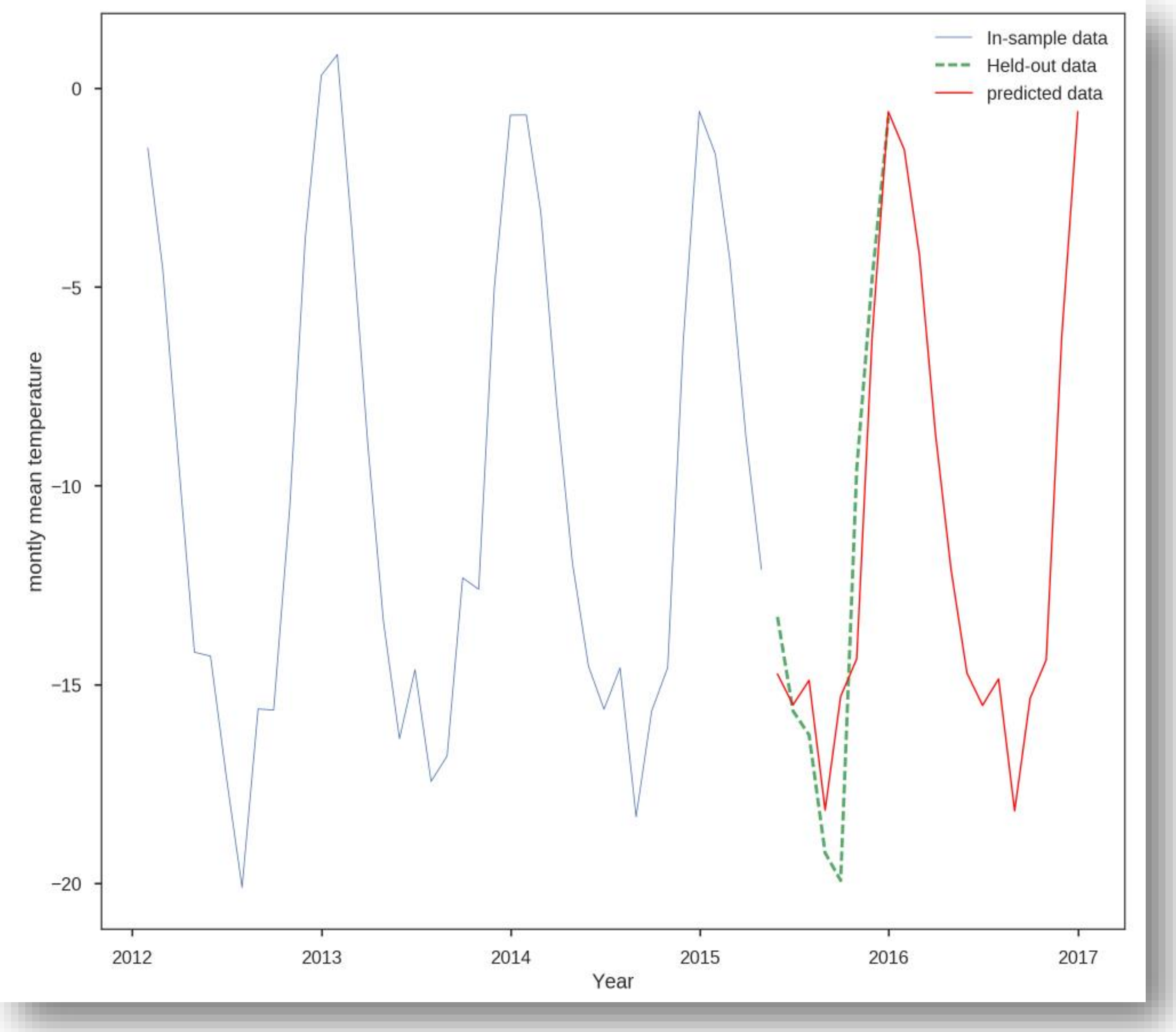


Fig 1.9 plot with predicted values

- The above plot has three components
  - Blue line – indicated original data used for training the model
  - Green dotted line – data not used for training but available
  - Red line – predicted values given by the our SARIMA model

WINDSPEED ANALYSIS

In this part we conduct a preliminary analysis on wind speed data and then find out the number of blizzards each month.

DATA SOURCE

IIG MAITRI ,ANTARTICA
DURATION : 2012 JANUARY to 2015 DECEMBER

PART 1 – MAX- MIN ANALYSIS

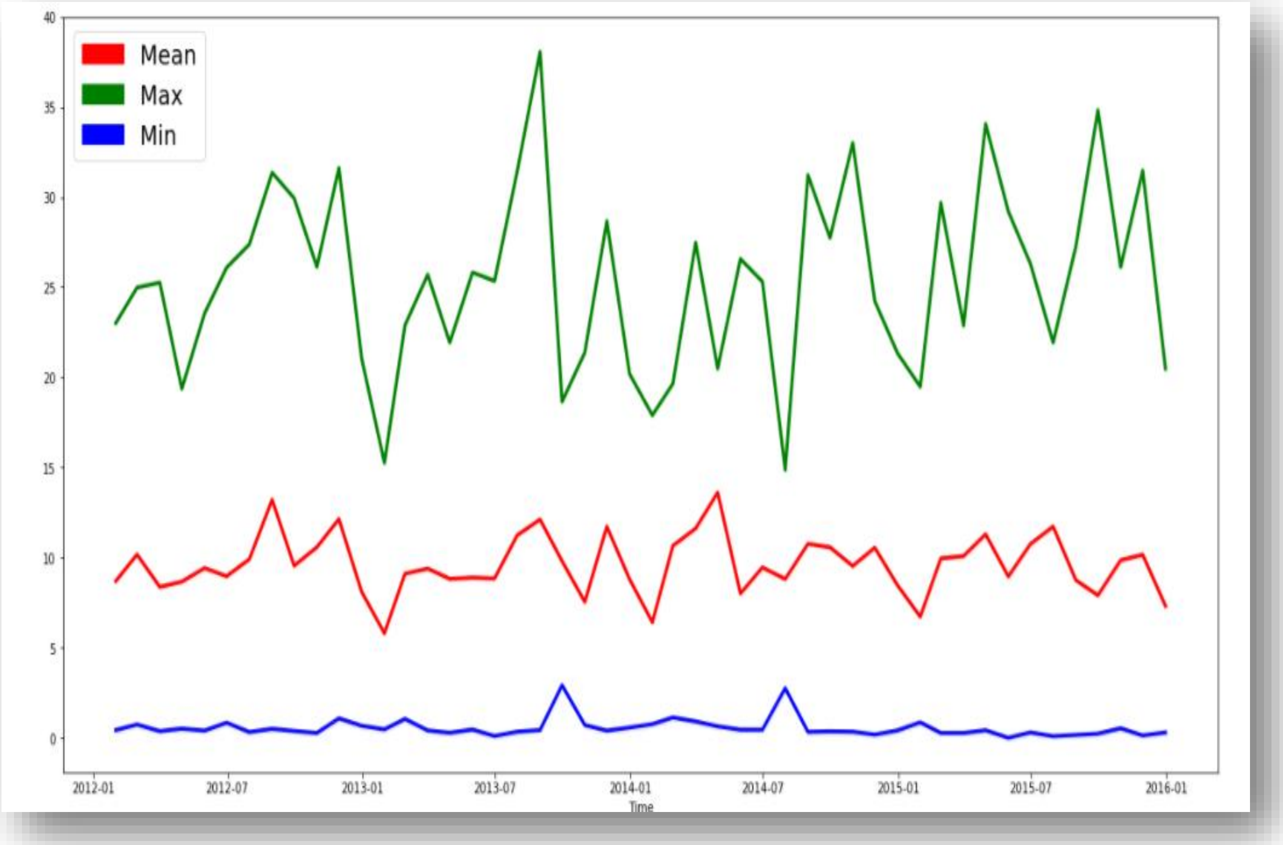


Fig. 2.1 min,mean,max plots of daily windspeed data

PART 2 – BLIZZARD ANALYSIS

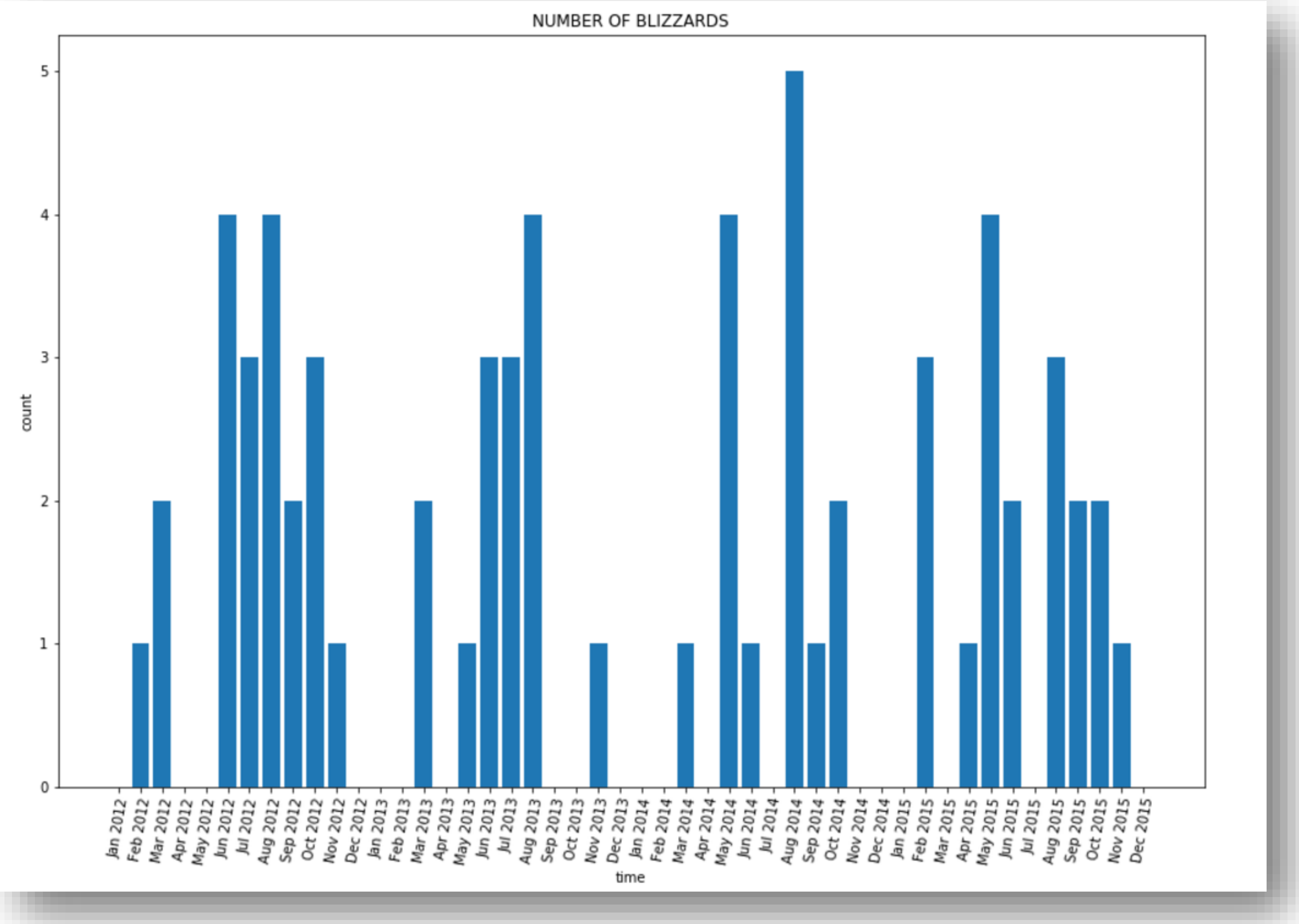


Fig 2.2 monthly blizzard occurrence plot

- Here number of blizzards each month was calculated(code in appendix 2)
- Conditions for a blizzard were.

  - wind speed greater than 23 knots for at least 3 hours
- It was observed that most of the blizzards occur in June to October period
- The **longest blizzard** was found out:
  - It ended on 2015-04-27 17:00:00 and lasted for 50 hours

REFERENCES

- *Stoffer, D.S. and R.H. Dhumway, 2010. Time Series Analysis and its Application. 3rd Edn., Springer, New York, ISBN-10: 1441978658, pp: 596*

- *Wang, Y., 2008. Applied Time Series Analysis. 1st Edn., China Renmin University Press, Beijing.*

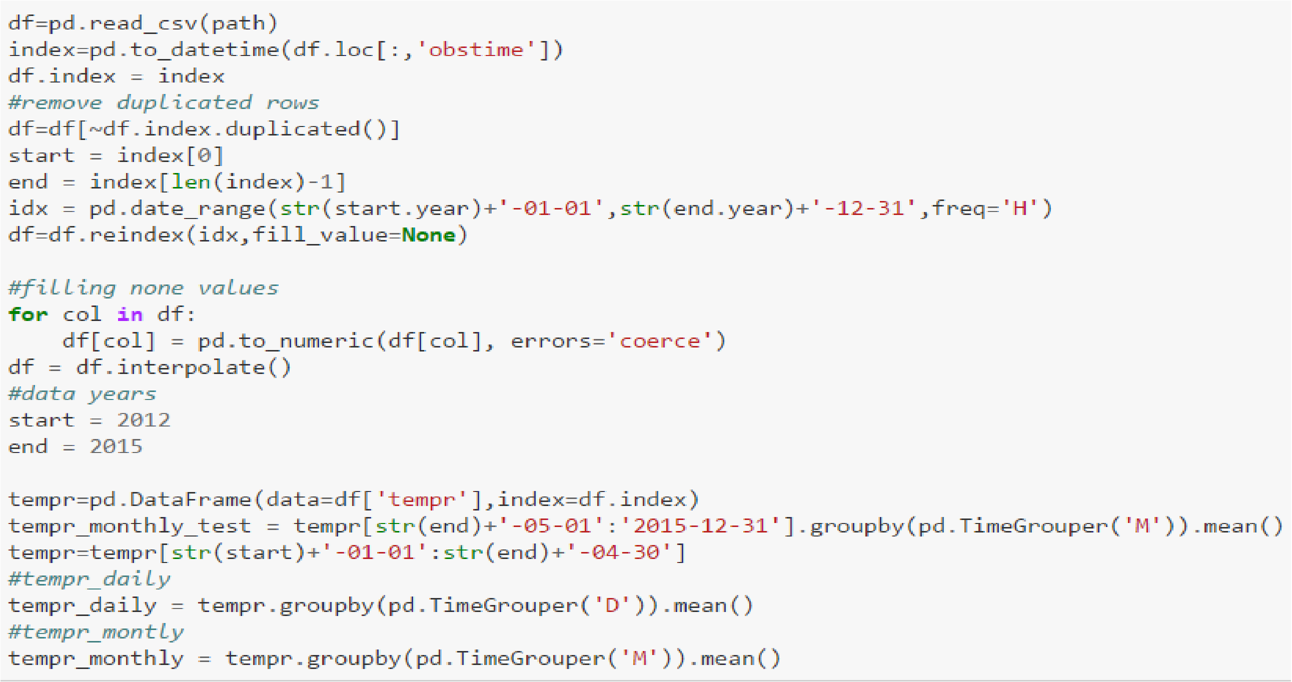
- *Guo, Z.W., 2009. The adjustment method and research progress based on the ARIMA model. Chinese J. Hosp. Stat., 161: 65-69.*

- *Book - Introduction to Time Series and Forecasting by Peter J. Brockwell*

APPENDIX

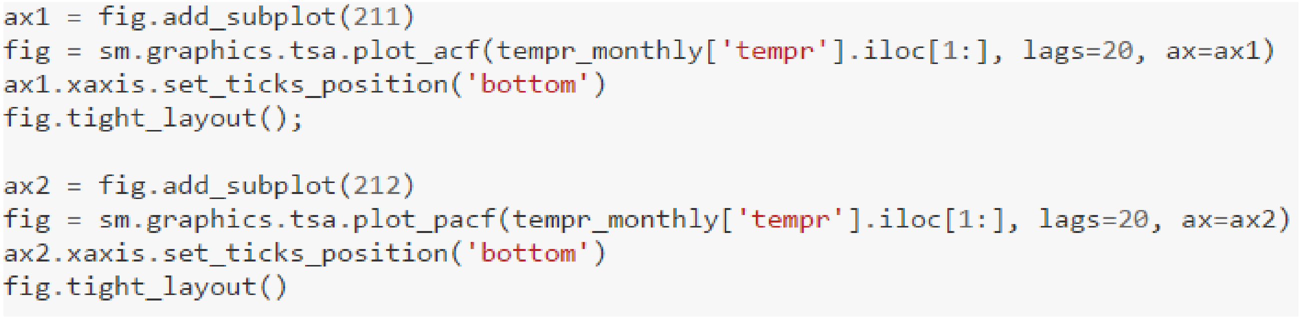
1) PROJECT - A CODE

- *reading data*



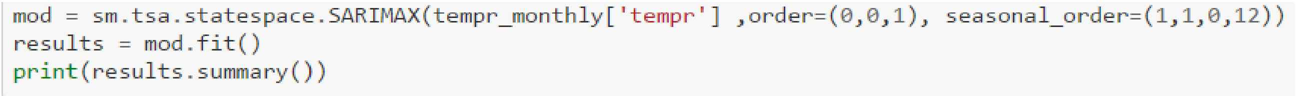
```
df=pd.read_csv(path)
index=pd.to_datetime(df.loc[:,'obstime'])
df.index = index
#remove duplicated rows
df=df[~df.index.duplicated()]
start = index[0]
end = index[len(index)-1]
idx = pd.date_range(str(start.year)+'-01-01',str(end.year)+'-12-31',freq='H')
df=df.reindex(idx,fill_value=None)

#filling none values
for col in df:
    df[col] = pd.to_numeric(df[col], errors='coerce')
df = df.interpolate()
#data years
start = 2012
end = 2015

tempr=pd.DataFrame(data=df['tempr'],index=df.index)
tempr_monthly_test = tempr[str(end)+'-05-01':'2015-12-31'].groupby(pd.TimeGrouper('M')).mean()
tempr=tempr[str(start)+'-01-01':str(end)+'-04-30']
#tempr_daily
tempr_daily = tempr.groupby(pd.TimeGrouper('D')).mean()
#tempr_montly
tempr_monthly = tempr.groupby(pd.TimeGrouper('M')).mean()
```

- *plot ACF and PACF*



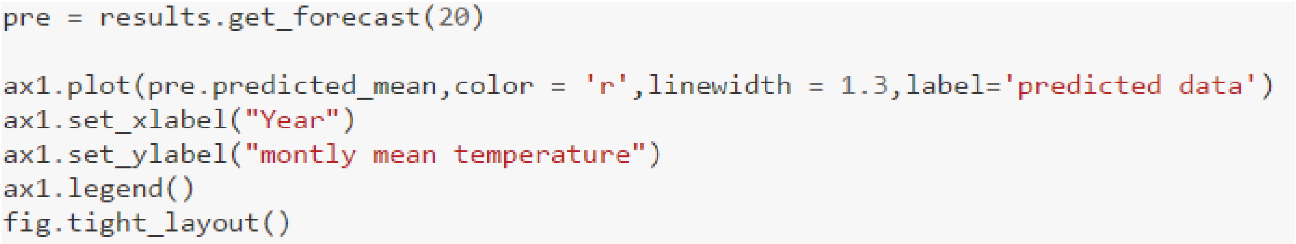
```
ax1 = fig.add_subplot(211)
fig = sm.graphics.tsa.plot_acf(tempr_monthly['tempr'].iloc[1:], lags=20, ax=ax1)
ax1.xaxis.set_ticks_position('bottom')
fig.tight_layout();

ax2 = fig.add_subplot(212)
fig = sm.graphics.tsa.plot_pacf(tempr_monthly['tempr'].iloc[1:], lags=20, ax=ax2)
ax2.xaxis.set_ticks_position('bottom')
fig.tight_layout()
```

- *fitting SARIMA model*



```
mod = sm.tsa.statespace.SARIMAX(tempr_monthly['tempr'] ,order=(0,0,1), seasonal_order=(1,1,0,12))
results = mod.fit()
print(results.summary())
```

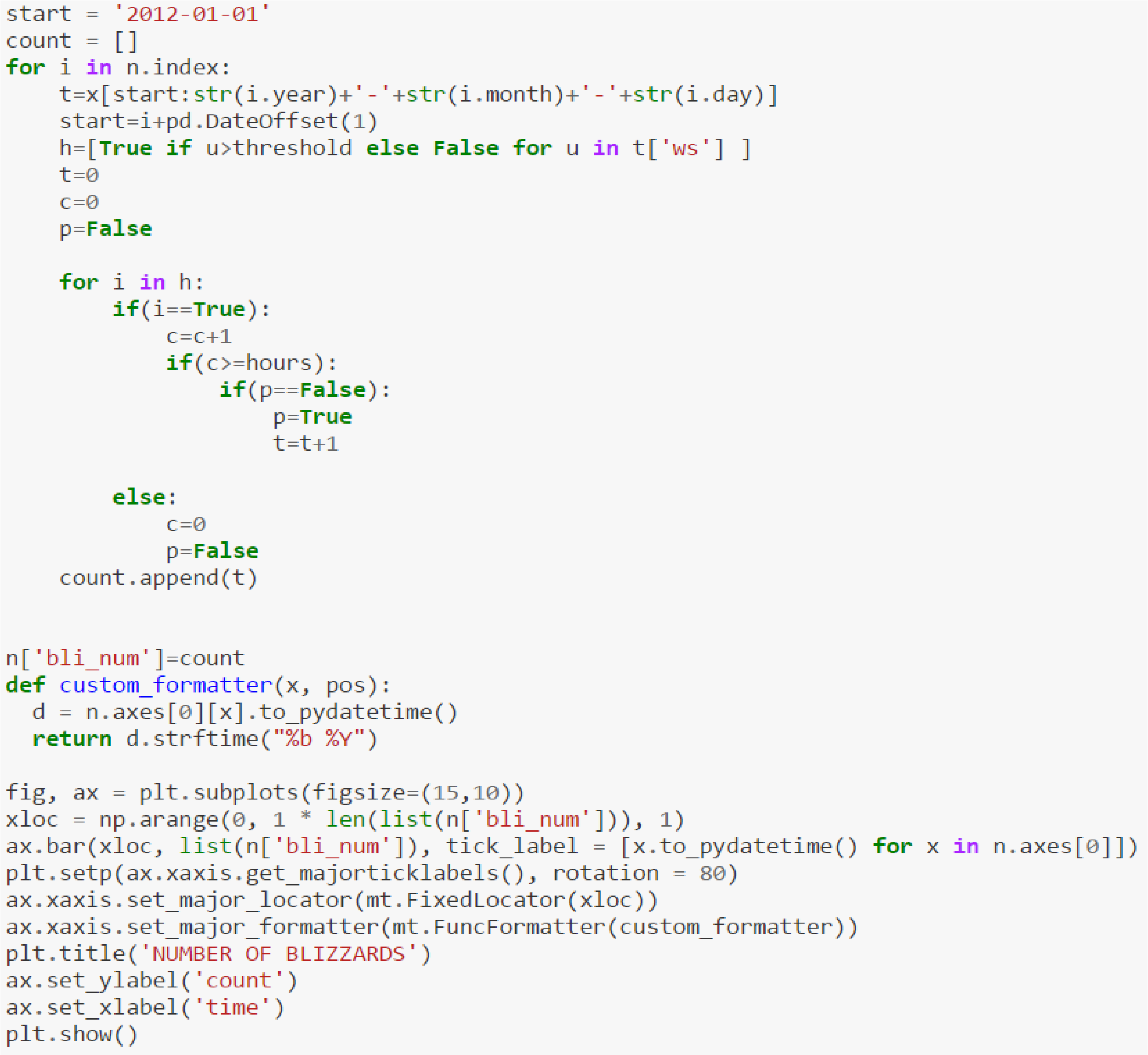
- *forecasting the values and plotting*



```
pre = results.get_forecast(20)

ax1.plot(pre.predicted_mean,color = 'r',linewidth = 1.3,label='predicted data')
ax1.set_xlabel("Year")
ax1.set_ylabel("montly mean temperature")
ax1.legend()
fig.tight_layout()
```

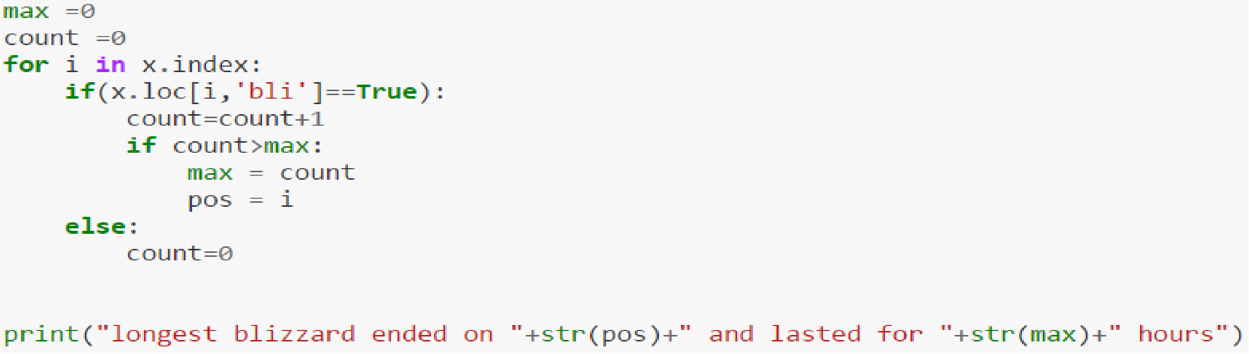
2) PROJECT - B CODE

- counting monthly blizzards and plotting



```
start = '2012-01-01'
count = []
for i in n.index:
    t=x[start:str(i.year)+'-'+str(i.month)+'-'+str(i.day)]
    start=i+pd.DateOffset(1)
    h=[True if u>threshold else False for u in t['ws'] ]
    t=0
    c=0
    p=False

    for i in h:
        if(i==True):
            c=c+1
            if(c>=hours):
                if(p==False):
                    p=True
                    t=t+1

        else:
            c=0
            p=False
    count.append(t)


n['bli_num']=count
def custom_formatter(x, pos):
  d = n.axes[0][x].to_pydatetime()
  return d.strftime("%b %Y")

fig, ax = plt.subplots(figsize=(15,10))
xloc = np.arange(0, 1 * len(list(n['bli_num'])), 1)
ax.bar(xloc, list(n['bli_num']), tick_label = [x.to_pydatetime() for x in n.axes[0]])
plt.setp(ax.xaxis.get_majorticklabels(), rotation = 80)
ax.xaxis.set_major_locator(mt.FixedLocator(xloc))
ax.xaxis.set_major_formatter(mt.FuncFormatter(custom_formatter))
plt.title('NUMBER OF BLIZZARDS')
ax.set_ylabel('count')
ax.set_xlabel('time')
plt.show()
```

- finding longest blizzard



```
max =0
count =0
for i in x.index:
    if(x.loc[i,'bli']==True):
        count=count+1
        if count>max:
            max = count
            pos = i
    else:
        count=0


print("longest blizzard ended on "+str(pos)+" and lasted for "+str(max)+" hours")
```