The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLVI-4/W6-2021
Philippine Geomatics Symposium 2021, 17–19 November 2021, virtual meeting

A TIME-SERIES AND SPATIAL ANALYSIS OF 56 YEARS (1961-2017) OF RAINFALL HISTORICAL DATA FROM MALAYBALAY, BUKIDNON

B. J. Buenaobra [1], MK L. Alleto[2], JM V. Manhuyod [1]

[1] Plantation Geomatics Dept., Del Monte Philippines Inc., Agusan Canyon, Manolo Fortich,
2 Crop Production Unit, Del Monte Philippines Inc., Agusan Canyon, Manolo Fortich,
Bukidnon, Northern Mindanao - (buenaobrabj, alletoml, manhuyodjmv) @delmonte-phil.com

KEYWORDS: Time Series Analysis, Rainfall Density, Rainfall Seasonality, Autocorrelation, Stationarity

ABSTRACT:

This paper focuses on using time series and spatial analysis methods to detect climate change indicators in Malaybalay, Bukidnon. We look at 56 years of historical rainfall data between the years 1961 to 2017 and perform a computational method for data processing to arrive at spatial statistics and provide data visualization. We demonstrate the use of the Augmented Dickey-Fuller test (ADF), where a *p-value* is tested versus a threshold to reject or accept the null hypothesis for a stationarity test. For the seasonality test, we perform a time-domain signal processing by an autocorrelation function. The time-series analysis shows that for Malaybalay, Bukidnon rainfall data shows ADF statistic of -16.348964, a *p-value*=0.000000 with critical values 1%:-3.431, 5%:-2.862, 10%:-2.567. Hence, the significant negative values indicate more likely to reject the null hypothesis. We showed that rainfall does not demonstrate periodicity, is not seasonal, and is non-stationary. This work does not cover those that can be detected and attributed to anthropogenic causes.

1. INTRODUCTION

1.1 Malayblay, Bukidnon as a region of interest

In the definition we find from the United States Geological Survey (USGS), the terms Global Warming and Climate Change are well differentiated. In verbatim, "Global warming" refers to the rise in global temperatures due mainly to the increasing concentrations of greenhouse gases in the atmosphere. "Climate change" refers to the growing changes in climate measures over a long period – including precipitation, temperature, and wind patterns. According to the Department of Science and Technology (DOST), the Philippines' oldest weather station system and weather data are located in Malaybalay, Bukidnon. The resolution specified by the agency is 50km.

To describe our area of interest in Region 10, Malaybalay City in Northern Mindanao has a total land area of 96,919 hectares, about 9.23% of the total area of Bukidnon province. An estimate of 65% is forestland/timberland, and the remaining 35% are alienable and disposable areas for purposes such as agriculture or industry. (PhilAtlas, 2021)

2. COMPUTATIONAL EXPERIMENT SETUP

2.1 Rainfall Data Pre-Processing

The rainfall data from the years 1961-2017 of Malaybalay, Bukidnon was purchased at the cost of Php.1.0 per parameter per day acquisition from the national weather bureau from CAD-PAGASA offices at the beginning of 2018. It was delivered in M.S. Excel format in 2 columns (dates) by 20,805 rows of rainfall data. (DOST PAGASA, 2021)

Marking of "T" designated no data as those "0" were considered no rainfall on the corresponding dates. As in the usual practice of data pre-processing, unknown data entries are removed or replaced by data transformation numerical operations such as averaging. Data with "T" markings were replaced by the average from their immediate neighboring cells value. It is assumed that no instrumental error was caused by the weather station instrument when the data of "0" appeared on the cells. The entire data processing and visualization were done Python programming using Jupyter Notebooks within the Anaconda development environment. (Anaconda Analytics, 2021). We demonstrate the plot generation programmatically in Python in the Appendices. (Anaconda, 2021)

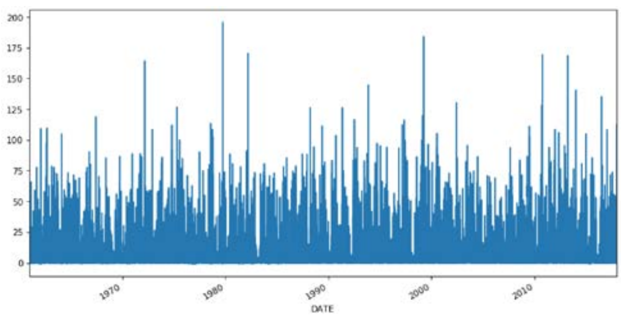


Figure 1. Line plot of Malaybalay rainfall between 1961-2017

2.2 General observations from January 1961 to 2017

For the same months but 56 years apart of rainfall, the later dates show more rain and higher intensity rain events. To site, on the16th January and 18th in 2017 is rainfall of 83.3mm and 108.4mm that when compared in 1961 on the same month and day there was only 0.0mm and 1.5mm. In 1961 the highest rainfall occurred on 29 January at 29.5mm, while for 2017, it was 108.4mm on 8 January. These were followed by more minor but still more intense than those occurring in 1961 on 28 January at 49.8mm and 74.3mm. Typhoons visit the island of Mindanao as shown by experience during the last quarter to the first quarter of the new year. (DOST PAGASA, 2021)

2.3 Descriptive Statistics of 1961-2017 Rainfall data

Calculating the descriptive statistics for the spanned years of the natural rainfall data from Malaybalay, Bukidnon, we have the following results from programmatic data processing:

Name: Rainfall	dtype: float64
count	20805
mean	7.127195
std	13.995723
min	-1.000000
25%	0.000000
50%	1.000000
75%	7.800000
max	195.900000

Table 1. Descriptive Statistics Malaybalay Rainfall

We can detect practically twice as much deviation from the mean value for the entire 56 years of rainfall. Also, at least 3/4th of rains as the whole is around 8mm of rain, and minima occur with zero precipitation or dry periods. Probably the most decisive rain event could be as high as under 200mm. We demonstrate programmatically using Python, obtaining the descriptive statistics in the Appendices. (Anaconda, 2021)

3. TIME-SERIES AND SPATIAL ANALYSIS

3.1 Temporal series of 56-year Malaybalay Rainfall

For a rainfall analysis, both the amount of rain with their temporal component is taken into account. The dimension is in units of day rain for each year for all 56 years of the entire observation. We have shown from the previous section the basic descriptive statistics. We offer the plot of the amplitudes on successive stacks of rainfall variation with time to visualize them over several years. This is shown in Figure 1.

3.2 Test for Stationarity

When the observations in time series are not dependent on time, we say it is stationary; otherwise, it's called non-stationary. It is also a test if our observations are consistent given the temporal structure. The most basic statistical test is the determination of the means and variances for stationarity or non-stationarity. The mean values are practically the same. Hence, Malaybalay rainfall is therefore stationary. (Brownlee, J., 2021)

mean1=6.943857	mean2=7.334317
variance1=186.006267	variance2=205.297065

Table 2. Means and Variance results from splitting data

3.3 Augmented Dickey-Fuller test for stationarity

We also made use of the so-called Augmented Dickey-Fuller test (ADF). A *p-value* that results from the ADF test below a threshold suggests rejecting the null hypothesis, which implies observation is stationary. Otherwise, a *p-value* above the threshold indicates means we failed to reject the null hypothesis, and thus it is non-stationary. Performing the ADF program, we have the following results in Table 3. The calculation is done programmatically from reading the measurement data using Python a code within the Jupyter Notebook (Augmented Dickey-Fuller Statistical Test). Calling the Autocorrelation function for the 1961-2017 Malaybalay, Rainfall data, we produced a plot in Figure 3.

ADF Statistic	-16.348964
p-value	0.000000
Critical Values	
1%	-3.431
5%	-2.862
10%	-2.567

Table 3. ADF Test Results

The *p-value* $<= 0.05$, we reject the null hypothesis (H0); rainfall does not have a unit root; hence, it is stationary. It does not have a time-dependent structure. (Brownlee, J., 2021**)**

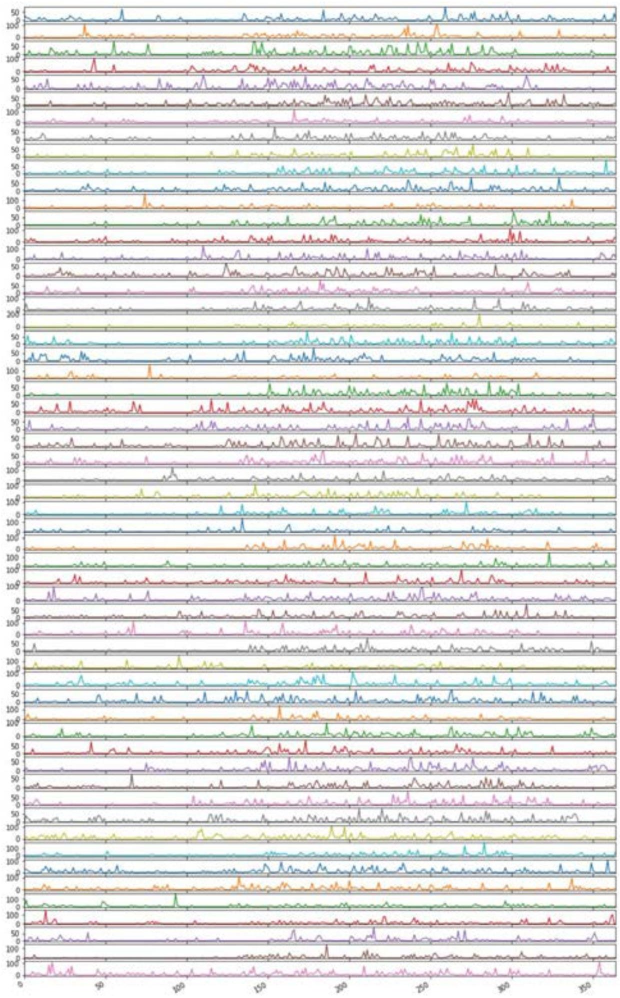


Figure 2. A stacked plot of Malaybalay Rainfall (1961-2017)

3.4 Test for Seasonality and Periodicity

For the test for seasonality, we perform the Autocorrelation function, also called self-correlation. When the time series data show statistically significant periodicity, it is seasonal.

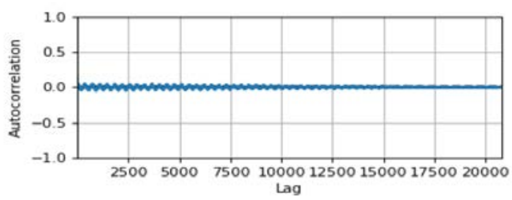


Figure 3. Autocorrelation of 1961-2017 Malaybalay rainfall

We can infer no significant periodicity or cycle of the same intensity across all the years, indicating no seasonality. (Proakis J., Manolakis D.G, 1995)

3.5 Spatial Visualization of 56-year Malaybalay Rainfall

For a 2D visualization of rainfall, as shown in Figure 3. Essentially the plots are 2D representations of a matrix where a column is made up of day rows, another for intensity, and generated a Gaussian interpolated heat map. These can show the intensity peaks of the month's rainfall for every year.

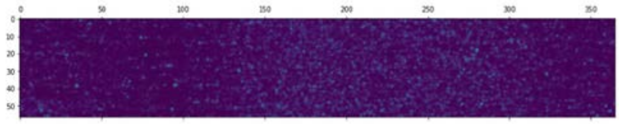


Figure 4. Heat Map of a typical yearly rainfall data

Figure 5 and Figure 6 below show the specific month data's heat map with Gaussian interpolation. Here we show the series for the years 1961 and 2017 below. Each column represents one month, with rows representing the days of the month from days 1 to 31.

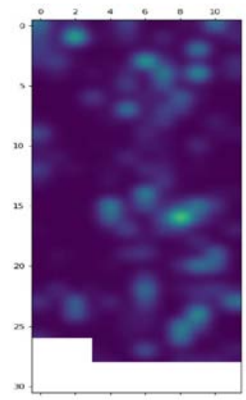


Figure 5. Heat Map 1961

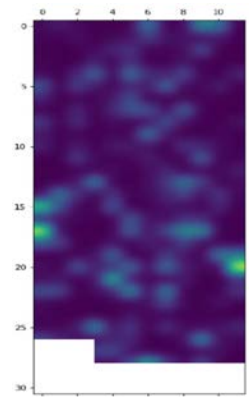


Figure 6. Heat Map 2017

We can further visualize the peaks of rainfall distribution in the 3D surface generation by image processing software, as shown in Figure 7 and Figure 8 (ImageJ)

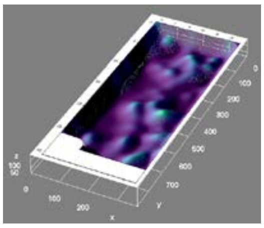


Figure 7. 3D surface plot 1961

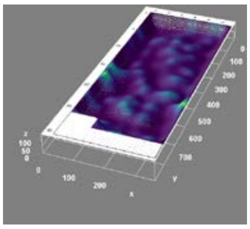


Figure 8. 3D surface plot 2017

3.6 Weather Station vs. Remote Sensed Dataset Quality

The Malaybalay 56 years of rainfall of 1961-2017 acquired and continues to this day collected by the agency PAGASA-DOST in Region X, is considered the oldest rainfall data obtained by the oldest weather station in the country. We are interested in comparing the old sensor on a mast in the ground with the newer satellite sensor over the atmosphere and extraterrestrial space. We have considered the data archive from LARC-NASA with its oldest available data as starting date. We have subsetted the years from January 1982 to December 2017 from the Malaybalay rainfall data dates to match the. We plot the histogram of the two datasets in Figure 9 below.

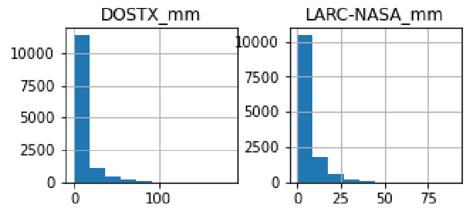


Figure 9. Histogram of rainfall dataset

We can see that the PAGASA-DOST rainfall histogram is practically the same as for frequency. However, the localized ground measurement data tends to spread more, reaching near 100mm, while LARC-NASA, which is remote sensed and is nearly 50mm with the integration process. For the line plots, we show both rainfall datasets we depict the matching span of years for visual comparison in Figure 10 below.

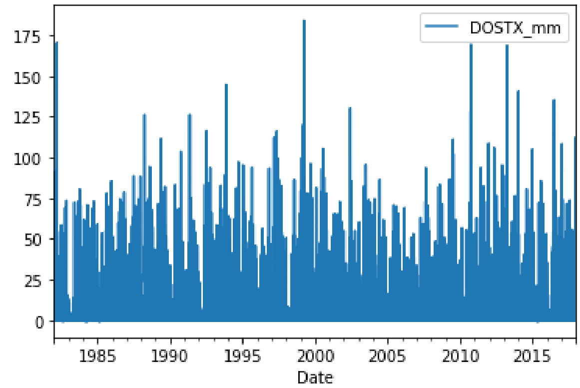


Figure 10. Weather Station Data PAGASA-DOST

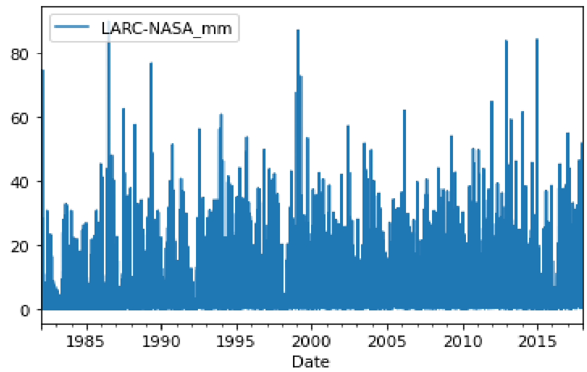


Figure 11. Remote Sensed Data LARC-NASA

We show the comparison of their individual shape of distribution by deriving their Density Plot programmatically from the two dataset. We can view the density plot as the smooth version of a histogram. In the following page we show the results for the density plot for Malaybalay historical rainfall data from PAGASA DOST collected by weather station and for a remote sensed data obtained from LARC-NASA archives. Figure 12 shows density plot for PAGASA-DOST dataset and followed by the Figure 13 that shows the density plot for LARC-NASA. The density plots below shows LARC-NASA data have narrower base than DOST dataset and with higher density.

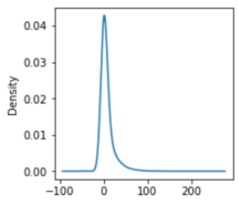


Figure 12. Density Plot DOST-PAGASA

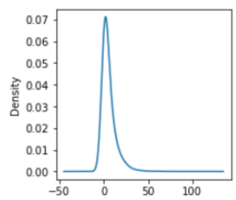


Figure 13. Density Plot LARC-NASA

We also performed the ADF Statistics for each dataset the results are shown below for each case.

```
Malaybalay DOST dataset:
ADF Statistic: -13.122388
p-value: 0.000000
Critical Values:
         1%: -3.431
         5%: -2.862
         10%: -2.567

LARC-NASA dataset:
ADF Statistic: -11.185995
p-value: 0.000000
Critical Values:
         1%: -3.431
         5%: -2.862
         10%: -2.567
```

A p-value below a threshold (such as 5% or 1%) suggests we reject the null hypothesis (stationary), otherwise, a p-value above the threshold suggests we fail to reject the null hypothesis (non-stationary).

We also perform the Stationarity test for each dataset looking at the results of the means and variance.The results shows are shown below.

```
Malaybalay DOST dataset:
1mean1=7.022192,mean2=7.529056
variance1=202.394168,variance2=205.186810
ADF Statistic: -13.122388
p-value: 0.000000
Critical Values:
         1%: -3.431
         5%: -2.862
         10%: -2.567

LARC-NASA dataset:
mean1=4.465135,mean2=6.506122
variance1=50.009078,variance2=59.453744
ADF Statistic: -11.185995
p-value: 0.000000
Critical Values:
         1%: -3.431
         5%: -2.862
         10%: -2.567
```

The mean values for Malaybalay are practically the same. Hence, Malaybalay rainfall is therefore stationary. There is a considerable difference in the mean values for the LARC-NASA dataset however, the p-values prevail indicative of it being stationary as well.

4. CONCLUSIONS

The result of computational tests of rainfall data of Malaybalay, Bukidnon, showed that the 56 years of rainfall is neither seasonal nor periodic. Furthermore, the examination of means and variance points to rain as being stationary. The visualization using a heat map shows no evidence of the same rainfall intensity from 1961 to 2017. Even with some indication of more significant rain in the later years that could reach under 200mm, it is not a strong case for a sign of climate change. The unavailability of the corresponding temperature on the same span of years as rainfall did permit the authors to investigate the temporal variation of temperature to support the cause of climate change detection.

REFERENCES

DOST, Department of Science and Technology, Philippine Atmospheric, Geophysical and Astronomical Services Administration
(https://www.pagasa.dost.gov.ph/climate/climate-data)
(20 October 2021)

Climate Change in the Philippines, DOST, Department of Science and Technology, Philippine Atmospheric, Geophysical and Astronomical Services Administration (https://www.pagasa.dost.gov.ph/information/climate-change-in-the-philippines) (20 October 2021)

PhilAtlas
Malaybalay, Province of Bukidnon
(https://www.philatlas.com/mindanao/r10/bukidnon/malaybalay.html) (20 October 2021)

Anaconda
Python 3.8 64-bit (Anaconda Individual Edition Version)
(https://www.anaconda.com/products/individual)
(18 October 2021)

Brownlee, J., 2021: *Deep Learning for Time Series Forecasting* (https://machinelearningmastery.com/deep-learning-for-time-series-forecasting/) (20 October 2021)

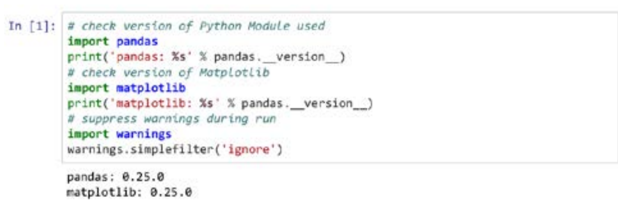
ImageJ
64-bit Java 1.8.0_172 (Platform Independent version) (https://imagej.nih.gov/ij/download.html) (20 October 2021)

Proakis J., Manolakis D.G, 1995: Digital Signal Processing: Principle, Algorithms, and Applications. Prentice-Hall 3rd Ed.

Data Access Viewer – NASA Power
(https://power.larc.nasa.gov/data-access-viewer/)
(6 November 2021)

APPENDIX

Loading and checking Python modules



```
In [1]: # check version of Python Module used
        import pandas
        print('pandas: %s' % pandas.__version__)
        # check version of Matplotlib
        import matplotlib
        print('matplotlib: %s' % pandas.__version__)
        # suppress warnings during run
        import warnings
        warnings.simplefilter('ignore')

        pandas: 0.25.0
        matplotlib: 0.25.0
```


https://doi.org/10.5194/isprs-archives-XLVI-4-W6-2021-71-2021

Reading in rainfall data

```
In [2]: from pandas import Series
        from pandas import read_csv
        from matplotlib import pyplot
        series = read_csv('Malaybalay_Daily_Total-Rainfall_(1961-2017)DataPrep.csv',he
        ader=0, index_col=0,
                          parse_dates=True, squeeze=True)
        print(series.head())

        DATE
        1961-01-01    29.0
        1961-01-02    16.8
        1961-01-03    14.2
        1961-01-04    10.7
        1961-01-05     1.3
        Name: RAINFALL, dtype: float64
```

Calculate the number of observations from data

```
In [3]: # summarize the dimensions of a time series
        from pandas import read_csv
        series = read_csv('Malaybalay_Daily_Total-Rainfall_(1961-2017)DataPrep.csv',
                          header=0, index_col=0, parse_dates=True,
        squeeze=True)
        print(series.size)
```

Calculate descriptive statistics in Python

```
In [6]: # calculate descriptive statistics
        from pandas import read_csv
        series = read_csv('Malaybalay_Daily_Total-Rainfall_(1961-2017)DataPrep.csv',
                          header=0, index_col=0, parse_dates=True,
        squeeze=True)
        print(series.describe())

        count    20805.000000
        mean         7.127195
        std         13.995723
        min         -1.000000
        25%          0.000000
        50%          1.000000
        75%          7.800000
        max        195.900000
        Name: RAINFALL, dtype: float64
```

Create the historical rainfall line plot

```
In [7]: # create a line plot
        from pandas import read_csv
        from matplotlib import pyplot
        from matplotlib.pyplot import figure
        figure(num=None, figsize=(12, 6), dpi=80, facecolor='w', edgecolor='k')

        series = read_csv('Malaybalay_Daily_Total-Rainfall_(1961-2017)DataPrep.csv',
                          header=0, index_col=0, parse_dates=True, squeeze=True)
        series.plot()
        pyplot.show()
```

Create stacked line plots of temporal yearly rainfall

```
In [9]: # create stacked line plots
        from pandas import read_csv
        from pandas import DataFrame
        from pandas import Grouper
        from matplotlib import pyplot

        series = read_csv('Malaybalay_Daily_Total-Rainfall_(1961-2017)DataPrep.csv',
                          header=0, index_col=0,parse_dates=True, squeeze=True)
        groups = series.groupby(Grouper(freq='A'))
        years = DataFrame()

        for name, group in groups:
            years[name.year] = group.values
        years.plot(subplots=True, legend=False,figsize=(15,28))
        pyplot.show()
```

Subsetting dates of interest from rainfall data

```
In [4]: # query a dataset using a date-time index
        from pandas import read_csv
        series = read_csv('Malaybalay_Daily_Total-Rainfall_(1961-2017)DataPrep.csv',
                          header=0, index_col=0, parse_dates=True, squeeze=True)
        print(series['1961-01'])
```

Create a heat map of yealy rainfall data

```
In [15]: # create a heat map of yearly data
         from pandas import read_csv
         from pandas import DataFrame
         from pandas import Grouper
         from matplotlib import pyplot
         series = read_csv('Malaybalay_Daily_Total-Rainfall_(1961-2017)DataPrep.csv',
                           header=0, index_col=0,parse_dates=True, squeeze=True)
         groups = series.groupby(Grouper(freq='A'))
         years = DataFrame()

         for name, group in groups:
             years[name.year] = group.values
         years = years.T
         pyplot.matshow(years, interpolation='gaussian', aspect='auto')
         pyplot.show()
```

Create a heat map of monthly rainfall data

```
In [16]: # create a heat map of monthly data
         from pandas import read_csv
         from pandas import DataFrame
         from pandas import Grouper
         from matplotlib import pyplot
         from pandas import concat

         series = read_csv('Malaybalay_Daily_Total-Rainfall_(1961-2017)DataPrep.csv', h
         eader=0, index_col=0,
         parse_dates=True, squeeze=True)
         one_year = series['1961']
         groups = one_year.groupby(Grouper(freq='M'))
         months = concat([DataFrame(x[1].values) for x in groups], axis=1)
         months = DataFrame(months)
         months.columns = range(1,13)
         pyplot.matshow(months, interpolation='gaussian', aspect='auto')
         pyplot.show()
```

Test for Stationarity by Means and Variances

```
[13]: # we split the rainfall data and compare their means and variances
      from pandas import read_csv
      series = read_csv('Malaybalay_Daily_Total-Rainfall_(1961-2017)@365-Day.csv',
                        header=0, index_col=0, parse_dates=True, squeeze=True)
      X = series.values
      split = len(X) // 2 # use // for Python 3.x for integer result
      X1, X2 = X[0:split], X[split:]
      mean1, mean2 = X1.mean(), X2.mean()
      var1, var2 = X1.var(), X2.var()
      print('mean1=%f, mean2=%f' % (mean1, mean2))
      print('variance1=%f, variance2=%f' % (var1, var2))

      mean1=6.943857, mean2=7.334317
      variance1=186.006267, variance2=205.297065
```

Augmented Dickey-Fuller Statistical Test

```
[14]: # for the use case of airline passenger records
      from pandas import read_csv
      from statsmodels.tsa.stattools import adfuller
      series = read_csv('Malaybalay_Daily_Total-Rainfall_(1961-2017)@365-Day.csv',
                        header=0, index_col=0, parse_dates=True,squeeze=True)

      X = series.values
      result = adfuller(X)

      print('ADF Statistic: %f' % result[0])
      print('p-value: %f' % result[1])
      print('Critical Values:')

      for key, value in result[4].items():
          print('\t%s: %.3f' % (key, value))
```

Autocorrelation Function

```
[47]: # plot the autocorrelation of any weather variable input on the parenthesis
      from matplotlib.pyplot import figure
      figure(num=None, figsize=(5, 2), dpi=80, facecolor='w', edgecolor='k')
      autocorrelation_plot(rainfall)
      pyplot.show()
```

Source Code for Complete Time Series Analysis Run

11/8/21, 11:31 PM Time Series And Spatial Analysis 56 years Historical Rainfall Data Malaybalay - Jupyter Notebook

In [55]:

```
# Point to the AWS weather file
import pandas as pd
from pandas import read_csv

# read the AWS data
series=read_csv('D:/Users/Berns/envs/Malaybalay MTH FSD Weather Time Series Analy
# make the dataframe and slice for the weather variable of interest
df=pd.DataFrame(series)

# Rainfall Autocorrelation
weather_var=df[['LARC-NASA_HUM']]
print(weather_var)
solar_rad=pd.DataFrame(weather_var)
solar_rad.plot()

# check for periodicity
figure(num=None, figsize=(5,4), dpi=80, facecolor='w', edgecolor='k')
autocorrelation_plot(solar_rad)
pyplot.show()

# Rainfall stationarity by test of means and variances
df=pd.DataFrame(series)
weather_var=df['LARC-NASA_HUM']
df = pd.DataFrame(weather_var)
X=df.values
split=len(X) // 2 # use for Python 3.x for integer result
X1,X2 = X[0:split], X[split:]
mean1, mean2 = X1.mean(), X2.mean()
var1, var2 = X1.var(), X2.var()
print('mean1=%f,mean2=%f' % (mean1,mean2))
print('variance1=%f,variance2=%f' % (var1,var2))

# Rainfall time series test by Augmented Dickey Fuller method
temp=df['LARC-NASA_HUM']
X=temp.values
result=adfuller(X)

# show results
print('ADF Statistic: %f' % result[0])
print('p-value: %f' % result[1])
print('Critical Values:')

for key, value in result[4].items():
    print('\t%s: %.3f' % (key, value))
```


https://doi.org/10.5194/isprs-archives-XLVI-4-W6-2021-71-2021 | © Author(s) 2021. CC BY 4.0 License.