Time Series Anomalies

Milestone 3 - 11/5/2016

Group: Dennis Milechin, Ivan Sinyagin, Hany Bassily, Edgar Calcaneo
TF: Zelong

NOAA Buoy Data

The data considered represents the climatological data gathered from ten Ocean Buoy located in the western Atlantic. The ten Buoy spans the region between southern Bermuda to Puerto Rico. The data includes atmospheric and oceanic information. The buoy considered in this study are owned by NOAA and the National Weather Service. However, there are other buoy operating in the same area and owned by different parties. In actual studies, all the data from all the buoys are considered which make this data accessible to all concerned entities. For this project, we are considering the NOAA buoys as their data are available to the public and the contained information is sufficient to the project objective. The following picture depicts how the buoy looks like

The significance of this data comes from the location of the buoy as they are all located in a very active region in terms of storms and hurricanes.

Data structure

Each buoy data is divided in separate files where each file corresponds to a specific month. The period considered is between January 2016 and September 2016. The features in the data includes:

- Wind Direction
- Wind Speed
- Wind Gust
- Wave Height
- Dominant Wave Period
- Average Period
- Mean Wave Direction
- Atmospheric Pressure
- Pressure Tendency
- Air Temperature
- Water Temperature
- Wind Speed at 10 Meters
- Wind Speed at 20 Meters
- And the observations are sampled on an hourly basis

Data Usability

Based on the information included in the data, the data is suitable to study climatic changes near the buoy. It can also be used to monitor the evolution of a storm system and it can also be a good basis to evaluate the buoy vandalizme

Data Conditioning

The data downloaded consists of 10 bouys located south of Bermuda to Puerto Rico in the western portion of the Atlantic. Each bouy data consists of nine text files, that represent each month of data collected from January to September of 2016. For each sensor, missing data is designated as series of 9's, such as 99.0 or 9999.0. These values are converted into null values during the importation of the data. Otherwise, no other conditioning is done.

Data Exploration

The following are some basic data descriptors for the entire dataset.

In [189]:
```
print "Bouy data was downloaded for the following bouy IDs:\n"

for bouy in bouy_data['ID'].unique():
    print bouy

print ""
print "Date Range: ", bouy_data['DATETIME'].min(), ' to ', bouy_data['DATETIM
E'].max()

print "Total Number of Records: ", bouy_data.shape[0]
```

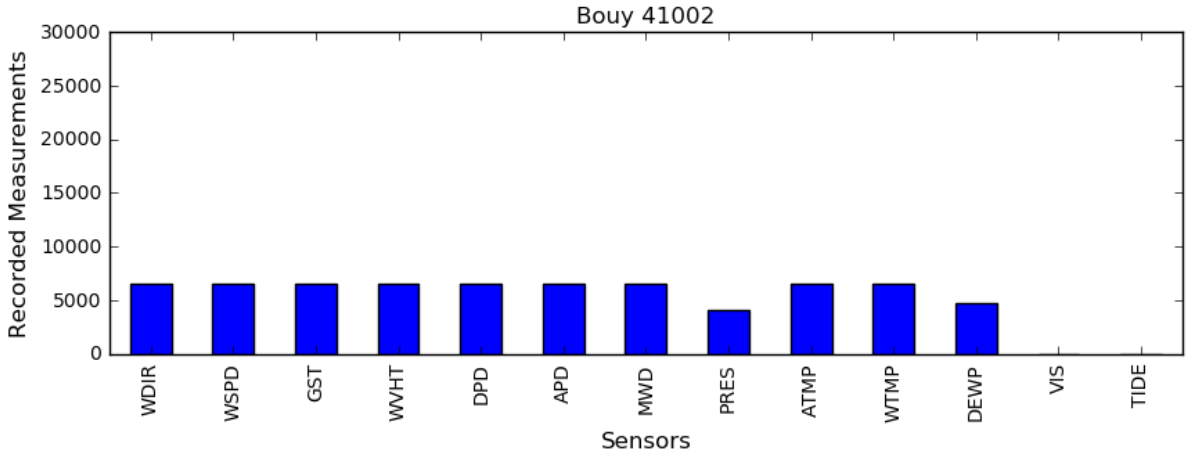
```
Bouy data was downloaded for the following bouy IDs:

41002
41040
41041
41043
41044
41046
41047
41048
41049
42059

Date Range:  2015-12-31 23:50:00  to  2016-09-30 23:50:00
Total Number of Records:  152340
```

Descriptive Statistics by Bouy

In [268]:
```
pd.set_option('precision', 1)

for bouy in bouy_data['ID'].unique():

    display(HTML("<h2><center> Descriptive Statistics for Bouy ID: " + str(bou
y) + "</center></h2>"))


    subset = bouy_data[bouy_data['ID'] == bouy][[u'WDIR', u'WSPD', u'GST', u'W
VHT', u'DPD', u'APD', u'MWD',
       u'PRES', u'ATMP', u'WTMP', u'DEWP', u'VIS', u'TIDE']].describe()

    print subset[[u'WDIR', u'WSPD', u'GST', u'WVHT', u'DPD', u'APD', u'MWD',
       u'PRES', u'ATMP', u'WTMP', u'DEWP']].loc[[u'mean', u'std', u'min', u'ma
x']]

    ax = subset.loc['count'].plot(kind='bar', figsize=(10, 3), title ="Bouy "
+ str(bouy))
    ax.set_xlabel("Sensors", fontsize=12)
    ax.set_ylabel("Recorded Measurements", fontsize=12)
    ax.set_ylim([0,30000])
    plt.show()
```

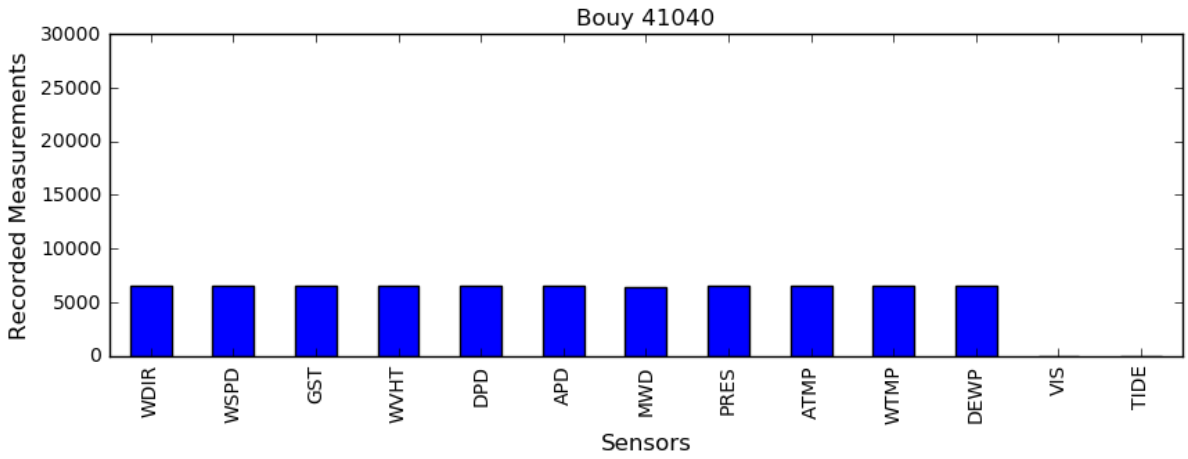
Descriptive Statistics for Bouy ID: 41002

	WDIR	WSPD	GST	WVHT	DPD	APD	MWD	PRES	ATMP	WTMP	DEWP
mean	186.4	6.5	8.0	1.7	8.3	5.8	139.5	1017.6	22.8	24.4	20.2
std	87.3	3.2	3.9	1.0	2.3	1.1	91.6	3.7	4.8	3.5	5.4
min	1.0	0.0	0.2	0.5	2.9	3.6	1.0	998.2	8.5	19.3	0.0
max	360.0	25.6	32.8	9.2	16.0	10.4	360.0	1025.2	29.7	32.2	27.4



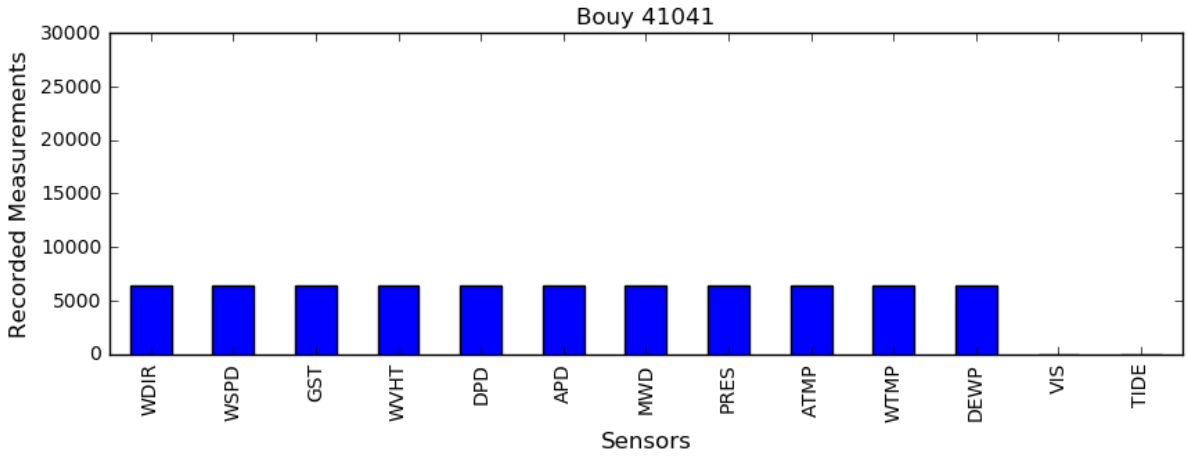
Descriptive Statistics for Bouy ID: 41040

	WDIR	WSPD	GST	WVHT	DPD	APD	MWD	PRES	ATMP	WTMP	DEWP
mean	82.1	7.1	8.6	2.0	9.2	6.2	89.0	1014.8	26.7	27.4	22.4
std	31.1	1.9	2.2	0.5	2.1	0.9	62.5	1.8	1.1	1.0	1.4
min	1.0	0.2	0.6	0.9	4.3	4.4	1.0	1008.0	22.4	25.2	16.9
max	359.0	15.5	19.3	4.0	21.1	11.9	359.0	1020.2	29.1	30.6	26.0



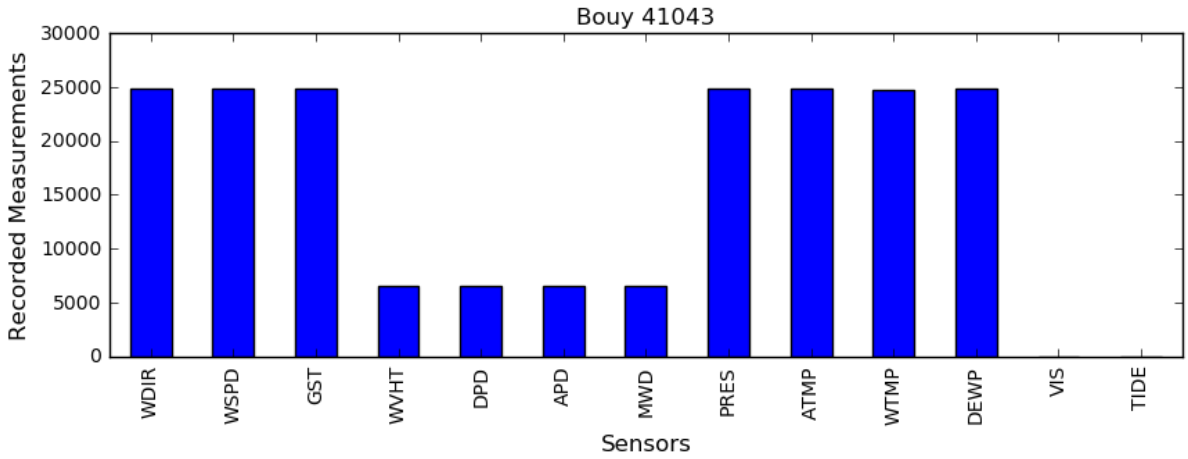
Descriptive Statistics for Bouy ID: 41041

	WDIR	WSPD	GST	WVHT	DPD	APD	MWD	PRES	ATMP	WTMP	DEWP
mean	76.7	7.2	8.7	2.0	9.4	6.2	88.9	1015.3	25.8	26.5	21.5
std	36.0	1.8	2.2	0.5	2.3	0.9	78.8	1.9	1.0	1.0	1.6
min	1.0	0.5	1.1	0.9	4.8	4.5	1.0	1009.5	22.3	25.0	15.5
max	359.0	11.9	15.7	4.8	21.1	11.8	359.0	1020.2	28.7	30.3	25.5



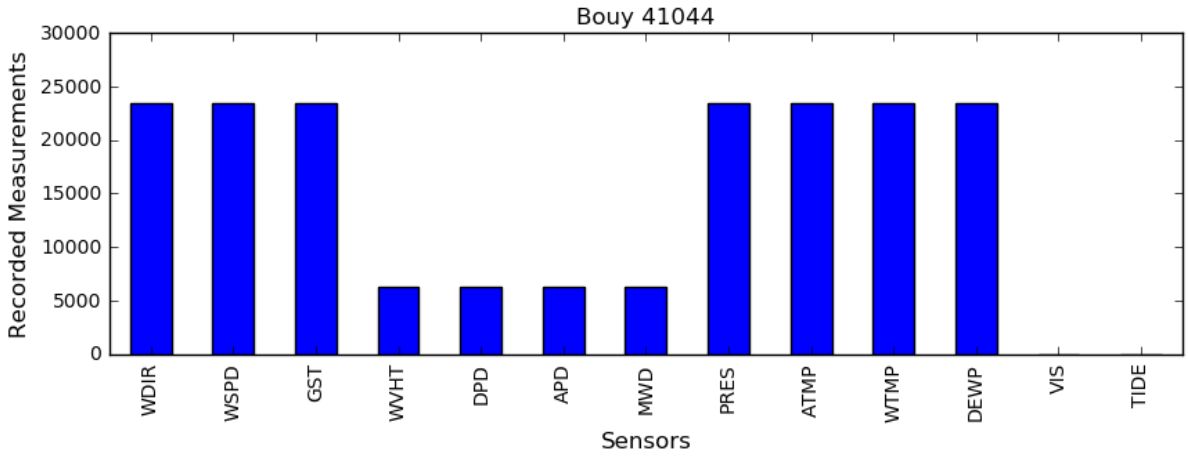
Descriptive Statistics for Bouy ID: 41043

	WDIR	WSPD	GST	WVHT	DPD	APD	MWD	PRES	ATMP	WTMP	DEWP
mean	95.6	6.1	7.7	1.7	9.2	6.2	103.8	1017.4	27.6	28.3	23.5
std	36.2	1.9	2.3	0.5	2.1	1.2	79.2	1.8	1.2	1.0	1.6
min	0.0	0.1	0.4	0.8	4.2	4.2	0.0	1009.5	21.9	26.0	12.2
max	359.0	16.1	20.7	4.4	19.1	12.6	359.0	1023.3	29.9	30.7	26.5



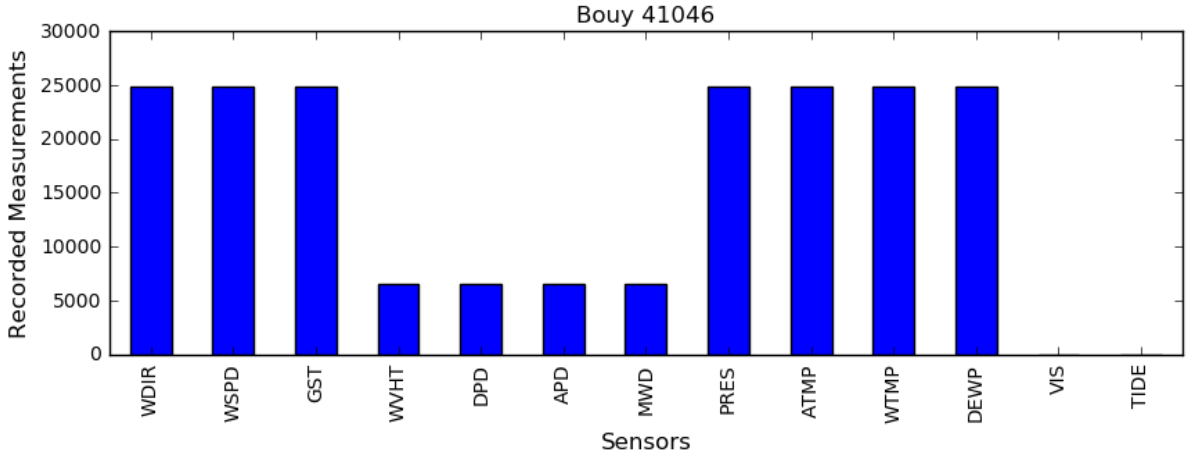
Descriptive Statistics for Bouy ID: 41044

	WDIR	WSPD	GST	WVHT	DPD	APD	MWD	PRES	ATMP	WTMP	DEWP
mean	93.7	6.1	7.7	1.8	9.4	6.3	112.0	1018.6	27.3	27.9	23.2
std	37.4	1.9	2.2	0.5	2.2	1.1	90.9	2.3	1.2	1.1	1.7
min	1.0	0.0	0.6	0.3	4.3	4.5	1.0	1007.3	21.3	25.1	12.3
max	360.0	16.2	22.5	4.1	19.1	12.7	359.0	1024.6	29.4	29.9	26.4



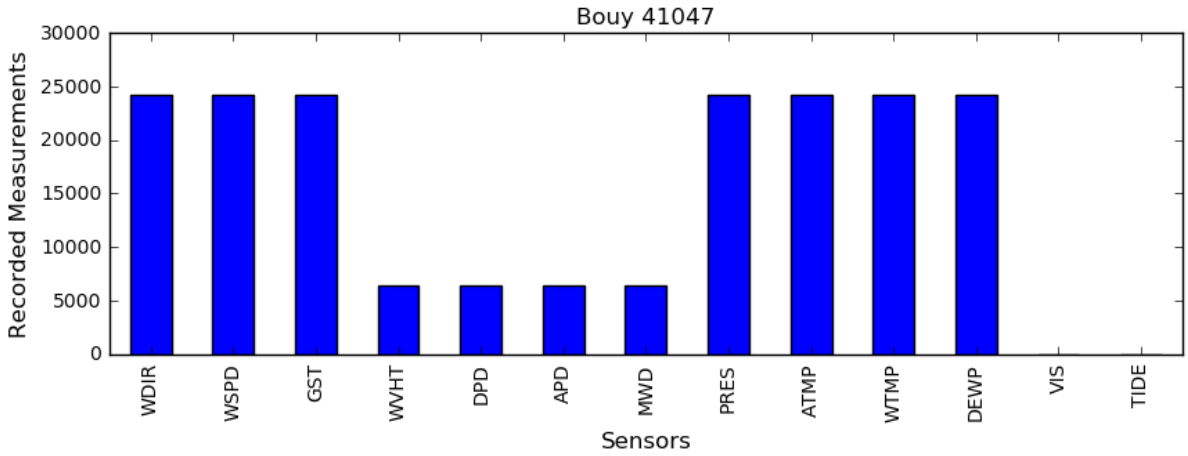
Descriptive Statistics for Bouy ID: 41046

	WDIR	WSPD	GST	WVHT	DPD	APD	MWD	PRES	ATMP	WTMP	DEWP
mean	114.0	5.5	7.0	1.6	9.0	6.0	106.5	1018.0	27.5	28.3	23.5
std	55.5	2.0	2.3	0.6	2.1	1.2	80.3	2.2	1.7	1.4	2.2
min	1.0	0.0	0.1	0.6	3.6	3.9	0.0	1010.2	20.7	25.2	10.2
max	360.0	15.1	19.1	5.1	16.0	12.6	359.0	1025.7	30.3	31.2	27.0



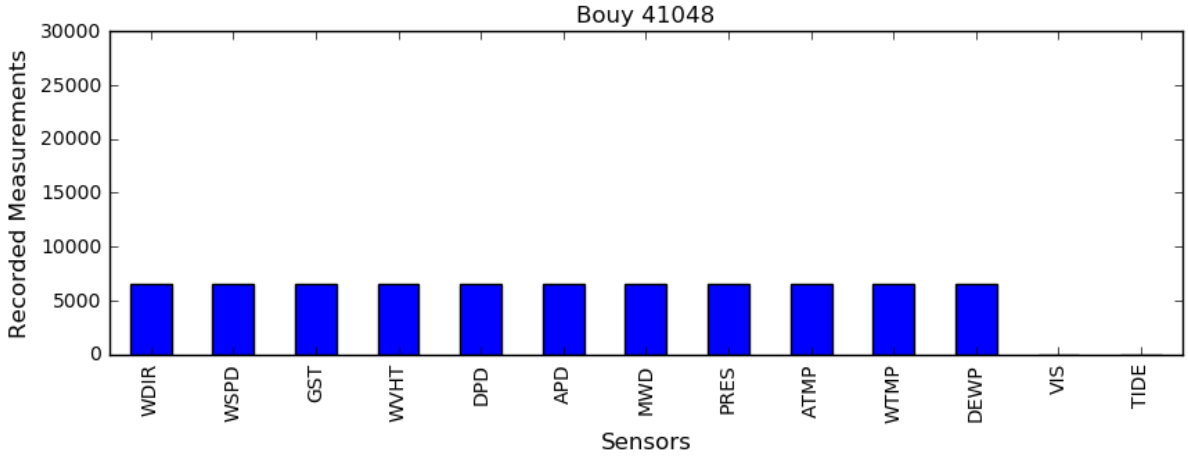
Descriptive Statistics for Bouy ID: 41047

	WDIR	WSPD	GST	WVHT	DPD	APD	MWD	PRES	ATMP	WTMP	DEWP
mean	148.3	4.4	5.7	1.6	9.0	6.2	119.3	1018.3	27.0	28.1	22.5
std	78.5	2.3	2.7	0.8	2.0	1.0	81.4	2.8	2.4	2.2	3.0
min	0.0	0.0	0.0	0.5	3.2	4.0	0.0	1004.0	17.7	22.7	7.7
max	360.0	16.5	23.3	6.5	17.4	11.4	359.0	1028.6	30.1	31.1	26.6



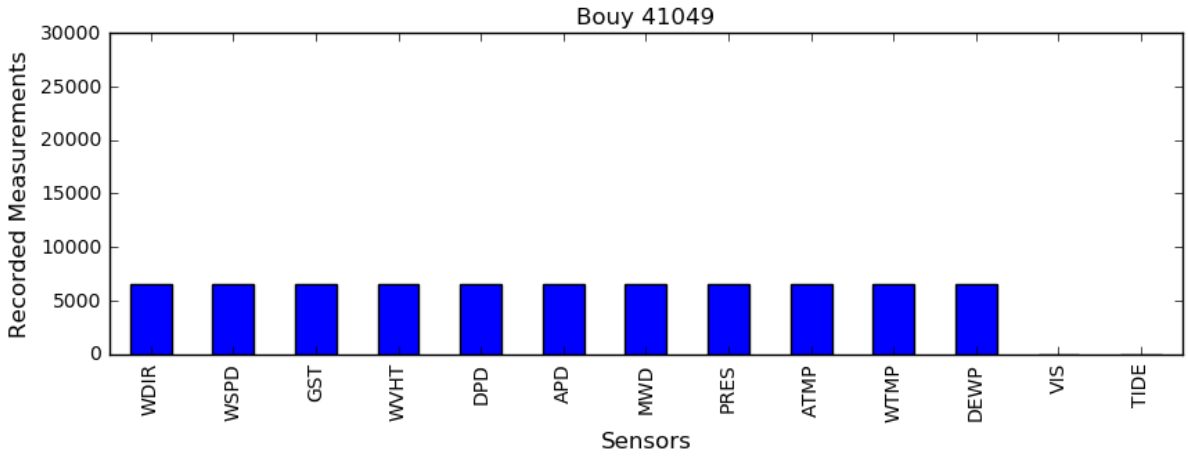
Descriptive Statistics for Bouy ID: 41048

	WDIR	WSPD	GST	WVHT	DPD	APD	MWD	PRES	ATMP	WTMP	DEWP
mean	191.1	6.2	7.7	1.8	8.8	6.2	154.6	1018.0	23.2	24.3	18.9
std	90.7	3.4	4.0	1.1	2.1	1.1	100.6	4.9	4.1	3.6	5.6
min	1.0	0.0	0.0	0.5	3.6	3.8	1.0	988.3	11.9	20.0	3.5
max	359.0	24.2	29.5	12.1	17.4	12.0	359.0	1031.5	29.9	32.1	27.4



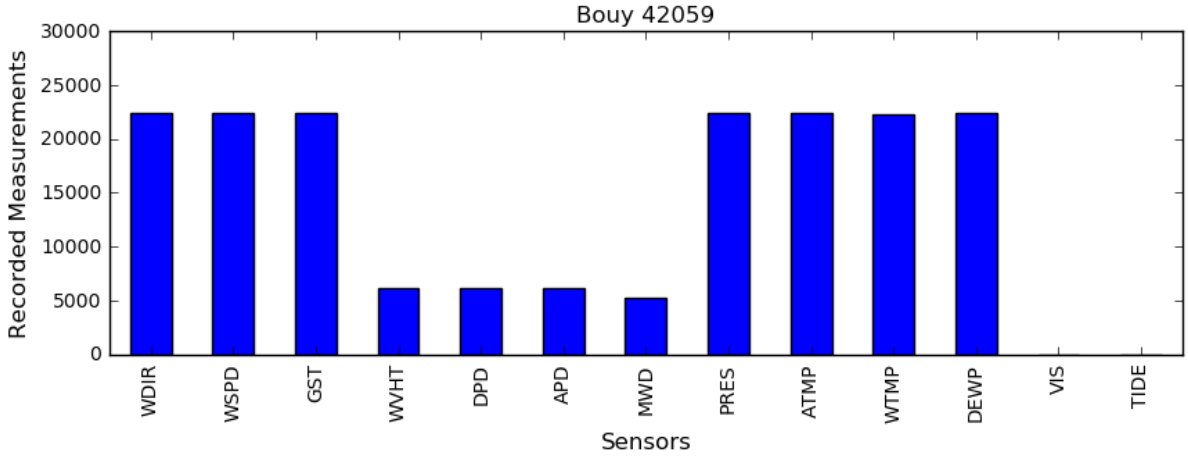
Descriptive Statistics for Bouy ID: 41049

	WDIR	WSPD	GST	WVHT	DPD	APD	MWD	PRES	ATMP	WTMP	DEWP
mean	155.7	5.5	6.8	1.8	9.2	6.5	136.8	1019.7	25.2	26.1	20.7
std	87.6	2.6	3.1	0.8	2.1	1.1	102.2	3.3	2.8	2.7	3.4
min	1.0	0.0	0.2	0.6	3.9	4.2	1.0	1005.4	18.0	22.4	8.7
max	359.0	15.7	19.5	5.7	17.4	11.8	359.0	1029.5	30.3	33.1	25.8



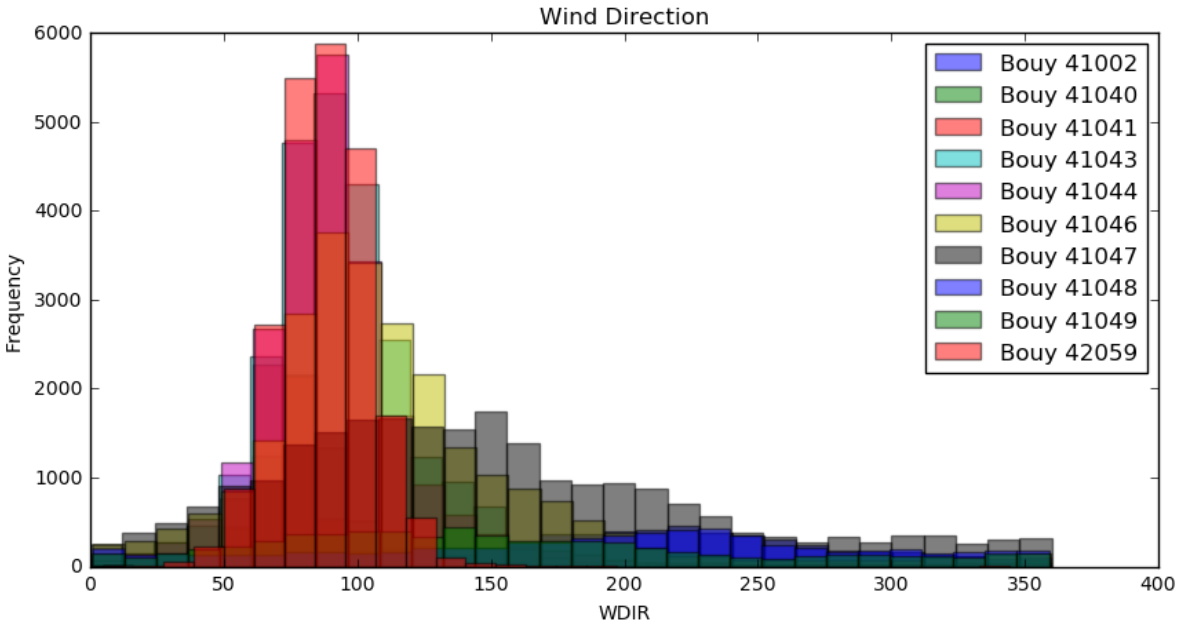
Descriptive Statistics for Bouy ID: 42059

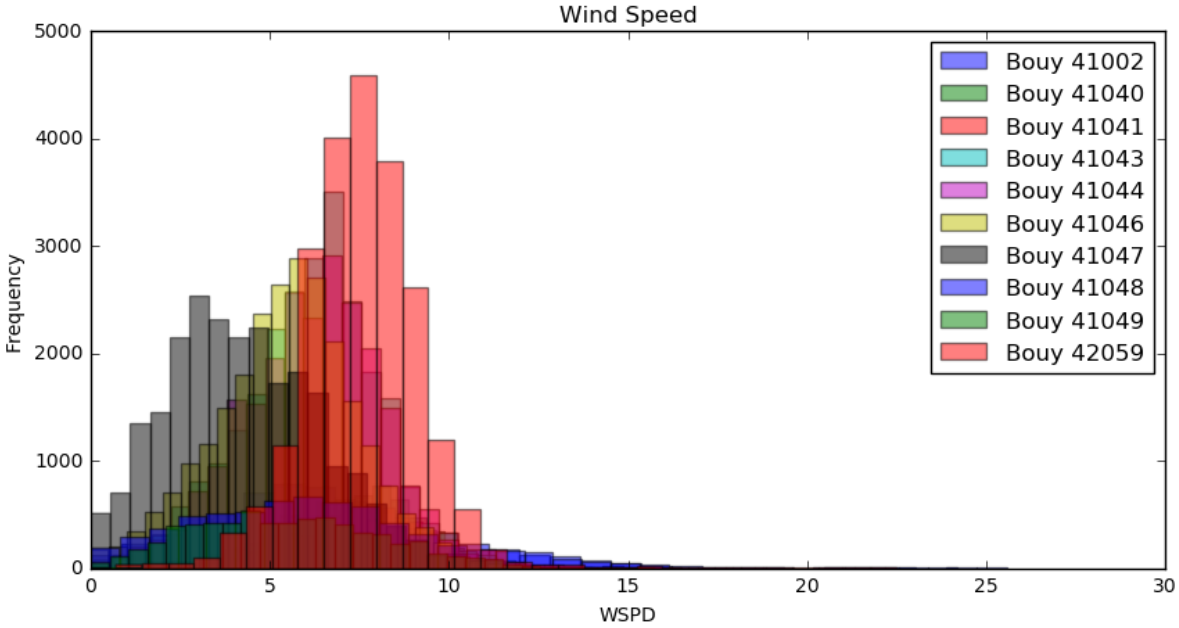
	WDIR	WSPD	GST	WVHT	DPD	APD	MWD	PRES	ATMP	WTMP	DEWP
mean	87.7	7.6	9.4	1.6	7.1	5.2	94.6	1014.6	28.4	28.7	25.1
std	16.6	1.7	2.1	0.5	1.8	0.6	31.1	1.8	0.8	0.7	1.2
min	5.0	0.7	1.6	0.6	3.7	3.6	1.0	1004.8	24.2	27.0	17.3
max	344.0	22.5	31.4	9.2	16.0	10.1	360.0	1020.2	30.1	33.7	27.6

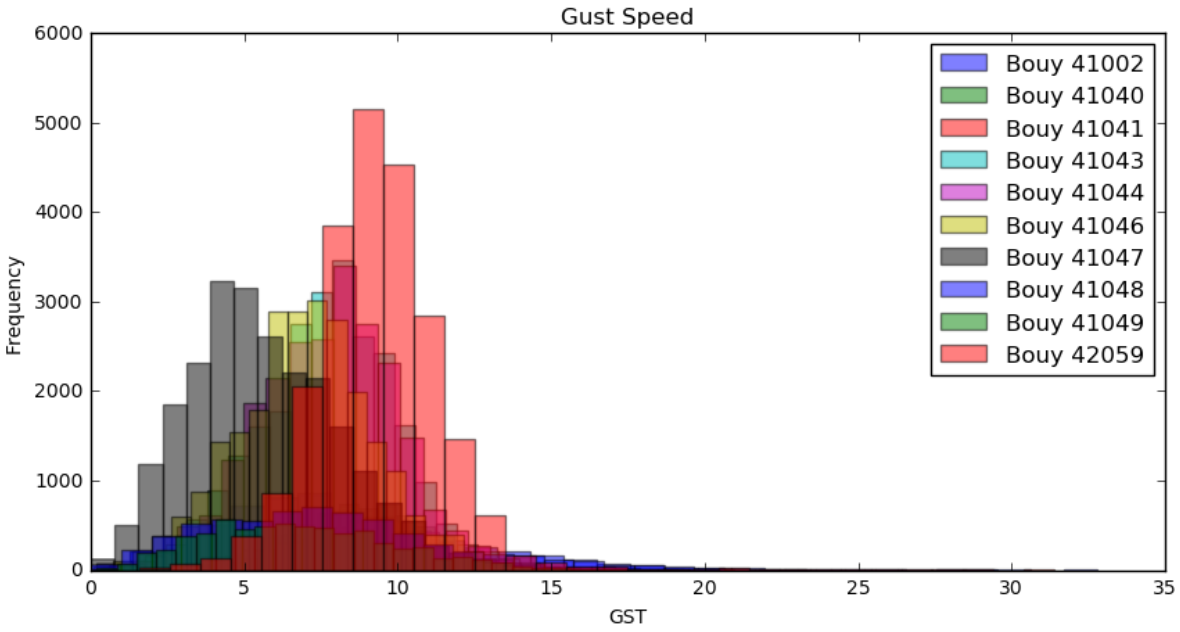


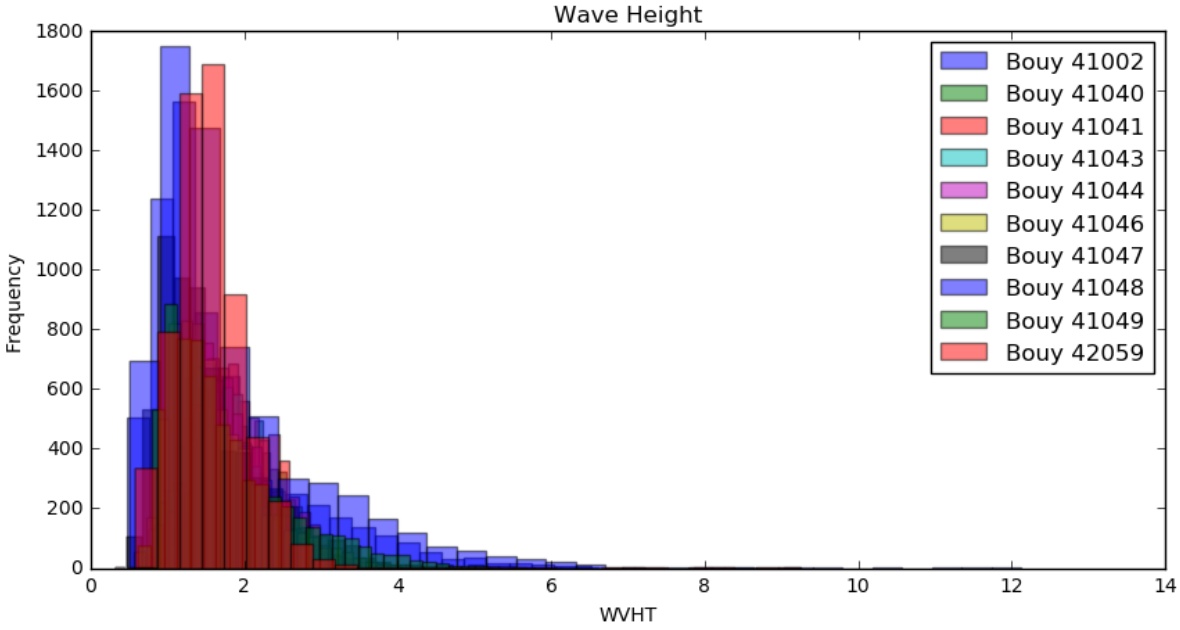
The histograms plots of measurements recorded for each sensor shows that all instruments do not have the same number of recorded measurements for a select bouy. In addition, each bouy doesn't have the same number of measurements when compared to each other. Some bouys have approaximately 25,000 data points, while some only have a little over 5,000. In addition, none of the bouys have records for Visability (VIS) or Tide sensors.

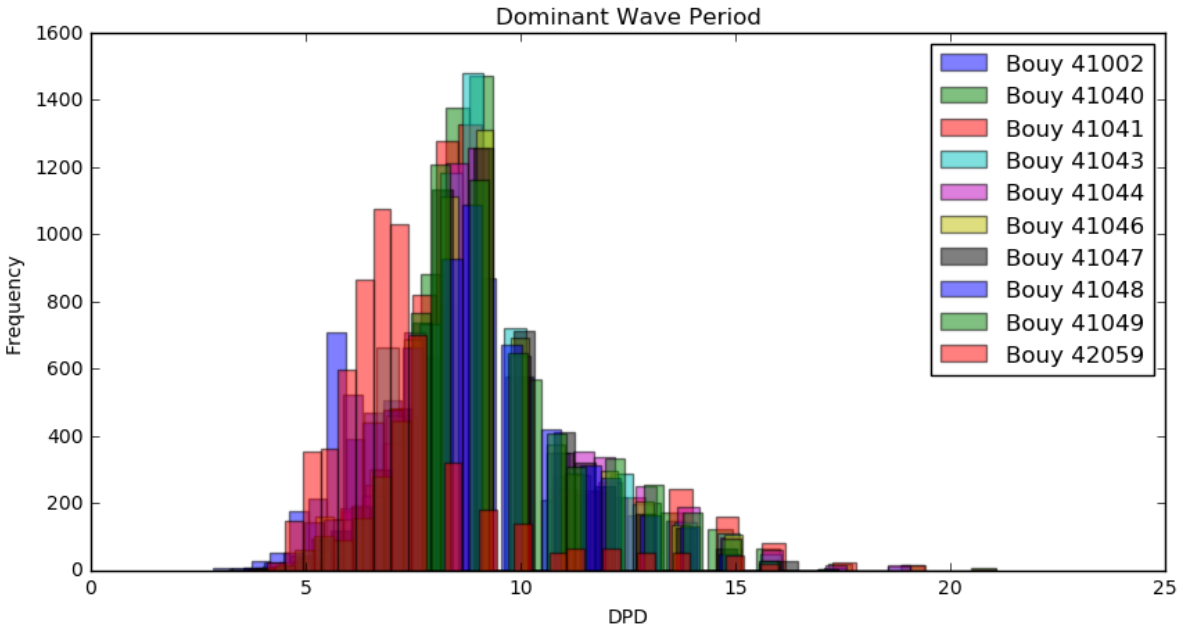
In [238]:
```
for sensor in [u'WDIR', u'WSPD', u'GST', u'WVHT', u'DPD', u'APD', u'MWD',u'PRE
S', u'ATMP', u'WTMP', u'DEWP']:

    fig, ax = plt.subplots(1, figsize=(10,5))

    for bouy in bouy_data['ID'].unique():
        data = bouy_data[bouy_data['ID'] == bouy][sensor].dropna()
        ax.hist(data, bins=30, alpha=0.5, label='Bouy ' + str(bouy))

    ax.set_ylabel('Frequency')
    ax.set_xlabel(sensor)
    plt.title(get_name(sensor))
    plt.legend(loc='best')
    plt.show()
```

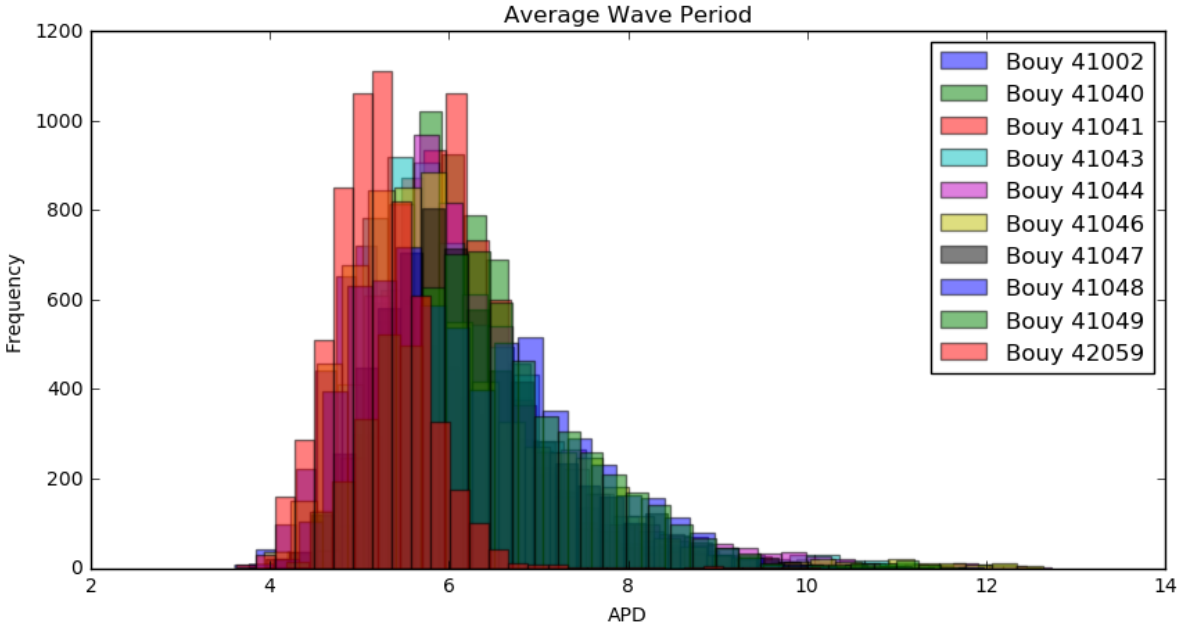

Wind Direction
Frequency
WDIR
0
1000
2000
3000
4000
5000
6000
0
50
100
150
200
250
300
350
400
Bouy 41002
Bouy 41040
Bouy 41041
Bouy 41043
Bouy 41044
Bouy 41046
Bouy 41047
Bouy 41048
Bouy 41049
Bouy 42059

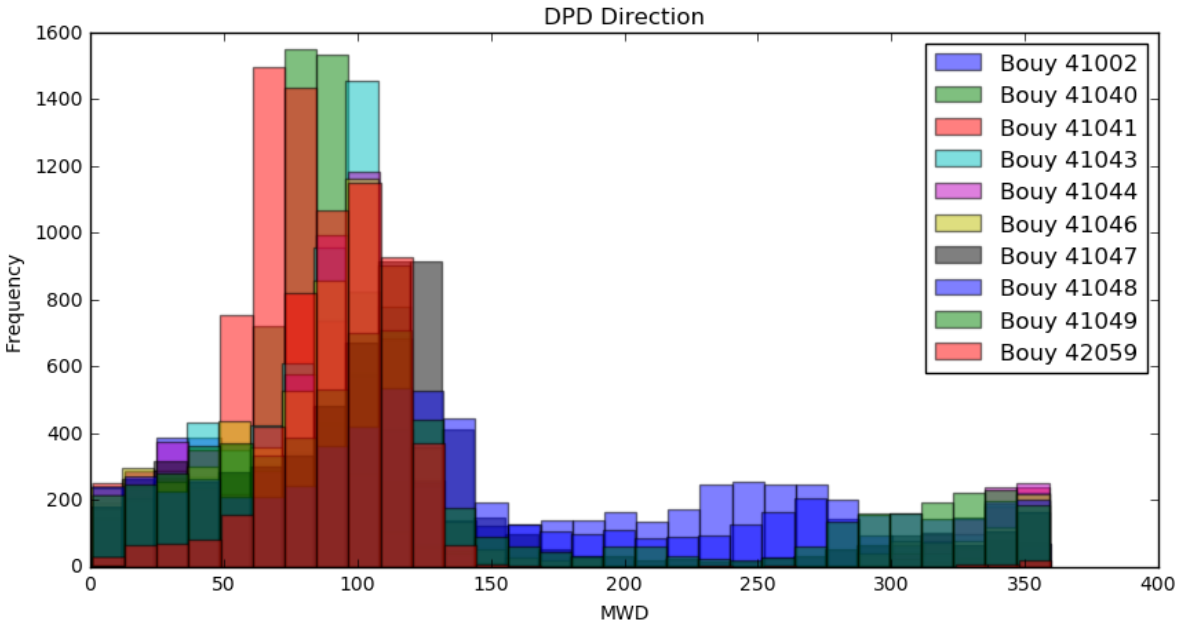

Wind Speed
Frequency
WSPD
0
1000
2000
3000
4000
5000
0
5
10
15
20
25
30
Bouy 41002
Bouy 41040
Bouy 41041
Bouy 41043
Bouy 41044
Bouy 41046
Bouy 41047
Bouy 41048
Bouy 41049
Bouy 42059

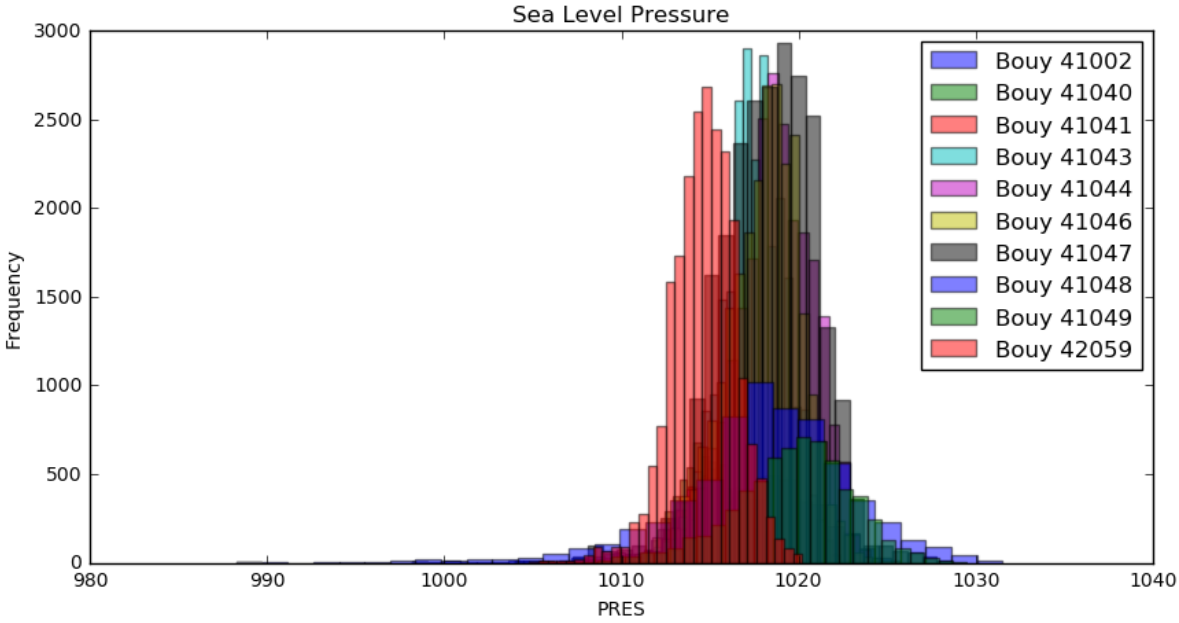

Gust Speed
Frequency
GST
Bouy 41002
Bouy 41040
Bouy 41041
Bouy 41043
Bouy 41044
Bouy 41046
Bouy 41047
Bouy 41048
Bouy 41049
Bouy 42059

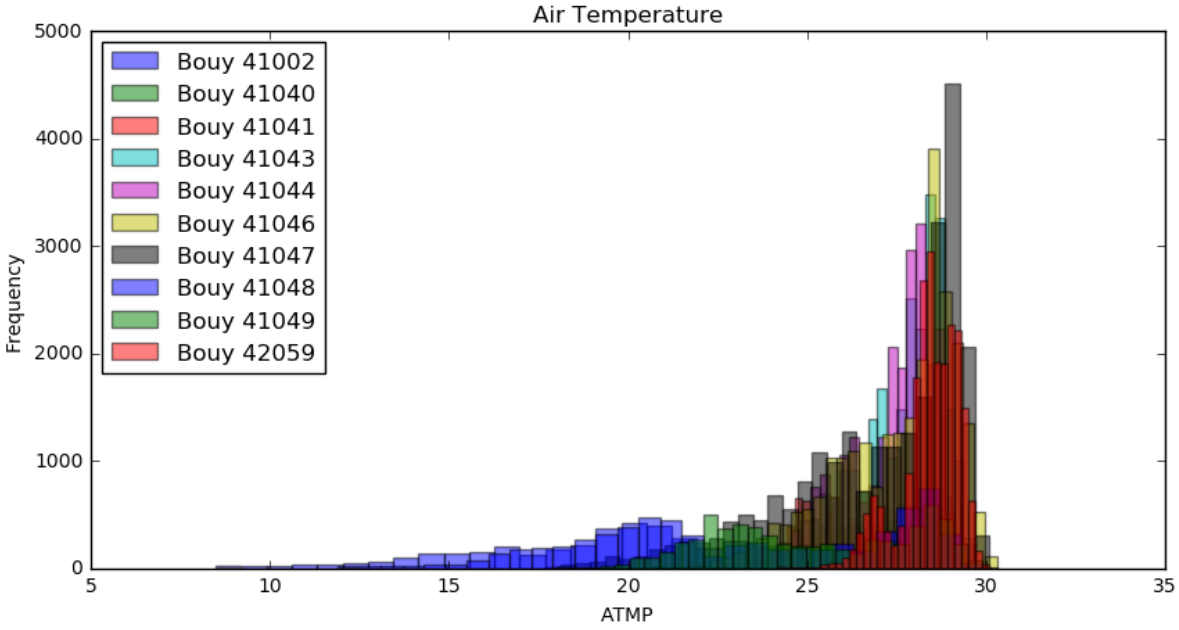

Wave Height
Frequency
WVHT
Bouy 41002
Bouy 41040
Bouy 41041
Bouy 41043
Bouy 41044
Bouy 41046
Bouy 41047
Bouy 41048
Bouy 41049
Bouy 42059

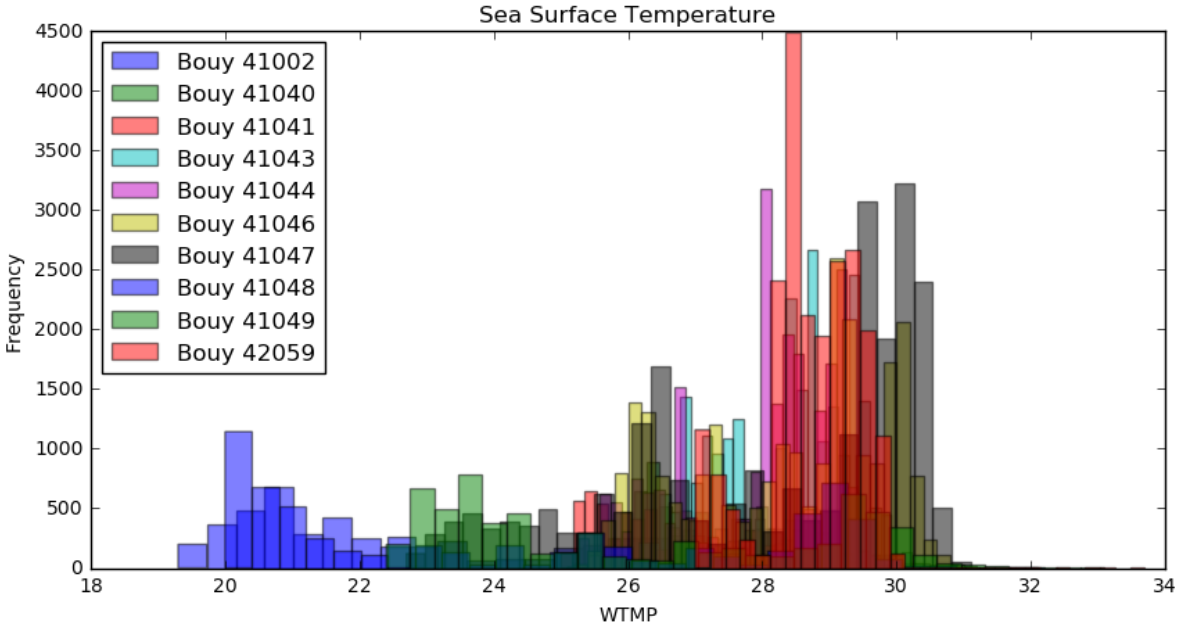

Dominant Wave Period
Frequency
DPD
Bouy 41002
Bouy 41040
Bouy 41041
Bouy 41043
Bouy 41044
Bouy 41046
Bouy 41047
Bouy 41048
Bouy 41049
Bouy 42059

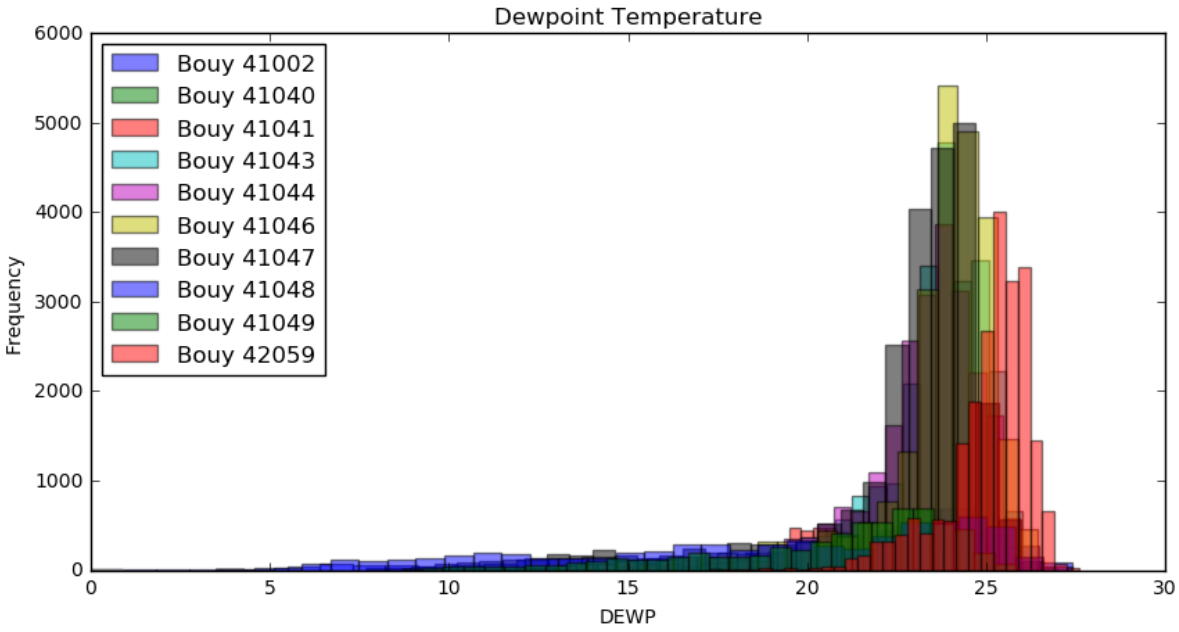

Average Wave Period
Frequency
APD
Bouy 41002
Bouy 41040
Bouy 41041
Bouy 41043
Bouy 41044
Bouy 41046
Bouy 41047
Bouy 41048
Bouy 41049
Bouy 42059


DPD Direction
Frequency
MWD
Bouy 41002
Bouy 41040
Bouy 41041
Bouy 41043
Bouy 41044
Bouy 41046
Bouy 41047
Bouy 41048
Bouy 41049
Bouy 42059


Sea Level Pressure
Frequency
PRES
Bouy 41002
Bouy 41040
Bouy 41041
Bouy 41043
Bouy 41044
Bouy 41046
Bouy 41047
Bouy 41048
Bouy 41049
Bouy 42059


Air Temperature
Bouy 41002
Bouy 41040
Bouy 41041
Bouy 41043
Bouy 41044
Bouy 41046
Bouy 41047
Bouy 41048
Bouy 41049
Bouy 42059
Frequency
ATMP
0
1000
2000
3000
4000
5000
5
10
15
20
25
30
35


Sea Surface Temperature
Bouy 41002
Bouy 41040
Bouy 41041
Bouy 41043
Bouy 41044
Bouy 41046
Bouy 41047
Bouy 41048
Bouy 41049
Bouy 42059
Frequency
WTMP
0
500
1000
1500
2000
2500
3000
3500
4000
4500
18
20
22
24
26
28
30
32
34

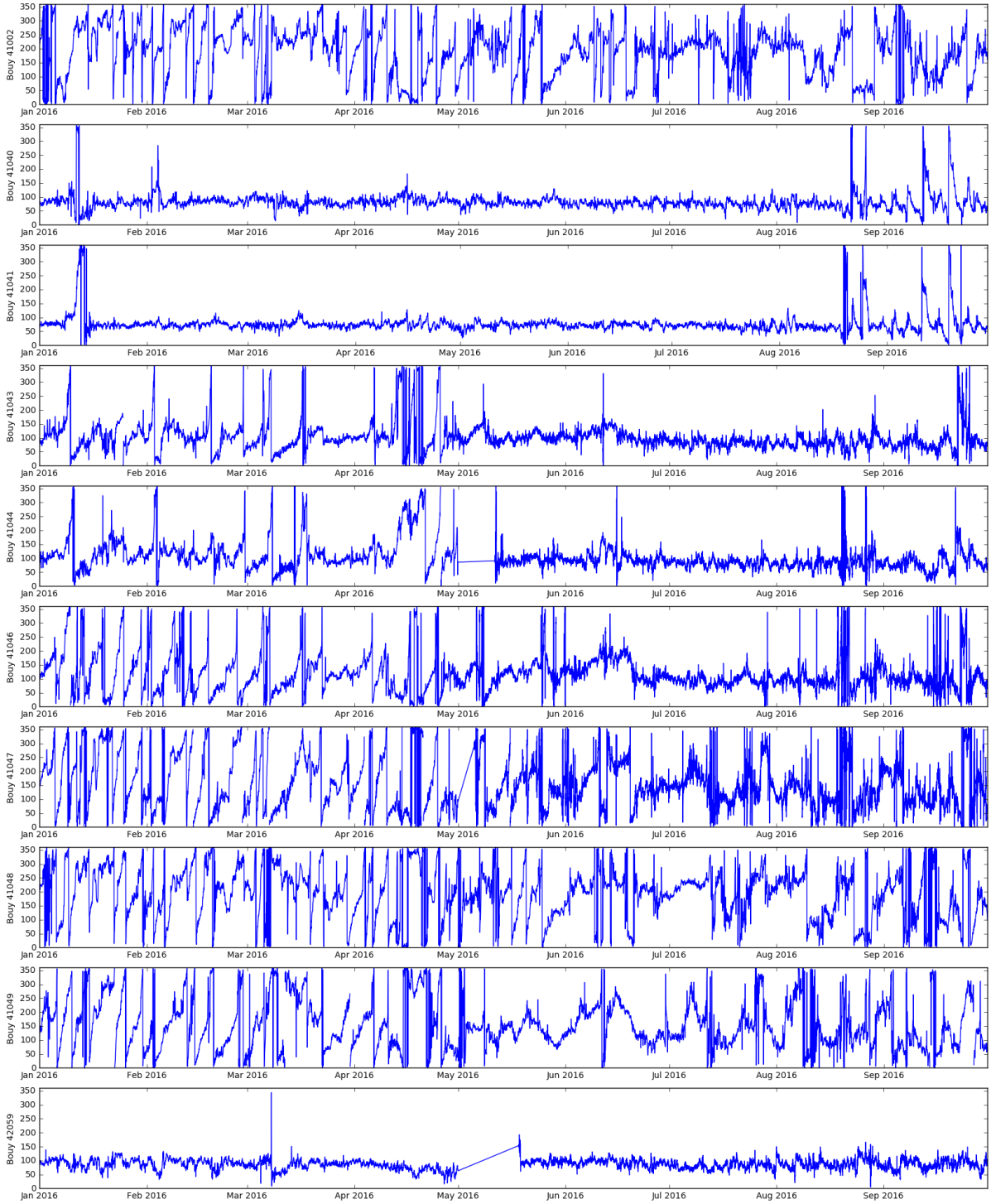


By looking at the histograms of of each sensor for all the bouys, we can learn a few things. For some sensors, the distribution of data follow a similiar pattern across all bouys, such as Sea Level Pressure and Wave Height. Some sensor data have normal distributions, such as Average Wave Period and Sea Level Period. Other sensors, such as Sea Surface Temperatures, don't seem to follow any pattern of a distribution or have a correlation on frequency with other bouys.

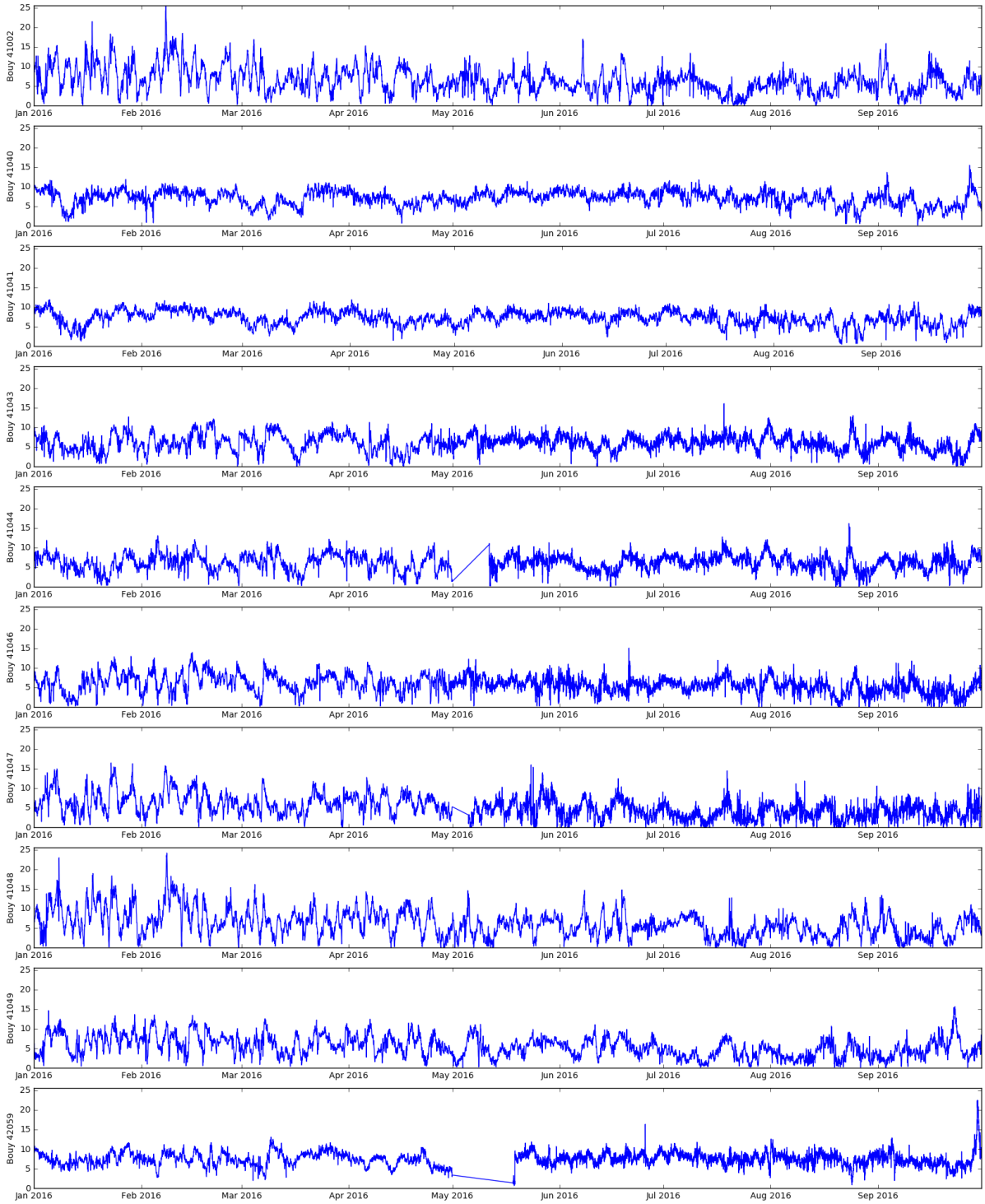
Time Series Plotting

In [265]:
```
for sensor in [u'WDIR', u'WSPD', u'GST', u'WVHT', u'DPD', u'APD', u'MWD',u'PRE
S', u'ATMP', u'WTMP', u'DEWP']:

    display(HTML("<h2><center>" + get_name(sensor) + "</center></h2>"))

    fig, ax = plt.subplots(len(bouy_data['ID'].unique()),1, figsize=(20,25))

    max_val = bouy_data[sensor].max()
    min_val = bouy_data[sensor].min()

    index = 0
    for bouy in bouy_data['ID'].unique():
        data = bouy_data[bouy_data['ID'] == bouy][[sensor, 'DATETIME']]
        ax[index].plot(data['DATETIME'], data[sensor])
        ax[index].set_ylabel('Bouy ' + str(bouy))
        ax[index].set_ylim([min_val,max_val])
        index = index + 1

    plt.show()
```

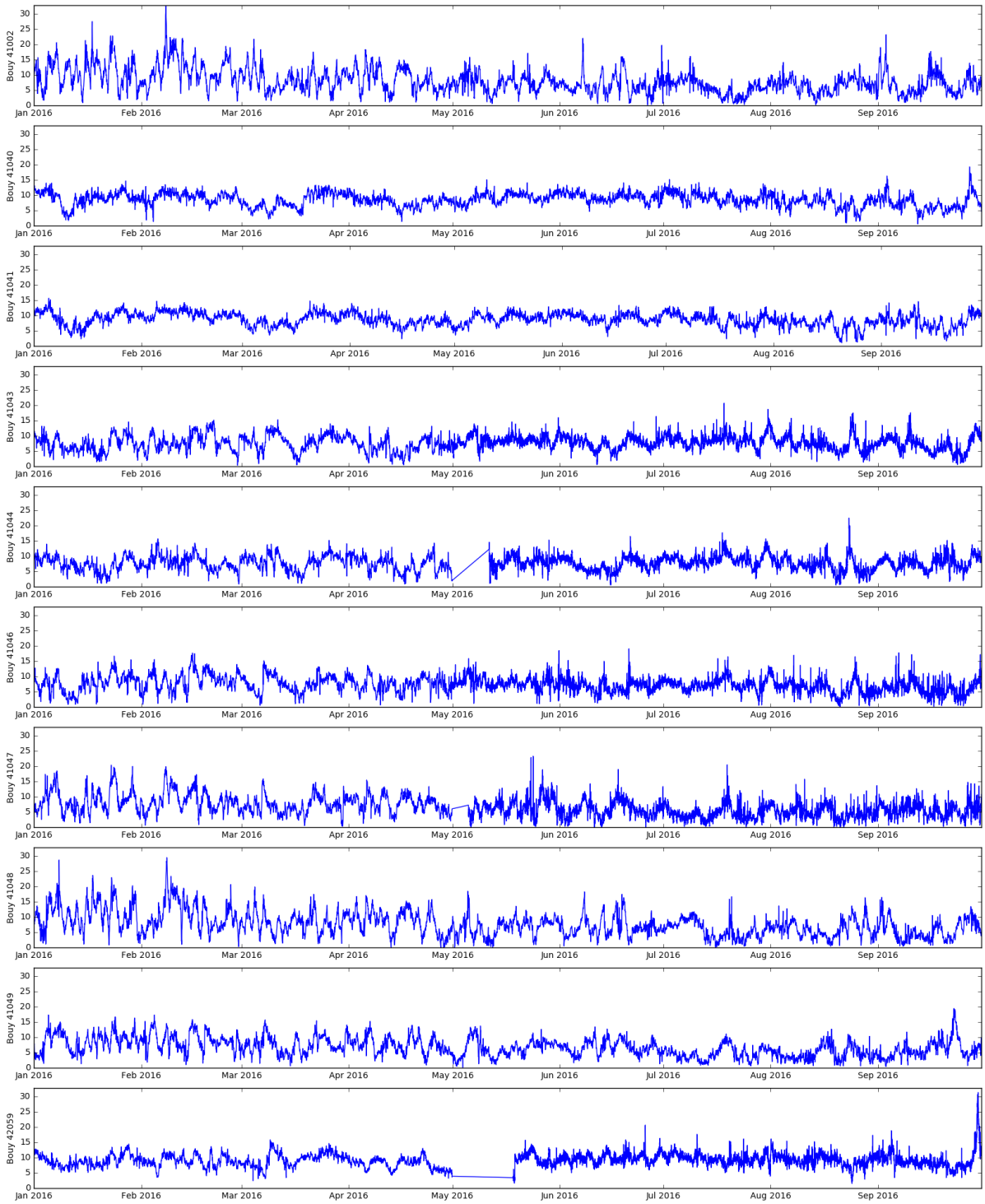
Wind Direction



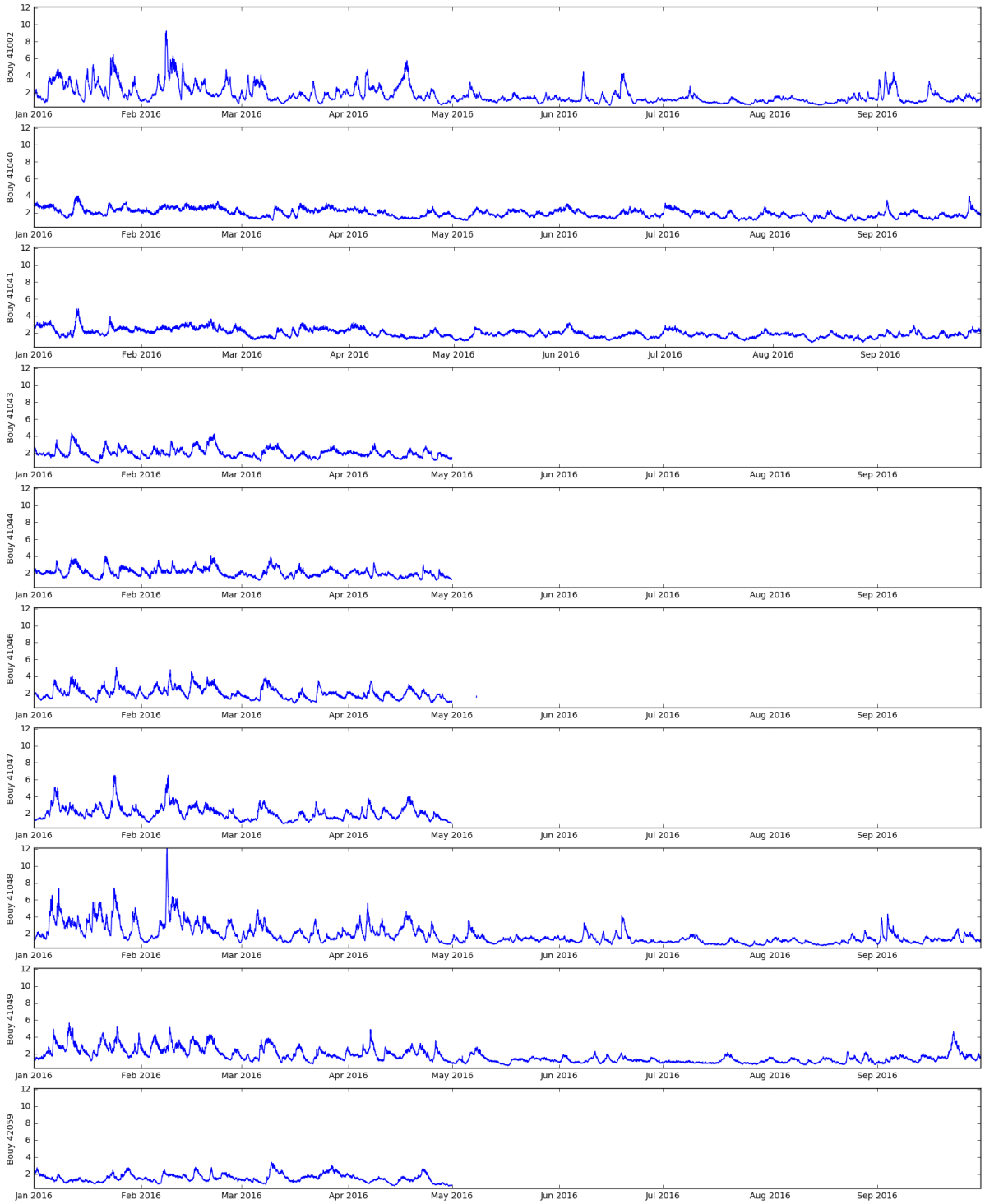
Wind Speed


Bouy 41002
Bouy 41040
Bouy 41041
Bouy 41043
Bouy 41044
Bouy 41046
Bouy 41047
Bouy 41048
Bouy 41049
Bouy 42059
0
5
10
15
20
25
Jan 2016
Feb 2016
Mar 2016
Apr 2016
May 2016
Jun 2016
Jul 2016
Aug 2016
Sep 2016

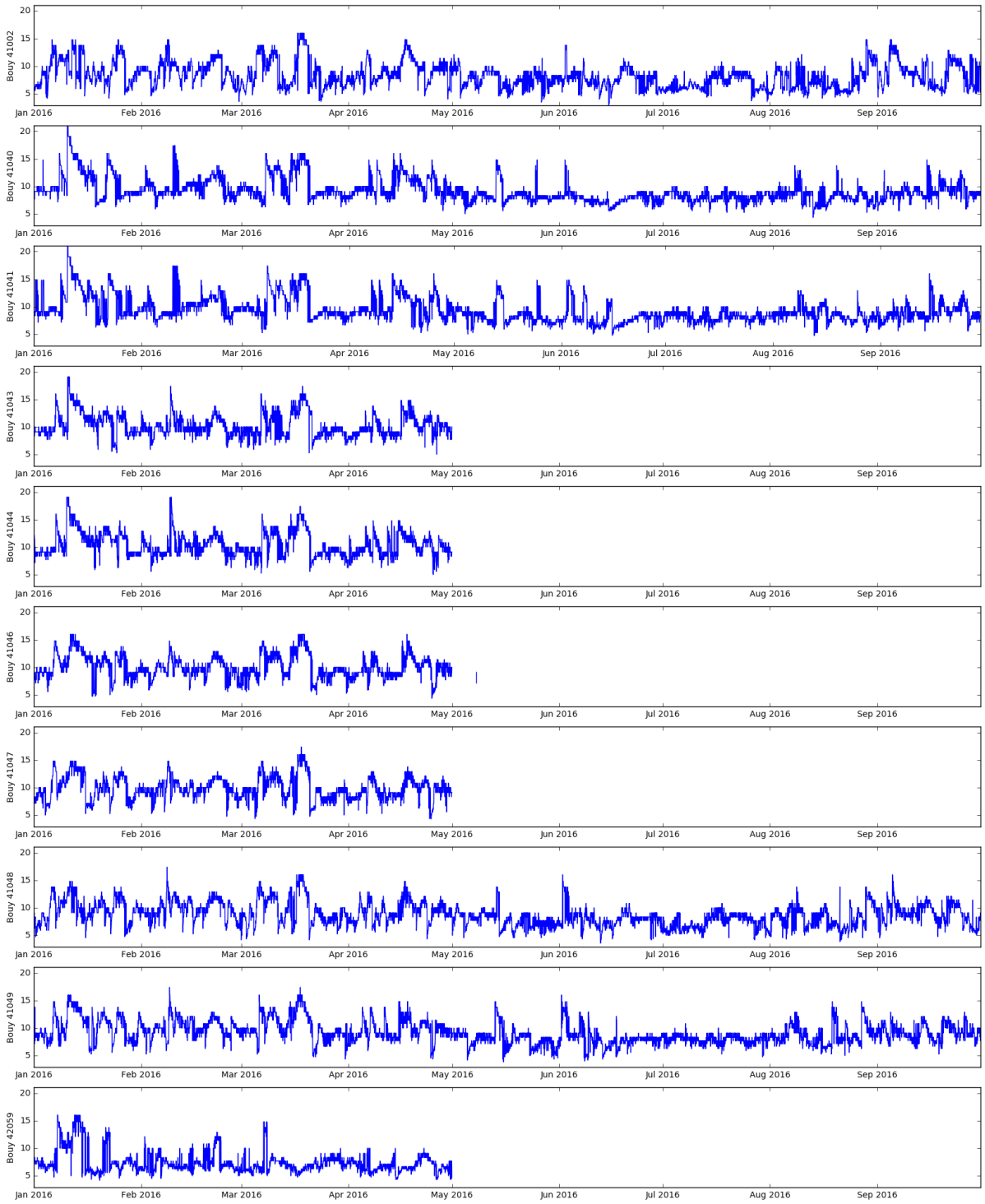
Gust Speed


Bouy 41002
Bouy 41040
Bouy 41041
Bouy 41043
Bouy 41044
Bouy 41046
Bouy 41047
Bouy 41048
Bouy 41049
Bouy 42059
0
5
10
15
20
25
30
Jan 2016
Feb 2016
Mar 2016
Apr 2016
May 2016
Jun 2016
Jul 2016
Aug 2016
Sep 2016

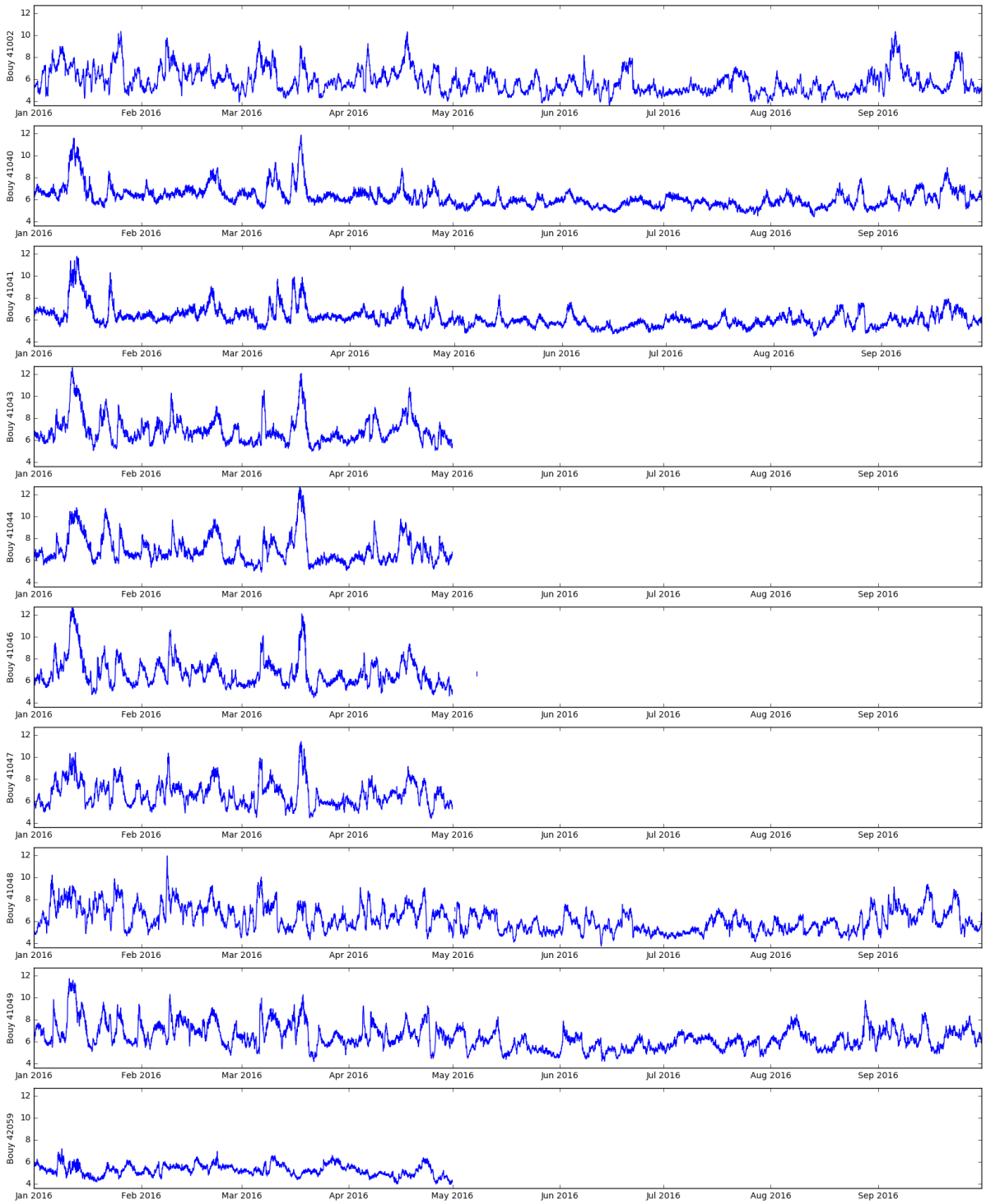
Wave Height



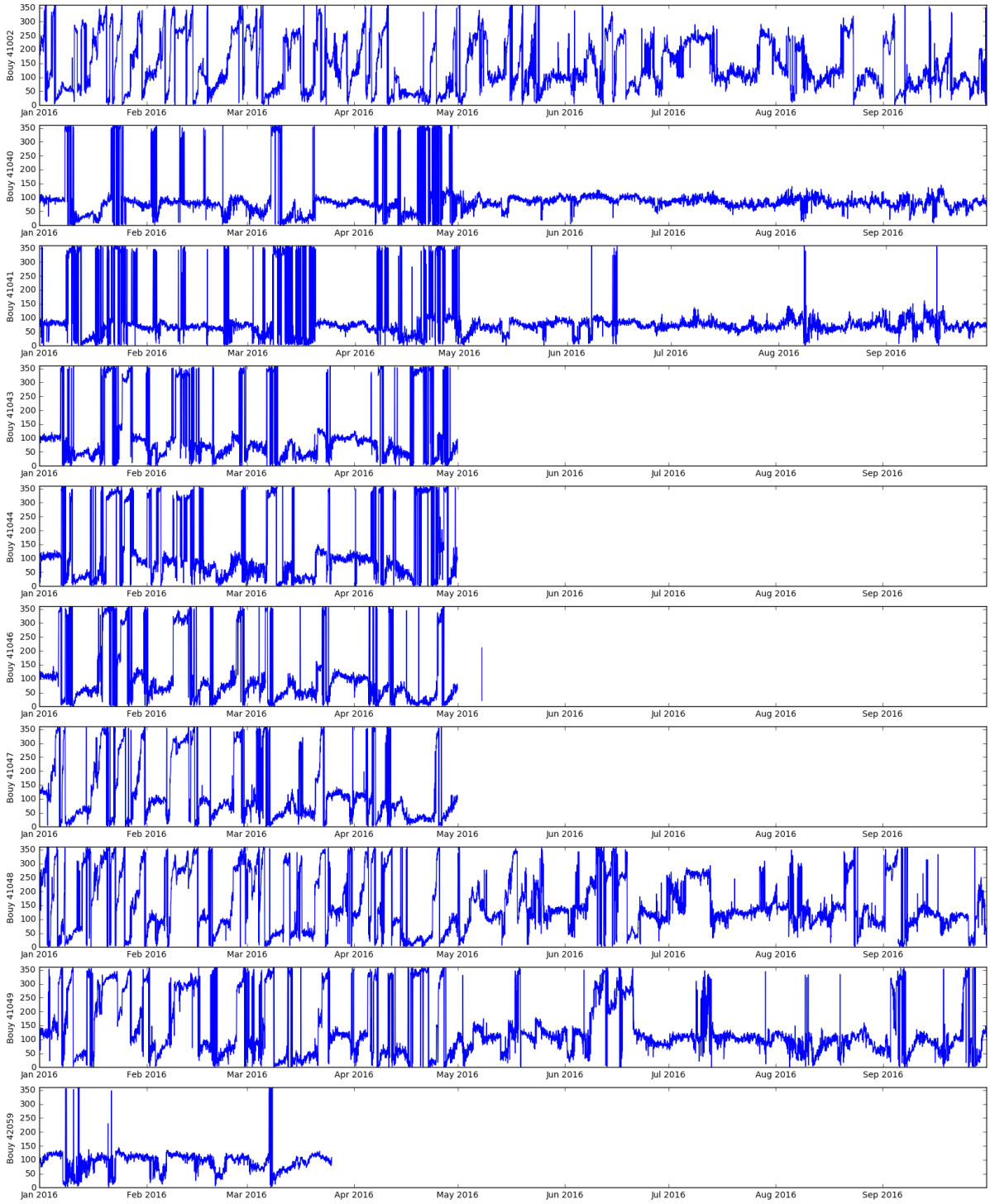
Dominant Wave Period



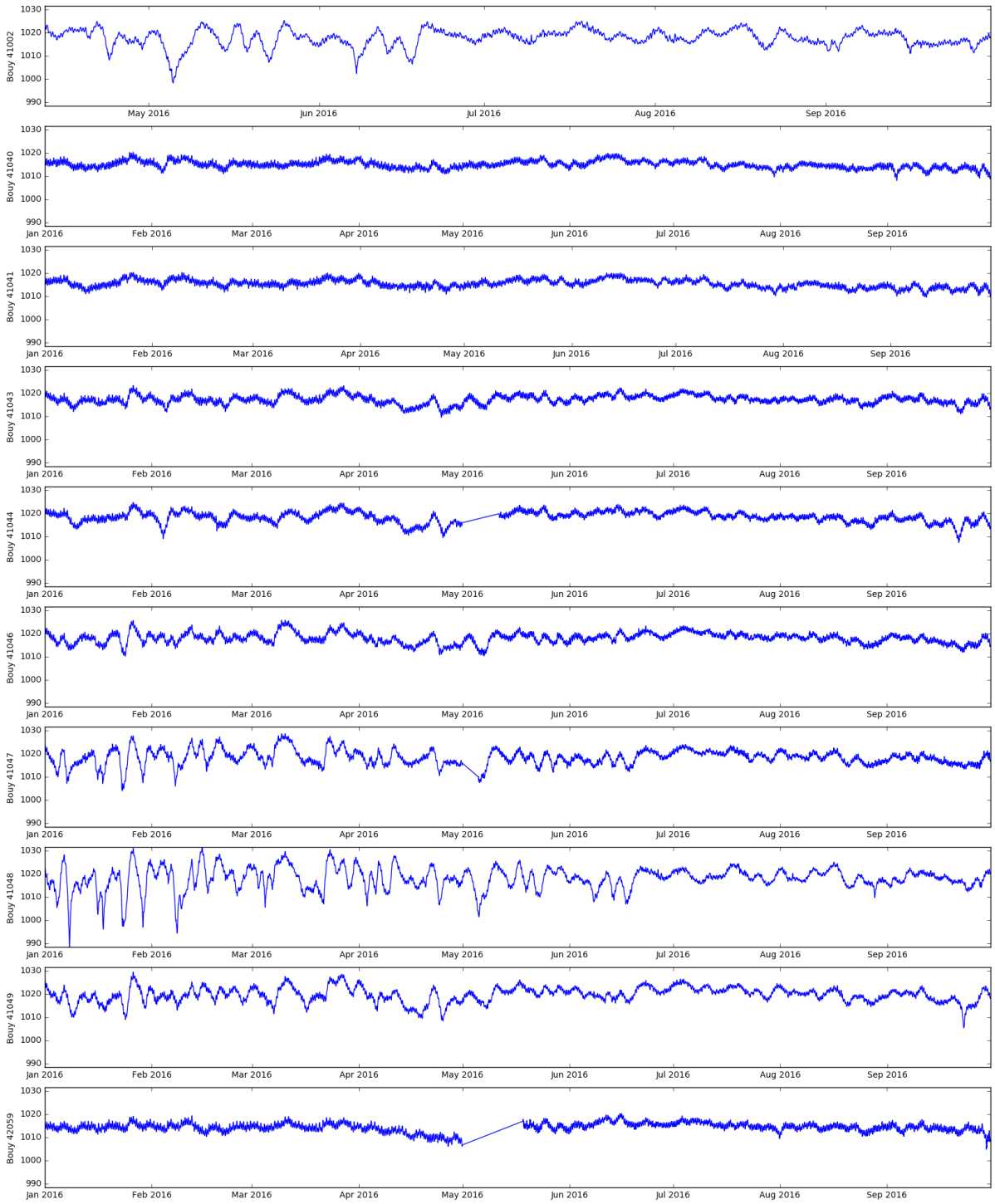
Average Wave Period



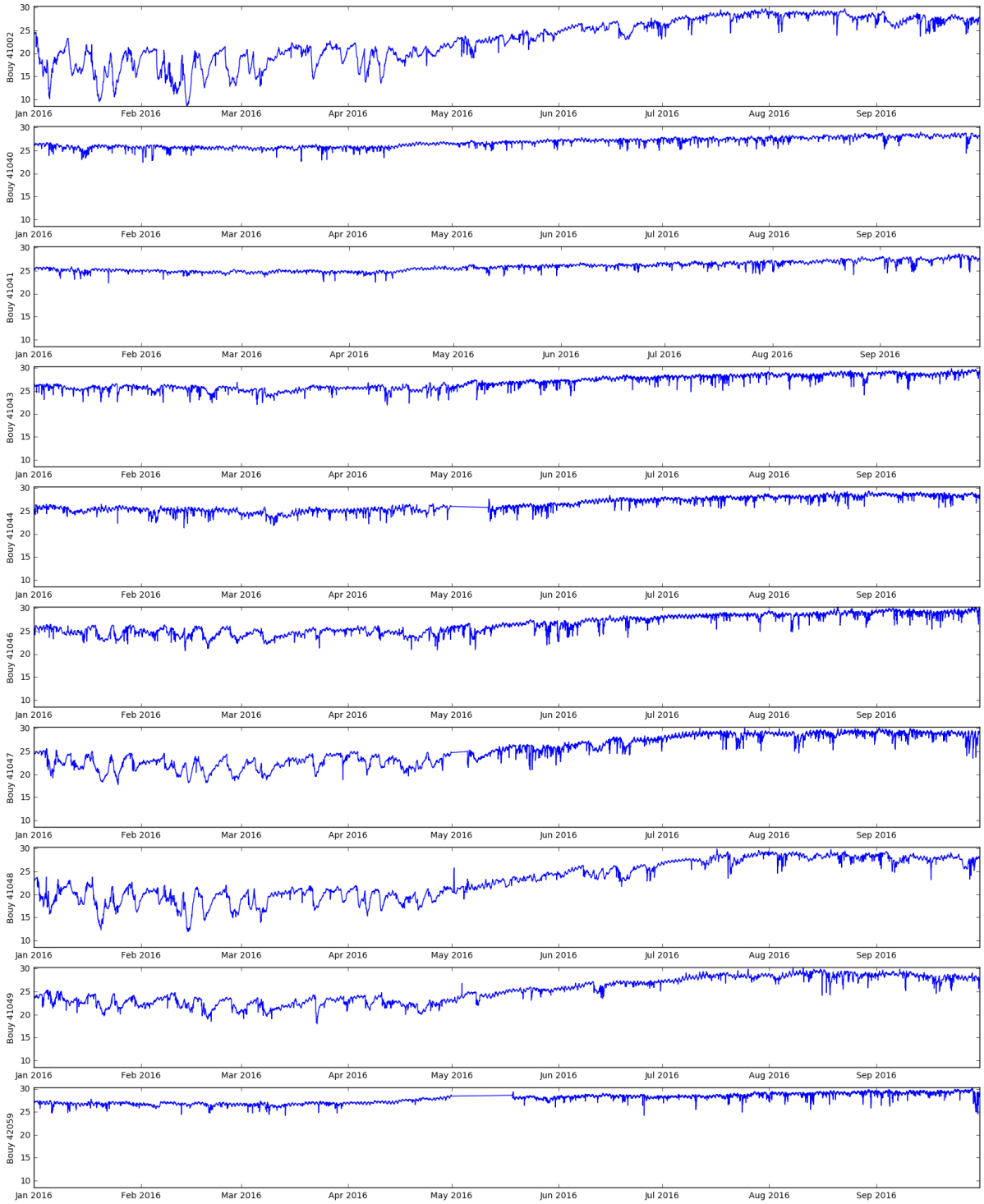
DPD Direction



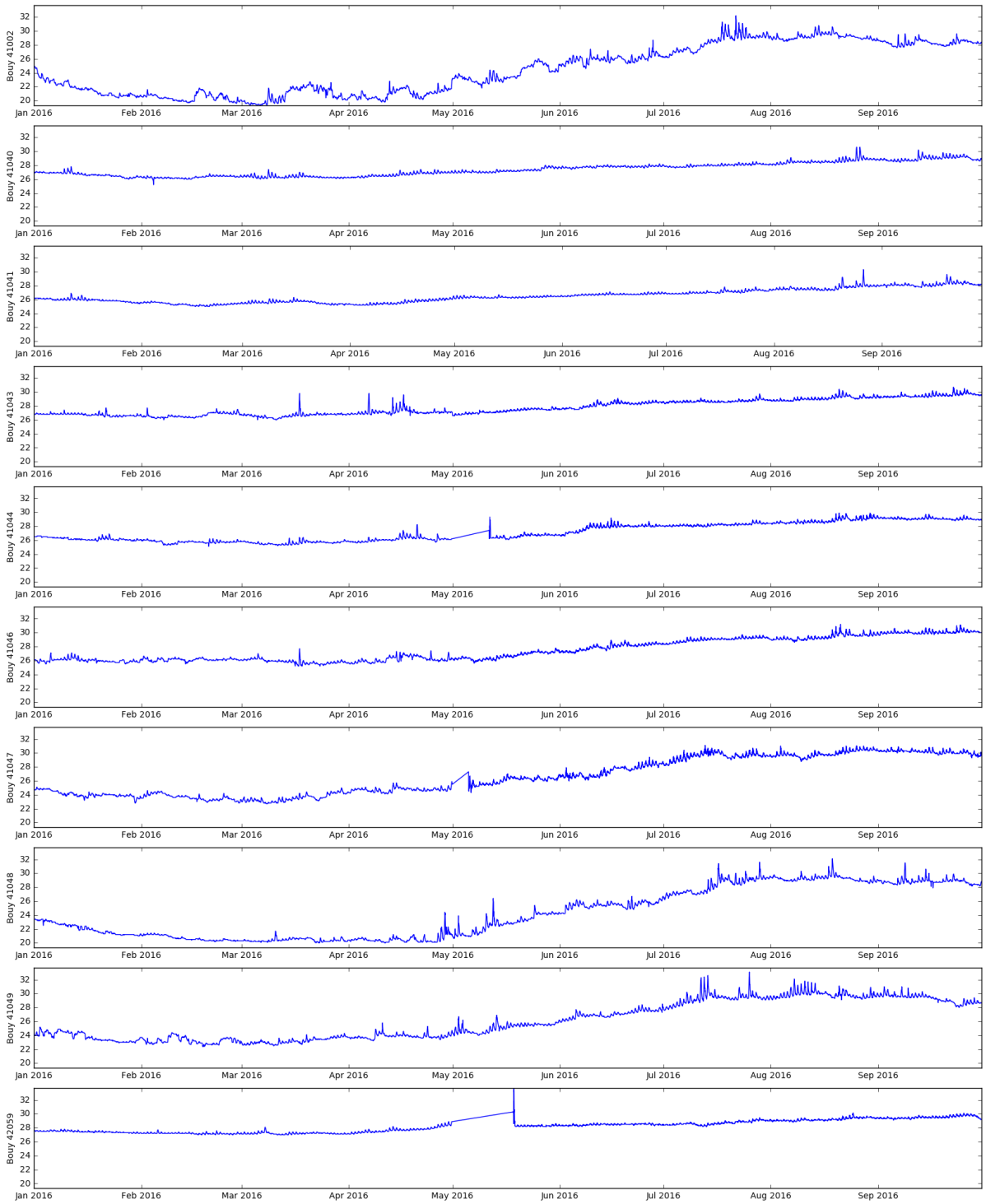
Sea Level Pressure



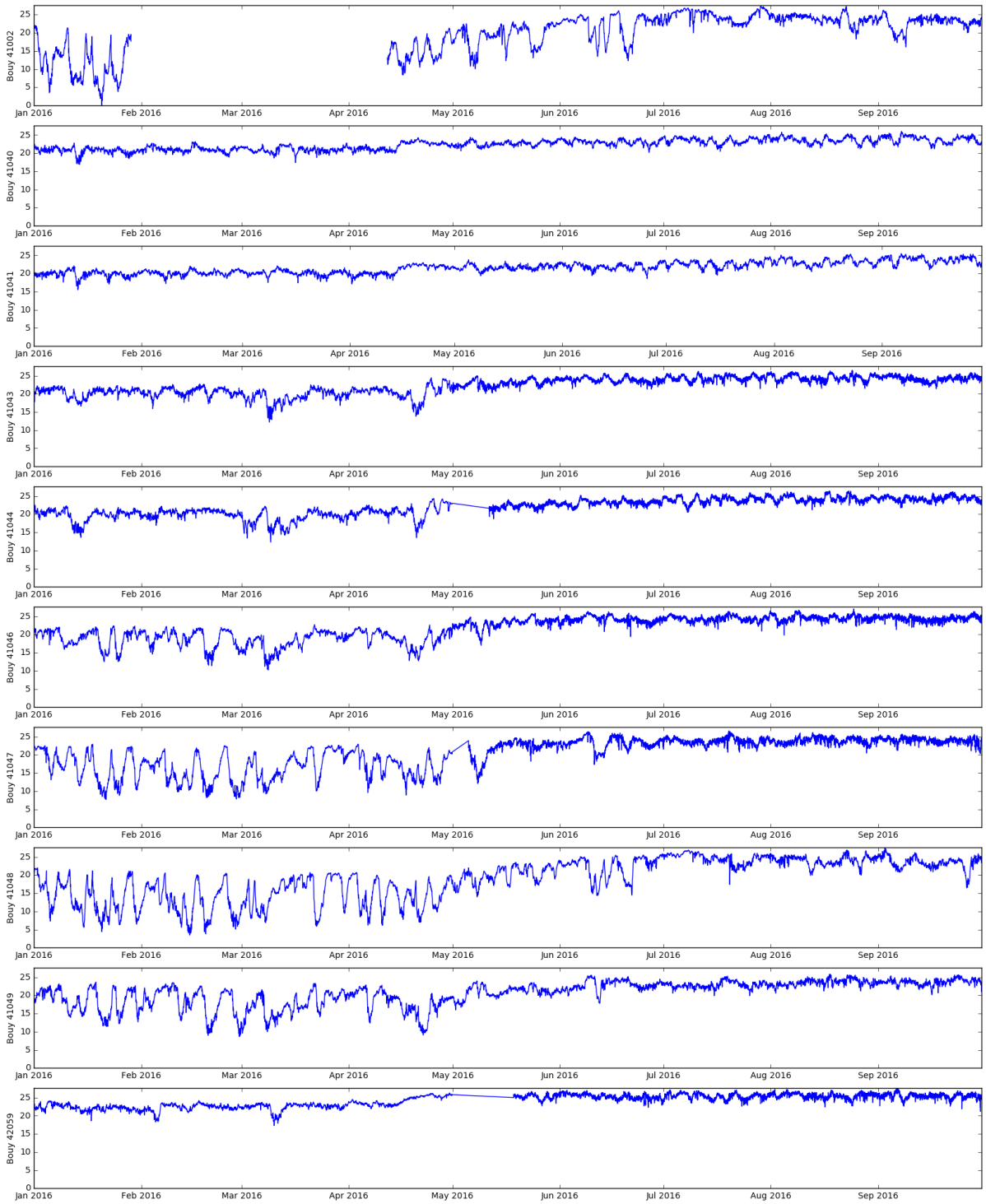
Air Temperature



Sea Surface Temperature



Dewpoint Temperature



The time series data tell us a few things. Some data, like Wind Direction, have high variability over short periods of time, while some sensors, like Sea Surface Temperature have more gradual changes overtime. It also appears that Bouys 41043, 41044, 41046, and 41047 had some sensors stop recording around May 2016 and Bouy 42059 also had some issues early in the year. There are also some straight lines in the plots, which would suggest those are actually data gaps, potentially the sensor failed at that time or could not transmit the data to a base station.

Define Functions For This Section

In [266]:
```
import numpy as np
import pandas as pd
import matplotlib
import matplotlib.pyplot as plt
from IPython.display import display, HTML
%matplotlib inline
```

In [7]:
```
# Months enums, haven't found python values
January = 1
February = 2
March = 3
April = 4
May = 5
June = 6
July = 7
August = 8
September = 9
```

In [199]:
```
def read_month(id, month, year):
    dateparse = lambda x: pd.datetime.strptime(x, '%Y %m %d %H %M')

    path = '../BuoyData/{}/{}{}{}.txt'.format(id, id, month, year)
    df = pd.read_csv(path,
                     delim_whitespace = True,
                     skiprows = [1],
                     parse_dates = {'DATETIME' : [0,1,2,3,4] },
                     date_parser=dateparse)
    df = clean_data(df)
    df['ID'] = id
    return df

def read_year(id, fromMonth, toMonth, year):

    bouy_data = []

    for month in range(fromMonth, toMonth + 1):
        bouy_data.append(read_month(id, month, year))

    return pd.concat(bouy_data)

# Removes missing values that are designated by a series of 9 values.
# Excerpt from NOAA Site: " Missing data in the Realtime files are denoted by
 "MM"
#                     while a variable number of 9's are used to denote missing
 data in the
#                     Historical files, depending on the data type (for exampl
e: 999.0 99.0)."
# Input:
#      df = dataframe to be cleaned
# Return:
#      df = cleaned dataframe
def clean_data(df):

    for column in df.columns[1:]:

        max_val = df[column].max()

        if(max_val % 9 == 0):
            df.loc[df[column] == max_val, column] = None

    return df
```

In [61]:
```
# data values
buyo_ids = [41002, 41040, 41041, 41043, 41044, 41046, 41047, 41048, 41049, 420
59]
startMonth = 1 # January
endMonth = 9 # September
totalMonths = 9
year = 2016
```

In [62]:
```
bs = { id : read_year(id, startMonth, endMonth, year) for id in buyo_ids }
```

In [200]:
```
# Concate all the data into one DataFrame
bouy_data = []

for id in buyo_ids:
    bouy_data.append(read_year(id, startMonth, endMonth, year))

bouy_data = pd.concat(bouy_data)
```

In [239]:
```
def get_name(predictor):
    values = {
     'WDIR' : 'Wind Direction',
     'WSPD' : 'Wind Speed',
     'GST' : 'Gust Speed',
     'WVHT' : 'Wave Height',
     'DPD' : 'Dominant Wave Period',
     'APD' : 'Average Wave Period',
     'MWD' : 'DPD Direction',
     'PRES' : 'Sea Level Pressure',
     'ATMP' : 'Air Temperature',
     'WTMP' : 'Sea Surface Temperature',
     'DEWP' : 'Dewpoint Temperature',
     'VIS' : 'Station Visibility',
     'TIDE' : 'Water Level'
    }
    return values[predictor]
```