Table of Contents and Notebook Setup

Contents

In [1]:
```
%%javascript
$.getScript('https://kmahelona.github.io/ipython_notebook_goodies/ipython_notebook_toc.js')
```

```
<IPython.core.display.Javascript object>
```

In [2]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import scipy as sp
from datetime import timedelta, datetime

pd.set_option('display.latex.repr', True)
pd.set_option('display.latex.longtable', True)
```

Importing Data

Importing Hourly Resolution Data

Obtaining Longitudes and Latitudes from All Stations

We start by reading the first two lines of the data file to obtain the longitude and latitude information of the stations. We zip them together into a list of tuples, and add an element to the beginning of this list corresponding to the time.

In [3]:
```
lon_and_lat = np.genfromtxt('AllStations_temperature_h_2017.dat', delimiter=" ", max_rows=2)
longitudes = lon_and_lat[0][~np.isnan(lon_and_lat[0])]
latitudes = lon_and_lat[1][~np.isnan(lon_and_lat[1])]

indices = list(zip(longitudes, latitudes))
indices.insert(0, "time")
```

Reading the Temperature Data

We now read in the temperature data, skipping the first two rows that contain the longitude and latitude information of the stations. The column names of the dataframe are the longitude and latitude tuples created above, and the index column is the timestamps.

In [4]:
```
df = pd.read_csv('AllStations_temperature_h_2017.dat', sep='\s+', skiprows=[0,1], names=indices, ind
df.head()
```

Out [4]:

	236.554	236.499	236.679	236.607	236.514	236.630	236.523	236.543	236.662	236.304	236.604	236
	48.5745	48.5376	48.4655	48.4608	48.4356	48.5273	48.4529	48.6804	48.4562	48.3891	48.4382	48.
time												
733408.0000	3.25	1.43	4.21	4.62	3.45	4.44	2.72	3.97	4.68	5.12	4.5	
733408.0417	3.06	1.38	4.28	4.77	3.46	4.21	2.64	4.15	4.61	4.95	4.5	
733408.0833	1.90	1.18	4.12	4.70	3.46	4.07	2.64	4.18	4.62	5.11	4.4	
733408.1250	1.69	0.98	4.20	4.77	3.65	3.67	2.81	4.27	4.71	5.33	4.5	
733408.1667	2.09	0.93	4.61	4.99	3.90	3.55	3.11	4.29	4.93	5.56	4.7	

Changing the Datetimes into Appropriate Timestamps

The Datetimes used in this file are MATLAB datetimes. We use a function to convert each time in the time column to the appropriate timestamp and then change the index column into timestamps.

In [5]:

```
def matlab_to_python_datetime(matlab_datenum):
    return datetime.fromordinal(int(matlab_datenum)) + timedelta(days=matlab_datenum%1) - timedelta(

df.index = [matlab_to_python_datetime(i) for i in df.index]
df.iloc[:,0:2].head()
```

Out [5]:

	236.554	236.499	
	48.5745	48.5376	
2008-01-01 00:00:00.000000	3.25	1.43	
2008-01-01 01:00:02.879995	3.06	1.38	
2008-01-01 01:59:57.120005	1.90	1.18	
2008-01-01 03:00:00.000000	1.69	0.98	
2008-01-01 04:00:02.879995	2.09	0.93	

Finding the Required Station

These are the coordinates where we wish to measure the temperature. We look for the station nearest to this point.

In [6]:

```
station_lon = 236.691
station_lat =  48.462
```

In [7]:

```
lon_diff = abs(longitudes - station_lon)
lat_diff = abs(latitudes - station_lat)

station_num = list((lon_diff+lat_diff)).index(min(lon_diff+lat_diff))
```

station_num provides the column number of the data we wish to observe.