In [1]:

```
from netCDF4 import Dataset
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.basemap import Basemap
```

In [2]:

```
data = Dataset("D:/Users/Owner/Documents/ArcGIS/Projects/Normals_Test/Spacetime/Normals.nc")

data
```

```
Out[2]: <class 'netCDF4._netCDF4.Dataset'>
        root group (NETCDF4 data model, file format HDF5):
            agg_shape_type: POLYGON
            geometry_unit: METER
            convert_factor: 1.0
            alignment: END_TIME
            reference_time:
            time_size: 1
            time_unit: MONTHS
            time_step_label: 1 Month
            first_start_time: 2009-12-01 06:00:00
            first_end_time: 2010-01-01 06:00:00
            last_start_time: 2010-12-01 06:00:00
            last_end_time: 2011-01-01 06:00:00
            data_min_time: 2010-01-01 06:00:00
            data_max_time: 2011-01-01 06:00:00
            start_bias: 100.0
            end_bias: 0.0
            description: Space-Time Pattern Mining Panel Cube
            history: Created by Sun Mar  7 15:09:31 2021
            source: Space Time Pattern Mining Tools;2.7.1
            feature_type: timeSeries
            extent: [-13894065.66105461   2708362.4448694   -7451448.34216333
           5478284.68523675]
            esri_pe_string: PROJCS['WGS_1984_World_Equidistant_Cylindrical',G
        EOGCS['GCS_WGS_1984',DATUM['D_WGS_1984',SPHEROID['WGS_1984',6378137.
        0,298.257223563]],PRIMEM['Greenwich',0.0],UNIT['Degree',0.01745329251
        99433]],PROJECTION['Equidistant_Cylindrical_Ellipsoidal'],PARAMETER['
        False_Easting',0.0],PARAMETER['False_Northing',0.0],PARAMETER['Centra
        l_Meridian',0.0],PARAMETER['Standard_Parallel_1',0.0],UNIT['Meter',1.
        0]];-20037700 -10002100 10000;-100000 10000;-100000 10000;0.001;0.00
        1;0.001;IsHighPrecision
            projection_authority_code: 4087
            raw_pe_string: PROJCS['WGS_1984_World_Equidistant_Cylindrical',GE
        OGCS['GCS_WGS_1984',DATUM['D_WGS_1984',SPHEROID['WGS_1984',6378137.0,
        298.257223563]],PRIMEM['Greenwich',0.0],UNIT['Degree',0.0174532925199
        433]],PROJECTION['Equidistant_Cylindrical_Ellipsoidal'],PARAMETER['Fa
        lse_Easting',0.0],PARAMETER['False_Northing',0.0],PARAMETER['Central_
        Meridian',0.0],PARAMETER['Standard_Parallel_1',0.0],UNIT['Meter',1.
        0]];-20037700 -10002100 10000;-100000 10000;-100000 10000;0.001;0.00
        1;0.001;IsHighPrecision
            location_id_field: Cell
            rate_info:
            dimensions(sizes): time(13), locations(481631)
            variables(dimensions): int32 projection(), float64 time(time), fl
        oat64 x(locations), float64 y(locations), float64 lat(locations), flo
        at64 lon(locations), int32 poly_breaks(locations), float64 poly_coord
        s(locations), int32 time_step_ID(time,locations), int32 location_ID(t
        ime,locations), int32 Cell(time,locations), float64 PRECIP_NONE_ZEROS
        (time,locations), float64 PRECIP_NONE_ZEROS_TREND_ZSCORE(locations),
        float64 PRECIP_NONE_ZEROS_TREND_PVALUE(locations), float64 PRECIP_NON
        E_ZEROS_TREND_BIN(locations)
            groups:
```

In [3]:

```
lats = data.variables['lat'][:]
lons = data.variables['lon'][:]
time = data.variables['time'][:]
precip = data.variables['PRECIP_NONE_ZEROS'][:]
```

In [4]:

```
mp = Basemap(projection = 'cyl',
            llcrnrlon = -125,
            llcrnrlat = 23,
            urcrnrlon = -65,
            urcrnrlat = 50,
            resolution = "i")
```

In [5]:

```
lon, lat = np.meshgrid(lons, lats, sparse = True)
x, y = mp(lon, lat)
```

In [6]:

```
c_scheme = mp.pcolor(x, y, np.squeeze(precip[0,:,:]), cmap = 'jet')
```

```
---------------------------------------------------------------------
------
IndexError                                Traceback (most recent call
last)
<ipython-input-6-c90c28afc744> in <module>
----> 1 c_scheme = mp.pcolor(x, y, np.squeeze(precip[0,:,:]), cmap =
'jet')
      2

D:\Users\Owner\Anaconda3\lib\site-packages\numpy\ma\core.py in __geti
tem__(self, indx)
   3172             # mask of being reshaped if it hasn't been set up pro
perly yet
   3173             # So it's easier to stick to the current version
-> 3174             dout = self.data[indx]
   3175             _mask = self._mask
   3176

IndexError: too many indices for array
```

In [ ]:

In [ ]: