蒙芳秀，12331305

1. Niño 3.4 index

The *Niño 3.4 anomalies* may be thought of as representing the average equatorial sea surface temperatures (SSTs) across the Pacific from about the dateline to the South American coast (`5N-5S` , `170W-120W`). The Niño 3.4 index typically uses a 3-month running mean, and El Niño or La Niña events are defined when the Niño 3.4 SSTs exceed `+/- 0.5° C` for a period of `5` months or more. Check Equatorial Pacific Sea Surface Temperatures for more about the Niño 3.4 index.

In this problem set, you will use the sea surface temperature (SST) data from NOAA. Download the `netCDF4` file (`NOAA_NCDC_ERSST_v3b_SST.nc`) here.

1.1 [10 points] Compute monthly climatology for SST from Niño 3.4 region, and subtract climatology from SST time series to obtain anomalies.

思路：首先读取 nc 数据，然后取出(5N-5S,170W-120W)的 SST，按照经纬度取平均，得到 nino3.4 区域的月 SST，做以上两者的差值，得到 SST 异常值。

In [37]:

```
# Load the netCDF4 file
ds = xr.open_dataset("NOAA_NCDC_ERSST_v3b_SST.nc", engine="netcdf4")
# Check the data
ds
# Using the slice method to find a nearby value  SST from Niño 3.4 region
sst_nino=ds.sst.sel(lat=slice(-5, 5),lon=slice(10, 60))
sst_nino_mean=sst_nino.mean(dim=['lat', 'lon']);
# subtract climatology from SST time series to obtain anomalies
sst_anom=sst_nino-sst_nino_mean
sst_anom
```

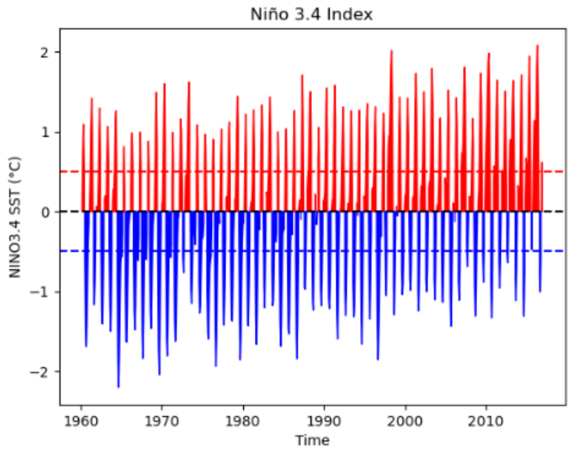
Out[37]: xarray.DataArray 'sst' (**time**: 684, **lat**: 5, **lon**: 26)

```
array([[[-0.23072624, -0.27137756,         nan, ...,  0.77036476,
          0.69953156,  0.683218  ],
        [-0.00307465,         nan,         nan, ...,  0.492342  ,
          0.5234051 ,  0.57826614],
        [ 0.54745674,         nan,         nan, ...,  0.06572342,
          0.22882843,  0.39834976],
        [ 1.0070438 ,         nan,         nan, ..., -0.2455616 ,
          0.0116806 ,  0.26471138],
        [ 1.1845741 ,         nan,         nan, ..., -0.3318653 ,
         -0.07986641,  0.18948174]],
```

1.2 [10 points] Visualize the computed Niño 3.4. Your plot should look similar to this one

思路：我从 CHATGPT 中得到灵感，首先计算(5N-5S,170W-120W)的 SST，使用 numpy.convolve()函数对其进行 3 个月滑动平均，并对所有数据减去均值，得到 Niño 3.4. 异常值，并画图进行可视化。

In [67]:
```python
# 将NINO3.4指数进行3个月滑动平均
window_size = 3
nino34 = np.convolve(sst_nino_mean, np.ones(window_size)/window_size, mode='valid')
nino34_anorm=nino34-np.mean(nino34)
nino34_anorm

times = ds['time']
# 创建一个新的布尔数组来标记正值和负值
positive_values = np.array(nino34_anorm) >= 0

# 创建一个具有相同长度的数组，其中正值为1，负值为-1
color_fill = np.where(positive_values, 1, -1)

# 绘制面积图
plt.fill_between(times[window_size-1:], nino34_anorm, where=color_fill>0, color='red')
plt.fill_between(times[window_size-1:], nino34_anorm, where=color_fill<0, color='blue')

# 添加水平线作为参考
plt.axhline(y=0.5, color='red', linestyle='--')
plt.axhline(y=0, color='black', linestyle='--')
plt.axhline(y=-0.5, color='blue', linestyle='--')
# 设置图表标题和轴标签
plt.title('Niño 3.4 Index')
plt.xlabel('Time')
plt.ylabel('NINO3.4 SST (° C)')

# 显示图表
plt.show()
```



2. Earth's energy budget

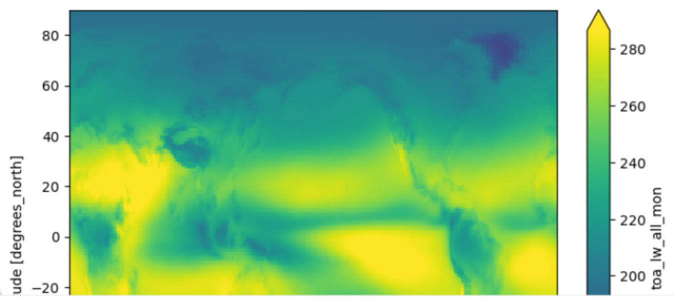
In this problem set, you will analyze top-of-atmosphere (TOA) radiation data from NASA's CERES project. Read this post for more about Earth's energy budget.

2.1 [5 points] Make a 2D plot of the time-mean TOA longwave, shortwave, and solar radiation for all-sky conditions. Add up the three variables above and verify (visually) that they are equivalent to the TOA net flux.

思路：首先读取 nc 文件，选择所需变量求均值，并画出各自的 2D map 图

In [90]:
```python
#提取数据
toa_lw_all_mon_mean= data.toa_lw_all_mon.mean(dim=['time'])
toa_sw_all_mon_mean=data.toa_sw_all_mon.mean(dim=['time'])
toa_solar_mon_mean= data.solar_mon.mean(dim=['time'])
toa_all_mon=toa_sw_all_mean+toa_lw_all_mon_mean+toa_solar_mon_mean
toa_lw_all_mon_mean.plot(robust=True, figsize=(8, 6))
toa_sw_all_mon_mean.plot(robust=True, figsize=(8, 6))
toa_solar_mon_mean.plot(robust=True, figsize=(8, 6))
toa_all_mon.plot(robust=True, figsize=(8, 6))
data.toa_net_all_mon.mean(dim=['time']).plot(robust=True, figsize=(8, 6))
```

Out[90]:
```
<matplotlib.collections.QuadMesh at 0x22c47315910>
```



2.2 [10 points] Calculate and verify that the TOA incoming solar, outgoing longwave, and outgoing shortwave approximately match up with the cartoon above.

[Hint: Consider calculating the area of each grid]

思路：这题不是很理解，分别读取了 incoming solar，outgoing longwave, and outgoing shortwave，并按照三个纬度进行计算

In [100]:
```python
#the TOA incoming solar
in_solar=data.solar_mon.mean(dim=['time','lon','lat'])
in_solar
# # # outgoing longwave
out_lw=data.toa_lw_all_mon.mean(dim=['time','lon','lat'])
# #outgoing shortwave
out_sw=data.toa_sw_all_mon.mean(dim=['time','lon','lat'])
out_all=out_lw+out_sw
out_all
```

Out[100]:

xarray.DataArray

```
array(327.05957, dtype=float32)
```

▶ Coordinates: (0)

▶ Indexes: (0)

▶ Attributes: (0)

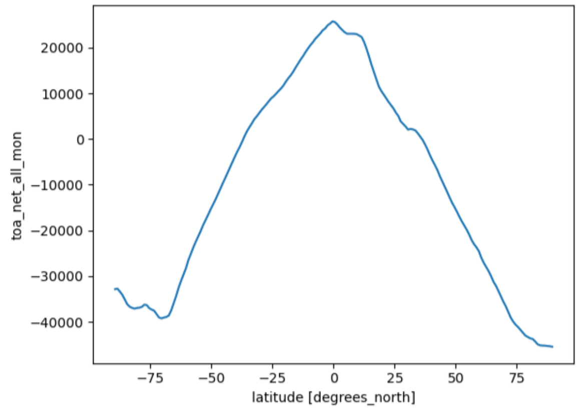
2.3 [5 points] Calculate and plot the total amount of net radiation in each 1-degree latitude band. Label with correct units.

思路：选取变量，按照时间求均值，再按照经度求和，然后画图

2.3 [5 points] Calculate and plot the total amount of net radiation in each 1-degree latitude band. Label with correct units.

```
In  [23]: m_net=data.toa_net_all_mon.mean(dim=['time'])
          m_net.sum(dim=['lon']).plot()
```

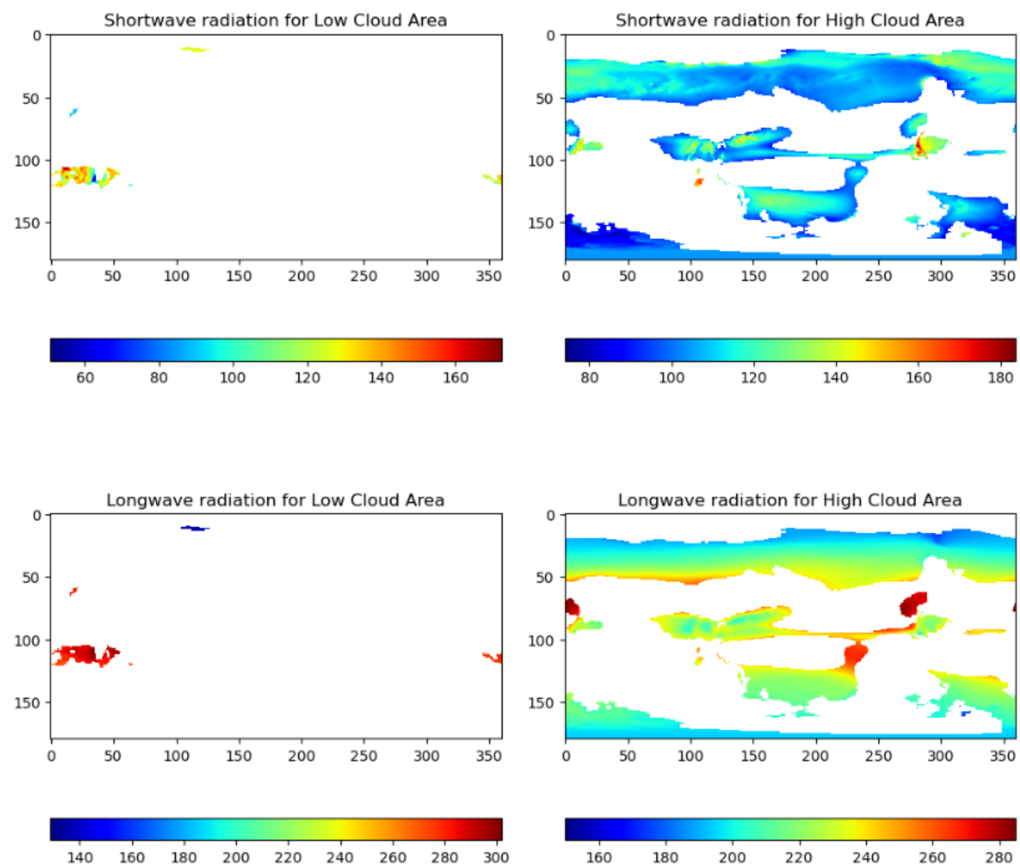
```
Out[23]: [<matplotlib.lines.Line2D at 0x206a6736390>]
```



2.4 [5 points] Calculate and plot composites of time-mean outgoing shortwave and longwave radiation for low and high cloud area regions. Here we define low cloud area as ≤25% and high cloud area as ≥75%. Your results should be 2D maps.

思路：我从 CHATGPT 中得到灵感，首先对选取输出长波、短波、云面积三个变量，寻找云面积中云面积小于 25 的位置为低云区，云面积中云面积大于 75 的位置定义为高云区。画出长波、短波中对应高云区，低云区的合成图。

```
In  [34]: #读取数据
          cloud_area_mean=data.cldarea_total_daynight_mon.mean(dim=['time'])
          all_sw_mean=data.toa_sw_all_mon.mean(dim=['time'])
          all_lw_mean=data.toa_lw_all_mon.mean(dim=['time'])
          #记录low cloud area as ≤25% and high cloud area as ≥75%的位置
          low_cloud_mask = cloud_area_mean <= 25
          high_cloud_mask = cloud_area_mean >= 75
          #寻找短波\长波辐射中低云区和高云区，通过将非云区域的元素遮罩（mask）掉
          swall_low_cloud_composite = np.ma.array(all_sw_mean, mask=~low_cloud_mask)
          swall_high_cloud_composite =np.ma.array(all_sw_mean, mask=~high_cloud_mask)
          lwall_low_cloud_composite= np.ma.array(all_lw_mean, mask=~low_cloud_mask)
          lwall_high_cloud_composite= np.ma.array(all_lw_mean,mask=~high_cloud_mask)
          #画图
          plt.figure(figsize=(10, 10))
          plt.subplot(221)
          plt.imshow(swall_low_cloud_composite, cmap='jet')
          plt.title('Shortwave radiation for Low Cloud Area')
          plt.colorbar(orientation='horizontal')
          plt.subplot(222)
          plt.imshow(swall_high_cloud_composite, cmap='jet')
          plt.title('Shortwave radiation for High Cloud Area')
          plt.colorbar(orientation='horizontal')
          plt.subplot(223)
          plt.imshow(lwall_low_cloud_composite, cmap='jet')
          plt.title('Longwave radiation for Low Cloud Area')
          plt.colorbar(orientation='horizontal')
          plt.subplot(224)
          plt.imshow(lwall_high_cloud_composite, cmap='jet')
          plt.title('Longwave radiation for High Cloud Area')
          plt.colorbar(orientation='horizontal')
          plt.tight_layout()
          plt.show()
```



2.5 [5 points] Calculate the global mean values of shortwave and longwave radiation, composited in high and low cloud regions. What is the overall effect of clouds on shortwave and longwave radiation?

思路：首先对选取输出长波、短波、云面积三个变量，寻找云面积中云面积小于 25 的位置为低云区，云面积中云面积大于 75 的位置定义为高云区。计算长波、短波中对应高云区，低云区的区域均值。也可类似第四题进行可视化。

2.5 [5 points] Calculate the global mean values of shortwave and longwave radiation, composited in high and low cloud regions. What is the overall effect of clouds on shortwave and longwave radiation?

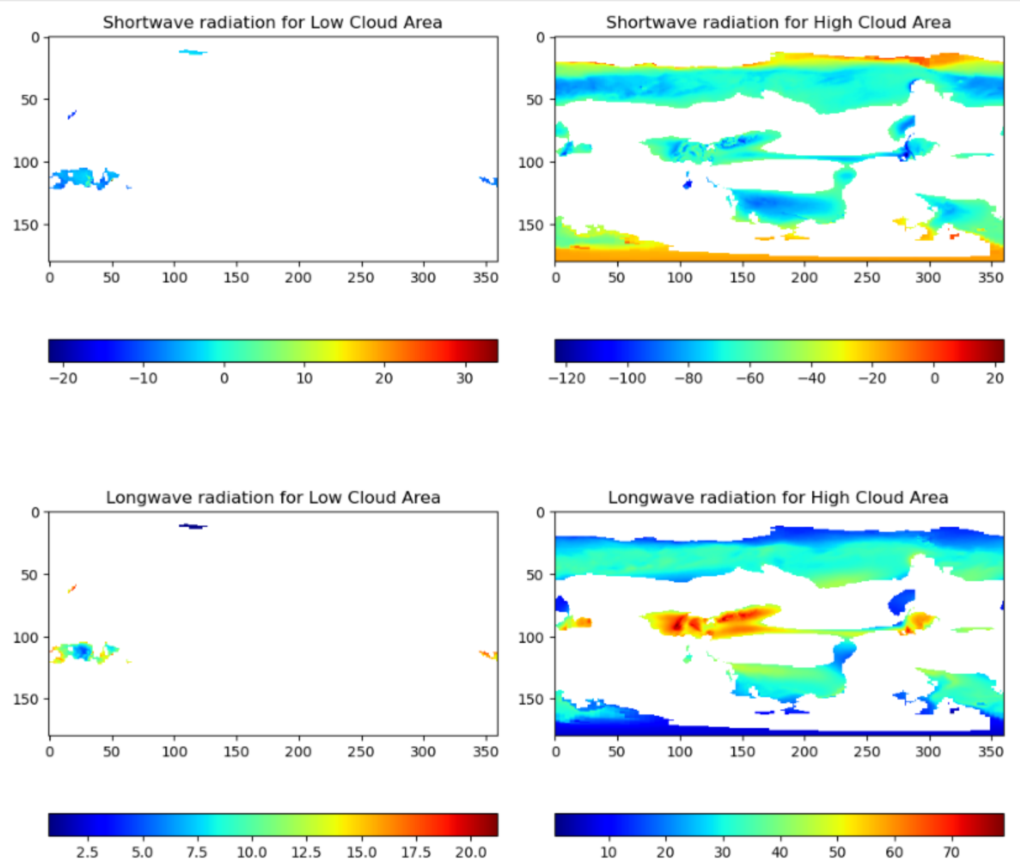
In [30]:
```
cloud_area_mean=data.cldarea_total_daynight_mon.mean(dim=['time'])
cre_sw_mean=data.toa_cre_sw_mon.mean(dim=['time'])
cre_lw_mean=data.toa_cre_lw_mon.mean(dim=['time'])
```

In [31]:
```
#记录low cloud area as ≤25% and high cloud area as ≥75%的位置
low_cloud_mask = cloud_area_mean <= 25
high_cloud_mask = cloud_area_mean >= 75
```

In [32]:
```
#寻找短波|长波辐射中低云区和高云区，通过将非云区域的元素遮罩（mask）掉,并进行区域平均
swcre_low_cloud_composite = np.mean(np.ma.array(cre_sw_mean, mask=~low_cloud_mask))
scre_high_cloud_composite = np.mean(np.ma.array(cre_sw_mean, mask=~high_cloud_mask))
lwcre_low_cloud_composite= np.mean(np.ma.array(cre_lw_mean, mask=~low_cloud_mask))
lwcre_high_cloud_composite= np.mean(np.ma.array(cre_lw_mean,mask=~high_cloud_mask))
lwcre_high_cloud_composite
```

Out[32]: 29.668634441384736

In [33]:
```python
#寻找短波\长波辐射中低云区和高云区，通过将非云区域的元素遮罩（mask）掉，我们可以得到只包含高云和低云区域的合成图像.
sw_low_cloud_composite = np.ma.array(cre_sw_mean, mask=~low_cloud_mask)
sw_high_cloud_composite = np.ma.array(cre_sw_mean, mask=~high_cloud_mask)
lw_low_cloud_composite= np.ma.array(cre_lw_mean, mask=~low_cloud_mask)
lw_high_cloud_composite= np.ma.array(cre_lw_mean,mask=~high_cloud_mask)

plt.figure(figsize=(10, 10))
plt.subplot(221)
plt.imshow(sw_low_cloud_composite, cmap='jet')
plt.title('Shortwave radiation for Low Cloud Area')
plt.colorbar(orientation='horizontal')
plt.subplot(222)
plt.imshow(sw_high_cloud_composite, cmap='jet')
plt.title('Shortwave radiation for High Cloud Area')
plt.colorbar(orientation='horizontal')
plt.subplot(223)
plt.imshow(lw_low_cloud_composite, cmap='jet')
plt.title('Longwave radiation for Low Cloud Area')
plt.colorbar(orientation='horizontal')
plt.subplot(224)
plt.imshow(lw_high_cloud_composite, cmap='jet')
plt.title('Longwave radiation for High Cloud Area')
plt.colorbar(orientation='horizontal')
plt.tight_layout()
plt.show()
```



可以看出，长波、短波辐射都主要集中在高云区，说明高云区的长波、短波辐射的影响较大。

3. Explore a `netCDF` dataset

Browse the NASA's Goddard Earth Sciences Data and Information Services Center (GES DISC) website. Search and download a dataset you are interested in. You are also welcome to use data from your group in this problem set. But the dataset should be in `netCDF` format, and have temporal information.

3.1 [5 points] Plot a time series of a certain variable with monthly seasonal cycle removed.

思路：选取东北地区 500hPa 5 月逐日位势高度场计算逐日东北冷涡指数，将东北冷涡指数定义为该区域 500hPa 位势高度场的区域平均值。

In [73]:
```
#选择5月,划定区域（120-130E,40-50N）,500hPa位势高度场
hgt500Data=hgt_data['hgt']#.sel(lat=slice(40, 50))#,lon=slice(300,310),level=500) #,time=slice("2017-05-01T00:00:00.000000000", "2017-05-31T
# 选择5月份的数据
hgt_may = hgt500Data.sel(time=hgt500Data['time.month']== 5)
hgt_box = hgt_may.sel(lon=slice(120, 130)).isel(lat=slice(16,21))
#计算东北冷涡指数
col_index=hgt_box.isel(level=5).mean(dim=['lat','lon'])
col_index
```

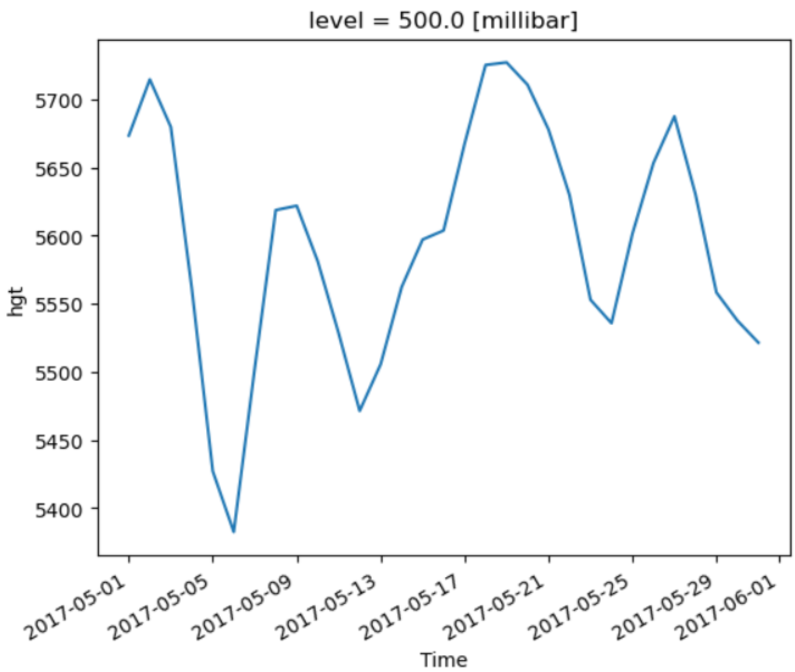
Out[73]:
```
xarray.DataArray 'hgt' (time: 31)

array([5673.49, 5714.63, 5679.78, 5561.27, 5427.2 , 5382.64, 5502.76,
       5618.77, 5621.95, 5581.25, 5528.31, 5471.44, 5505.69, 5562.29,
       5597.18, 5603.8 , 5667.44, 5725.24, 5727.13, 5710.71, 5677.85,
       5629.77, 5552.97, 5535.8 , 5601.53, 5653.29, 5687.55, 5630.54,
       5558.36, 5537.69, 5521.53], dtype=float32)

▼Coordinates:
level   ()      float32         500.0
time    (time)  datetime64[ns]  2017-05-01 ... 2017-05-31
►Indexes: (1)
►Attributes: (0)
```

3.2 [5 points] Make at least 5 different plots using the dataset.

1、画东北冷涡的时间序列图

In [70]:
```
#画东北冷涡指数的时间序列图
col_index.plot()
```

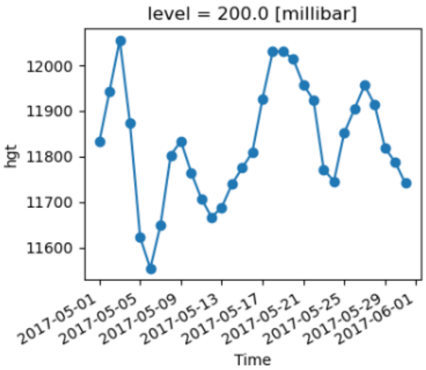
Out[70]: [<matplotlib.lines.Line2D at 0x206b6f5d910>]

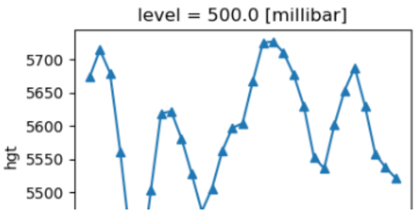


2、比较 200、500、850hPa 位势高度场时间序列图

In [87]:
```
#画关键区域的不同高层的 时间序列
hgt_box.sel(level=200).mean(dim=['lat','lon']).plot(marker="o", size=3)
hgt_box.sel(level=500).mean(dim=['lat','lon']).plot(marker="^", size=3)
hgt_box.sel(level=850).mean(dim=['lat','lon']).plot(marker="+", size=3)
```

Out[87]: [<matplotlib.lines.Line2D at 0x206c7574710>]

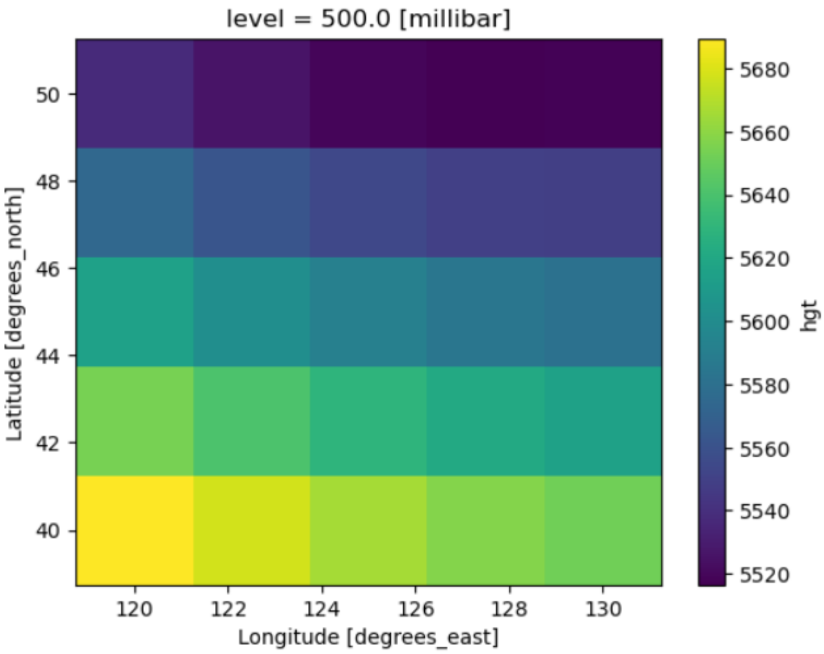




3、画 500hPa 位势高度场空间分布图

In [75]:
```
#画关键区域的500hpa 2Dmap
hgt_box.isel(level=5).mean(dim=['time']).plot()
```

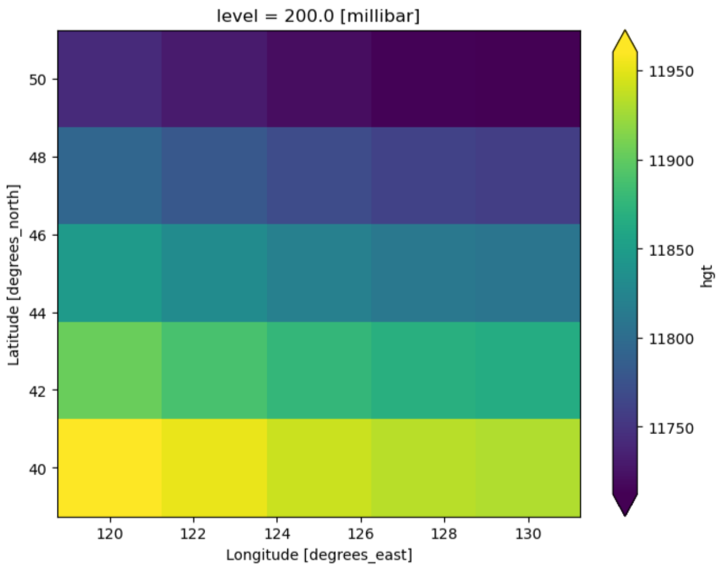
Out[75]: <matplotlib.collections.QuadMesh at 0x206b72dff90>



4、比较 200、500、850hPa 位势高度场空间分布图

In [82]:
```python
#画关键区域的不同高层的 2Dmap
hgt_box.sel(level=200).mean(dim=['time']).plot(robust=True, figsize=(8, 6))

hgt_box.sel(level=500).mean(dim=['time']).plot(robust=True, figsize=(8, 6))

hgt_box.sel(level=850).mean(dim=['time']).plot(robust=True, figsize=(8, 6))
```

Out[82]: <matplotlib.collections.QuadMesh at 0x206b7a4d650>



5、按照每个纬度、经度位势高度场均值

In [91]:
```python
#计算每个经度和纬度上的500hgt位势高度场均值
hgt_box.isel(level=5).mean(dim=['time']).mean(dim=['lon']).plot(marker="o", size=3)
hgt_box.isel(level=5).mean(dim=['time']).mean(dim=['lat']).plot(marker="*", size=3)
# m_net=data.toa_net_all_mon.mean(dim=['time'])
# m_net.sum(dim=['lon']).plot()
```

Out[91]: [<matplotlib.lines.Line2D at 0x206c77694d0>]