

## PS4

### PS4\_1

```
import numpy as np
import xarray as xr
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.ticker as mticker
import cartopy.crs as ccrs
import cartopy.feature as cfeature
%matplotlib inline
```

```
#cite:https://scitools.org.uk/cartopy/docs/v0.14/examples/global\_map.html
```

```
#select the data
```

```
eq= pd.read_csv('usgs_earthquakes.csv')
top_50=(eq.loc[:,['latitude','longitude','mag']].sort_values('mag',
ascending=False).head(50)
lons=top_50['longitude'].values.tolist()
lats=top_50['latitude'].values.tolist()
```

```
#draw the map
```

```
plt.figure(figsize=(10,10), dpi=100)
ax = plt.axes(projection=ccrs.Robinson(central_longitude=180))
ax.set_global()
ax.stock_img()
ax.set_title('Top 50 Earthquakes of 2014')
```

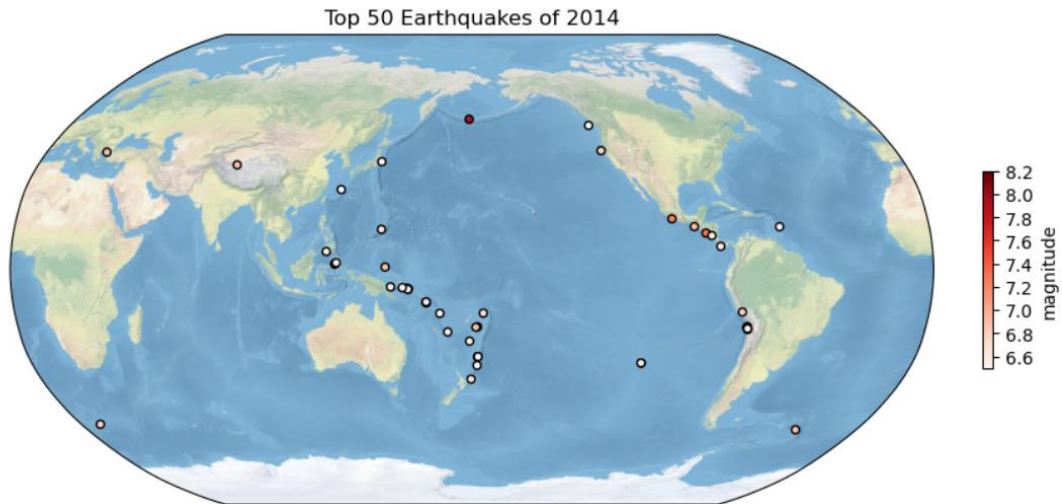
```
#visualize the data on the map
```

```
map_v=ax.scatter(lons, lats, marker='o',c=top_50['mag'], s=20, cmap='Reds',
edgecolors='k', transform=ccrs.PlateCarree())
```

```
#color bar
```

```
ticks=[6.6, 6.8, 7.0, 7.2, 7.4, 7.6, 7.8, 8.0, 8.2]
cbar= plt.colorbar(map_v, label='magnitude', orientation='vertical', fraction=0.01)
cbar.set_ticks(ticks)
```

```
plt.show()
```



## PS4\_2

```
import numpy as np
import xarray as xr
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.ticker as mticker
%matplotlib inline
import cartopy.feature as cfeature
import cartopy.crs as ccrs
from cartopy.mpl.ticker import LongitudeFormatter, LatitudeFormatter
import matplotlib.pyplot as pl
from matplotlib.offsetbox import AnchoredText
```

### #load the data

```
ds = xr.open_dataset("wspd.mon.mean.nc", engine="netcdf4")
wspd=ds.wspd.sel(level=1000).isel(time=-1)
wspd
```

## 2.1

### # Create and define the size of a figure object

```
plt.figure(figsize=(5,5), dpi=100)
```

### # Create an axes with Orthographic projection style

```
central_lon, central_lat = 114.06, 22.54
proj = ccrs.Orthographic(central_lon, central_lat)
ax = plt.axes(projection=proj)
```

### # Add border lines over countries

```
ax.add_feature(cfeature.NaturalEarthFeature(category='cultural',name='admin_0_countries', scale='110m', facecolor='none', edgecolor='black', linewidth=0.5))
```

*# Add lat/lon gridlines, draw gridlines*

```
gl = ax.gridlines(crs=ccrs.PlateCarree(), linewidth=1, color='black',  
alpha=0.5,draw_labels=True)
```

*# Plot the windspeed*

```
wspd.plot(ax=ax, transform=ccrs.PlateCarree(),  
vmin=0, vmax=15, cbar_kwargs={'shrink': 0.4})
```

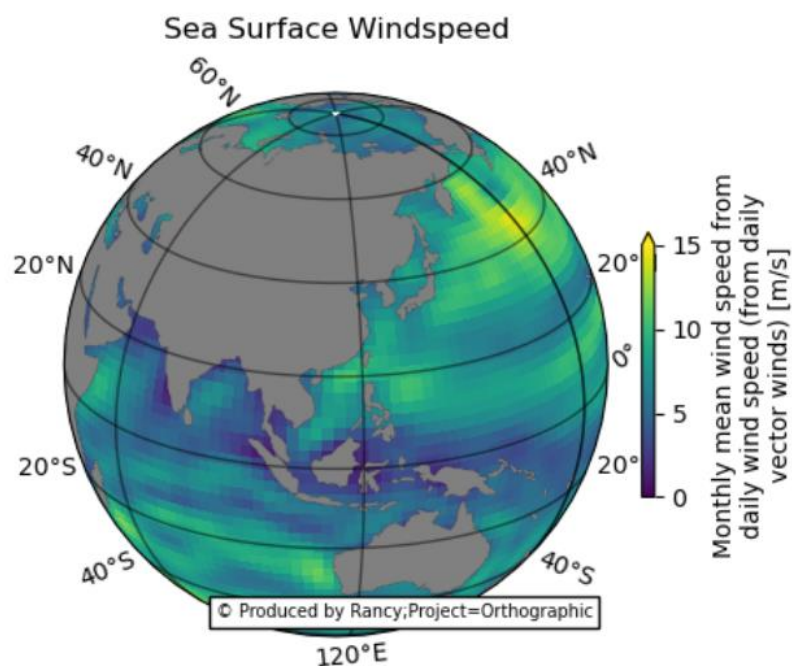
```
ax.set_title('Sea Surface Windspeed')
```

*#Mask land data by adding land feature and changing its zorder*

```
ax.add_feature(cfeature.LAND, edgecolor='grey', facecolor='grey',  
linewidths=0.5, zorder=1)
```

*# Add a text annotation*

```
SOURCE = 'Produced by Rancy; Project=Orthographic'  
text = AnchoredText('\u00A9 {}'.  
".format(SOURCE),  
loc=4, prop={'size': 7}, frameon=True)  
ax.add_artist(text)
```



## 2.2

[#draw a map around ShenZhen](#)

```
fig = plt.figure(figsize=(5,5), dpi=100)
ax = fig.add_subplot(1, 1, 1, projection=ccrs.PlateCarree())
ax.set_extent([104, 124, 12, 32], crs=ccrs.PlateCarree())
```

```
ax.add_feature(cfeature.LAND)
ax.add_feature(cfeature.OCEAN)
ax.add_feature(cfeature.COASTLINE)
ax.add_feature(cfeature.BORDERS, linestyle=':')
ax.add_feature(cfeature.LAKES, alpha=0.5)
ax.add_feature(cfeature.RIVERS)
```

[# Add lat/lon gridlines, draw gridlines](#)

```
gl = ax.gridlines(crs=ccrs.PlateCarree(), linewidth=1, color='black', alpha=0.5)
```

[# Manipulate latitude and longitude gridline numbers and spacing](#)

```
gl.ylocator = mticker.FixedLocator(np.arange(12,32,5))
gl.xlocator = mticker.FixedLocator(np.arange(104,124,5))
```

```
ax.set_xticks([104,109,114,119,124], crs=ccrs.PlateCarree())
ax.set_yticks([12,17,22,27,32], crs=ccrs.PlateCarree())
```

[# Plot the windspeed](#)

```
wspd.plot(ax=ax, transform=ccrs.PlateCarree(),
          vmin=0, vmax=7, cbar_kwargs={'shrink': 0.4})
```

[#set title](#)

```
ax.set_title('Windspeed Around ShenZhen')
```

[# Mask ocean data by adding ocean feature and changing its zorder](#)

```
ax.add_feature(cfeature.OCEAN, edgecolor='grey', facecolor='white',
               linewidths=0.5, zorder=1)
```

[# Add a text annotation](#)

```
SOURCE = 'Produced by Rancy'
text = AnchoredText("\u00A9 {}".format(SOURCE),
                    loc=4, prop={'size': 7}, frameon=True)
ax.add_artist(text)
```

