

Customizing maps

We've seen how to create simple global maps, and taken a look at some of the different projections Cartopy has to offer. But we might want to customize these maps for our purposes. Let's start looking at ways to customize.

credit

As always, lots of this lesson is based on Ryan Abernathy's course:

https://ravernat.github.io/research_computing_2018/maps-with-cartopy.html. Parts too are from the person who wrote the Cartopy package, [Phil Elson](#) , tutorial here:

[https://github.com/SciTools/cartopy-](https://github.com/SciTools/cartopy-tutorial/tree/42cb77062a08063a53e7a511a9681bdb15e70fe7)

[tutorial/tree/42cb77062a08063a53e7a511a9681bdb15e70fe7](https://github.com/SciTools/cartopy-tutorial/tree/42cb77062a08063a53e7a511a9681bdb15e70fe7).

In [32]:

```
# import statements

import cartopy.crs as ccrs
# import cartopy

import matplotlib.pyplot as plt

%matplotlib inline
```

zooming in on a region

We are each probably interested in a particular part of the world.

To customize our map we will want to use the `set_extent` method/function of Cartopy to do this.

How does it work? first create a geoaxis

In [33]:

```
plt.figure()
ax = plt.axes(projection=ccrs.PlateCarree())
```



Use the `?` (or SHIFT+TAB) to figure out what `ax.set_extent()` does:

```
In [34]: ax.set_extent?
```

Signature: `ax.set_extent(extents, crs=None)`

Docstring:

Set the extent (x0, x1, y0, y1) of the map in the given coordinate system.

If no crs is given, the extents' coordinate system will be assumed to be the Geodetic version of this axes' projection.

Parameters

extents

Tuple of floats representing the required extent (x0, x1, y0, y1).

File: `~/anaconda3/envs/swbc2021/lib/python3.8/site-packages/cartopy/mpl/geoaxes.py`

Type: `method`

Let's make a map of our area, use `extent = [-77, -70, 35, 43]` with `ax.set_extent()`

```
In [35]:
```

```
extent = [-77, -70, 35, 43]
ax = plt.axes(projection=ccrs.PlateCarree())
ax.gridlines()
ax.coastlines()
ax.set_extent(extent)
```



This defaults to an appropriate resolution, but you can check out the documentation in `coastlines` to see how to specify higher/lower resolution in the coastline

```
In [36]:
```

```
ax.coastlines?
```

Signature: `ax.coastlines(resolution='auto', color='black', **kwargs)`

Docstring:

Add coastal **outlines** to the current axes from the Natural Earth "coastline" shapefile collection.

Parameters

resolution : `str` or `:class:`cartopy.feature.Scaler``, optional

A named resolution to use from the Natural Earth dataset. Currently can be one of "auto" (default), "110m", "50m", and "10m", or a `Scaler` object.

File: ~/anaconda3/envs/swbc2021/lib/python3.8/site-packages/cartopy/mpl/geoaxes.py
Type: method

remake the same plot but with the resolution set to 50m:

```
In [37]: # Redo this, with low resolution
extent = [-77, -70, 35, 43]
ax = plt.axes(projection=ccrs.PlateCarree())
ax.set_extent(extent)
ax.gridlines()
ax.coastlines(resolution='110m')
```

Out[37]: <cartopy.mpl.feature_artist.FeatureArtist at 0x7fc93a155d60>



Adding Features to the Map

To give our map more styles and details, we add `cartopy.feature` objects. Many useful features are built in. These "default features" are at coarse (110m) resolution.

- `cartopy.feature.BORDERS` Country boundaries
- `cartopy.feature.COASTLINE` Coastline, including major islands
- `cartopy.feature.LAKES` Natural and artificial lakes
- `cartopy.feature.LAND` Land polygons, including major islands
- `cartopy.feature.OCEAN` Ocean polygons
- `cartopy.feature.RIVERS` Single-line drainages, including lake centerlines
- `cartopy.feature.STATES` (limited to the United States at this scale)

to do this, we need to import the `cartopy.feature` part of `cartopy`. We usually do this as

```
import cartopy.feature as cfeature
```

Lets make the same map again, but now use the `ax.add_feature()` function to add ocean, states, and land:

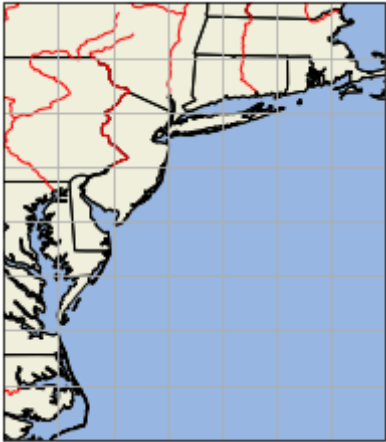
```
In [38]: import cartopy.feature as cfeature

extent = [-77, -70, 35, 43]
ax = plt.axes(projection=ccrs.PlateCarree())
```

```
ax.set_extent(extent)
ax.gridlines()
ax.coastlines()

ax.add_feature(cfeature.OCEAN)
ax.add_feature(cfeature.LAND)
ax.add_feature(cfeature.STATES)
ax.add_feature(cfeature.RIVERS, edgecolor='red')
```

Out[38]: <cartopy.mpl.feature_artist.FeatureArtist at 0x7fc8fb0c29a0>



Exercise 01

In []: