UNIVERSITÄT
LEIPZIG

Lesson 03

Geospatial data

Seminar room 0.06, October 8th, 2025

Francisco José Cuesta-Valero

TODAY IN THE COURSE

I. Statistics in several dimensions.
II. Plotting interesting statistics.
III. Data interpolation and masking.
IV. Spatial projections.
V. Scalar maps.
VI. Contours.
VII. Vector maps.
VIII. Practical session.
IX. Questions and answers.

STATISTICS IN SEVERAL DIMENSIONS

As you already know, xarray can be use to work with multidimensional arrays. Even more, it can estimate statistical quantities in the different dimensions:

```
1 >> avg_temp_over_time = ds['air_temperature'].mean(dim='time')
```

You could also combine xarray and scipy.stats to obtain statistics:

```
1 >> temp_at_location = air_temp.isel(lat=0, lon=0)
2 >> scipy.stats.mean(temp_at_location.values)
```

Or you could also use scipy.stats alone:

```
1 >> scipy.stats.mean(air_temp.values, axis=0)
```

PLOTTING INTERESTING STATISTICS

Once that you have processed your data, you will have some results to plot. For example, you may have a temporal average of a certain region. Then you can use the plot function in matplotlib to create a line plot:

```
1 >> plt.plot(years, values, color='b', marker='o', linestyle='-', markersize=4, label='Data label')
```

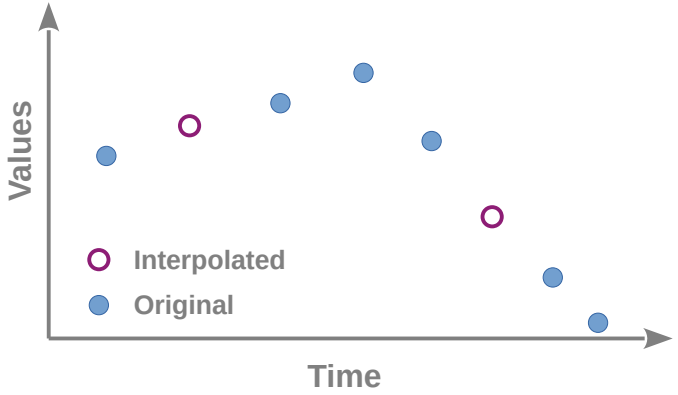
Or perhaps you are comparing the output of a model with some stations, so you have estimated some metric, like RMSE. Then you can use the bar function to plot the results:

```
1 >> plt.bar(station_names, rmse_values, width=bar_width, color=['black', 'red'])
```

More examples can be found in: https://matplotlib.org/stable/plot_types/index.html.

DATA INTERPOLATION AND MASKING

Sometimes, our time series may present missing values, or we may want to fill sampling gaps. This can be done in several ways, one of the most common being temporal interpolation.



Interpolate means to guess a missing data from the surrounding records. There are myriads of techniques, here we will cover just some examples.

DATA INTERPOLATION AND MASKING

An example or temporal interpolation using scipy:

```
1 >> f_linear = scipy.interpolate.interp1d(known_time, known_temperature, kind='linear')
2 >> interpolated_temperature = f_linear(desired_time)
```

An example or temporal interpolation using pandas:

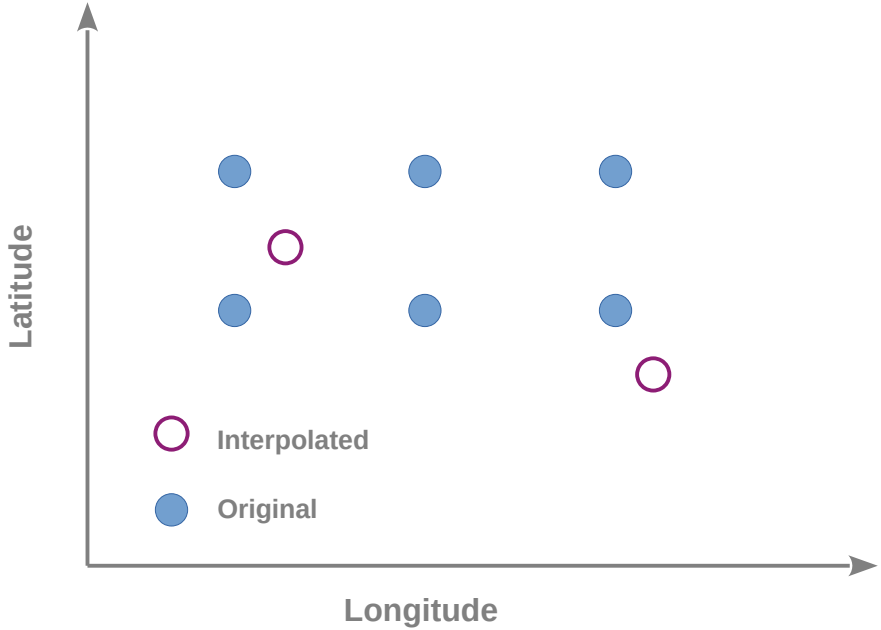
```
1 >> temperature_series = pd.Series([12, 13, 14, np.nan, 15, 16, 17, 18, np.nan, 20, 21, 22, 23, 24, np.nan, 25, 26, 27,
28, 29, 30, 31, 32, 33, np.nan, 34, 35, 36, 37, 38], index=dates)
2 >> interpolated_temperature = temperature_series.interpolate(method='linear')
```

An example or temporal interpolation using xarray:

```
1 >> temperature_xr = xr.DataArray(temperature_array, coords=[dates], dims=['time'])
2 >> interpolated_temperature = temperature_xr.interp(time=temperature_xr.time)
```

DATA INTERPOLATION AND MASKING

As in the case of temporal interpolation, we may have some missing data in a spatial field, or we may need to estimate the value of a variable at a certain latitude and longitude that was not sampled. In this case, we need to apply spatial interpolation techniques.



DATA INTERPOLATION AND MASKING

An example or temporal interpolation using scipy:

```
1 >> grid_z = scipy.interpolate.griddata(points, values, (grid_x, grid_y), method='linear')
```

An example or temporal interpolation using pandas:

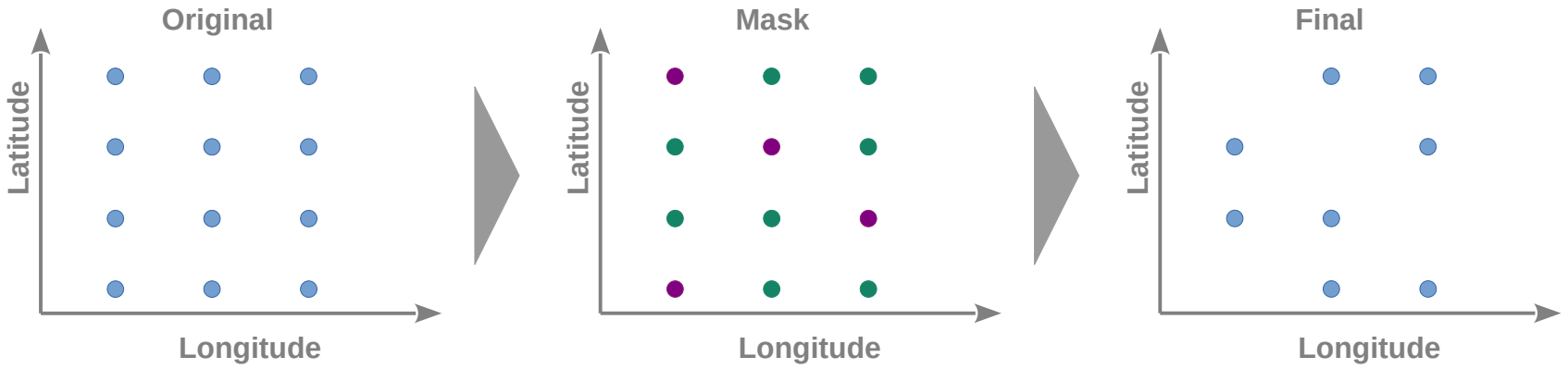
```
```

An example or temporal interpolation using xarray:

```
1 >> soil_moisture_xr = xr.DataArray(data_2d, coords={'x': x_coords, 'y': y_coords}, dims=['y', 'x'])
2 >> interpolated_moisture = soil_moisture_xr.interp(x=x_coords, y=y_coords)
```

DATA INTERPOLATION AND MASKING

There could also be important to remove some part of the data because of some reason. This is what is called masking. The mask typically consist on a dataset with True or False values with the same dimensions that the original dataset, and it is used to remove data that are uninteresting, wrong, or not significant.



DATA INTERPOLATION AND MASKING

An example of spatial masking using numpy:

```
1 >> threshold = 1000
2 >> mask = elevation < threshold
3 >> masked_elevation = np.ma.array(elevation, mask=mask)
```

An example of spatial masking using pandas:

```
1 >> threshold = 1000
2 >> masked_data = data[data['z'] > threshold]
3 >> data['z_masked'] = data['z'].where(masked_data)
```

An example of spatial masking using xarray:

```
1 >> elevation_xr = xr.DataArray(elevation_data, coords={'x': x_coords, 'y': y_coords}, dims=['y', 'x'])
2 >> threshold = 1000
3 >> masked_elevation = elevation_xr.where(elevation_xr > threshold)
```

DATA INTERPOLATION AND MASKING

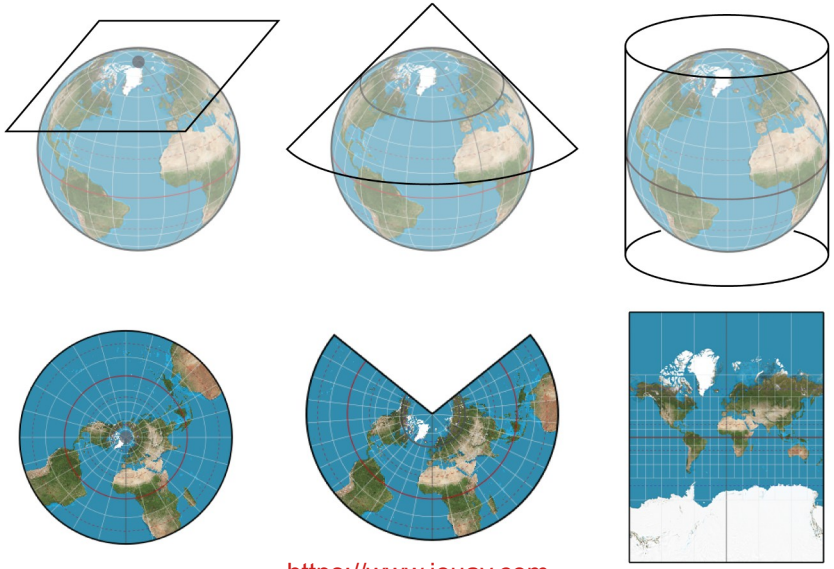
Another example of spatial masking using xarray:

```
>> min_lon, max_lon = -120, -70
>> min_lat, max_lat = 20, 50
>>
>> region_mask = (temperature_xr.lon >= min_lon) & (temperature_xr.lon <= max_lon) & \
>>               (temperature_xr.lat >= min_lat) & (temperature_xr.lat <= max_lat)
>>
>> masked_temperature = temperature_xr.where(region_mask)
```

SPATIAL PROJECTIONS

A spatial projection, also known as a map projection, is a method used to represent the curved, 3D surface of the Earth on a flat, 2D plane.

The Earth is a sphere (or more accurately, an oblate spheroid). When we project this shape onto a flat surface, we introduce distortion. This means that properties like area, shape, distance, and direction can't all be perfectly preserved. Every projection has its own unique distortions.

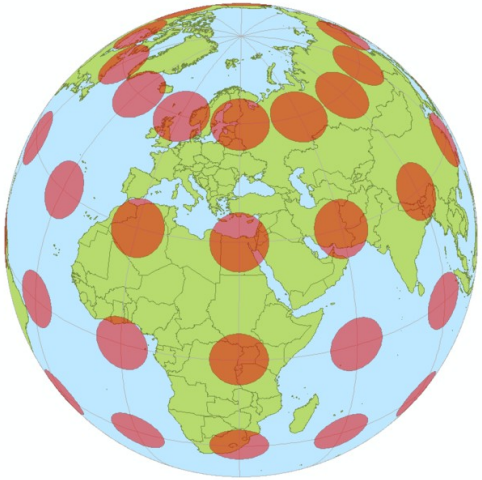


https://www.jouav.com

SPATIAL PROJECTIONS

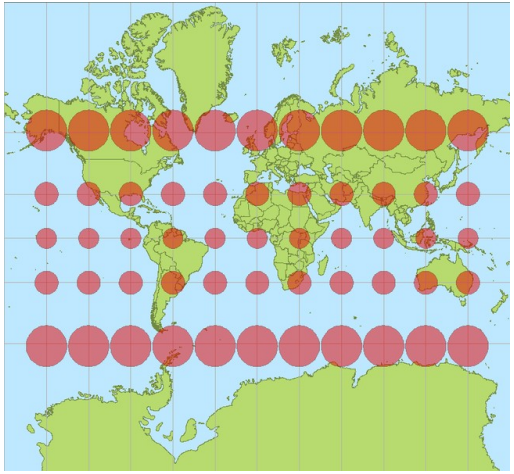
Tissot's distortions are a way to visually understand and quantify the distortions that occur when we project the curved surface of the Earth onto a flat map

Orthographic



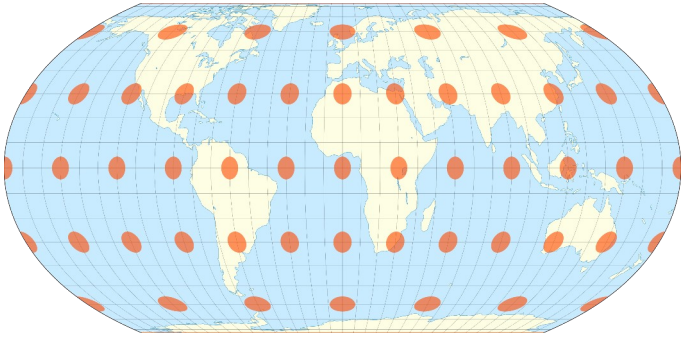
Wikipedia

Mercator



Wikipedia

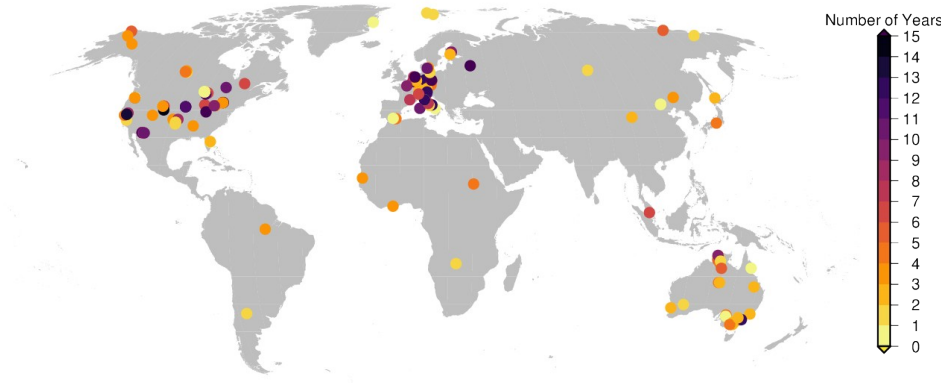
Equal Earth



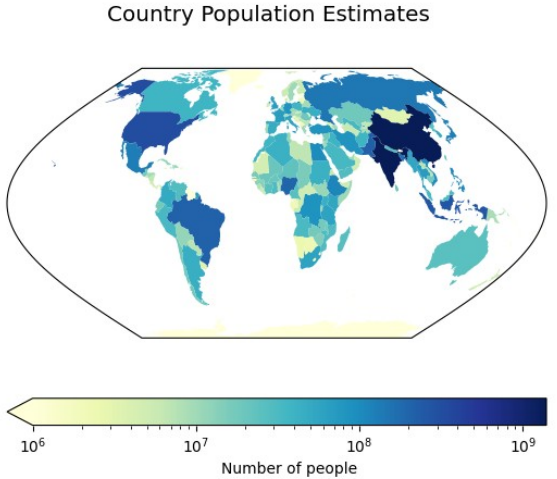
Wikipedia

SCALAR MAPS

Maps representing dot information or area information are called scalar maps. In these maps, each information record is represented with a color or a shade of a color.



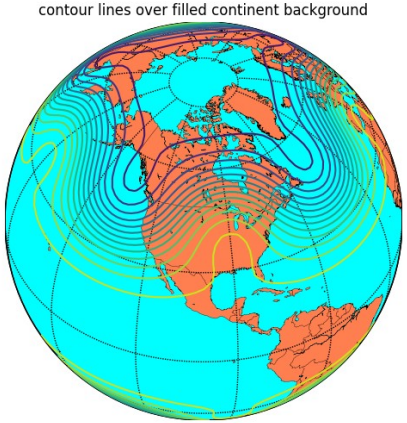
Cuesta-Valero, 2025



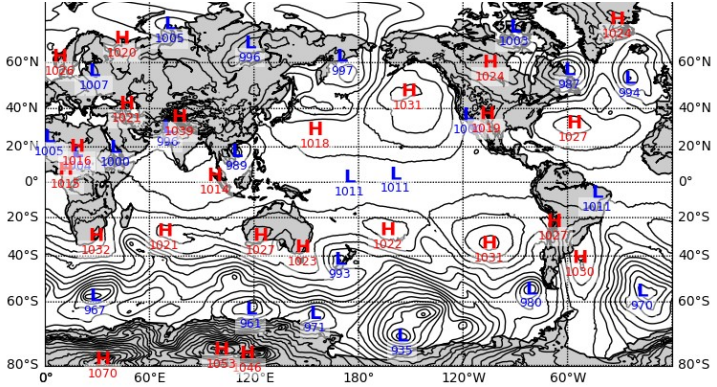
cartopy.readthedocs.io

CONTOUR MAPS

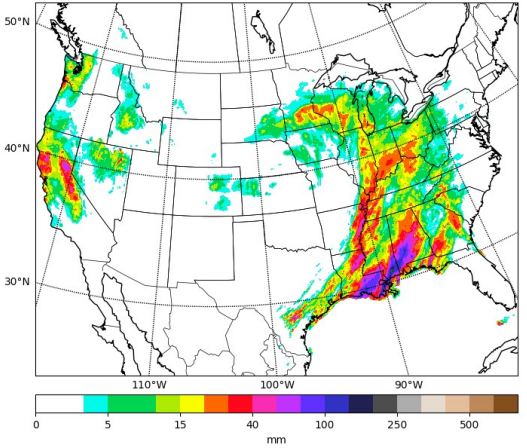
When scalar maps include too many points, it is not practical to plot each record. Instead, a more efficient approach consist in representing all points within a certain range with a color or a line. That is, to create a contour plot.



matplotlib.org



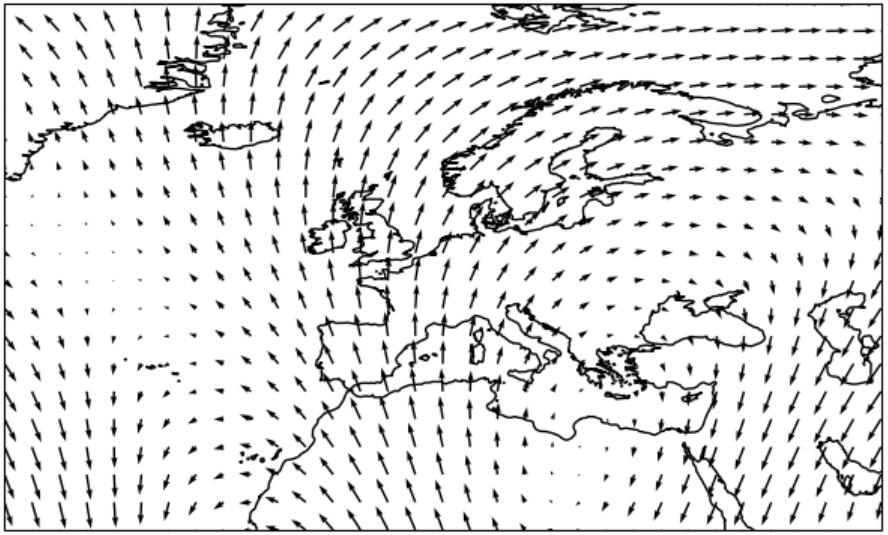
matpliotlib.org



matpliotlib.org

VECTOR MAPS

Maps representing vector information, or data with both magnitude and direction, are called vector maps. In these maps, each information record is represented with a sign indicating value and direction.



cartopy.readthedocs.io

UNIVERSITÄT
LEIPZIG

VIELEN DANK!

FEEDBACK IS ENCOURAGED!

Francisco José Cuesta-Valero
Institute for Earth System Sciences and Remote Sensing

Talstraße 35, 04103, Leipzig

francisco_jose.cuesta_valero@uni-leipzig.de
cuestavalero.github.io