Iris
A python package for the analysis and visualisation of Meteorological data

Philip Elson
30th Sept 2015
Outline

- What is Iris?
- Iris demo
- Using Iris for novel analysis
- Opportunities for combining Iris with other tools

Audience of this talk:

- Those who write code to do data analysis and visualisation
What is Iris?
What is Iris?
A simple example
>>> import iris

>>> air_temp = iris.load_cube(filename, 'air_temperature')

>>> print(air_temp)

air_temperature / (K) (latitude: 73; longitude: 96)
Scalar coordinates:
  pressure: 1000.0 hPa
  time: 1998-12-01 00:00:00, bound=(1994-12-01 00:00:00, 1998-12-01 00:00:00)

Attributes:
  STASH: m01s16i203
  source: Data from Met Office Unified Model
>>> import matplotlib.pyplot as plt
>>> import iris.quickplot as qplt

>>> qplt.pcolormesh(air_temp, cmap='RdBu_r')
>>> plt.gca().coastlines()
Regridding and interpolation

```python
>>> from iris.analysis import Linear

>>> exeter = [('longitude', [-3.5]), ('latitude', [50.7])]

>>> exeter_temp = air_temp.interpolate(exeter, Linear())

>>> mslp_euro = iris.load_cube(filename2)

>>> air_temp_euro = air_temp.regrid(mslp_euro, Linear())
```

Typically, Iris takes cubes as input, and returns cubes as output.
Maps in Iris are drawn by cartopy, a python package developed to solve common dateline and pole problems seen with traditional mapping libraries.

```python
>>> from cartopy.crs as ccrs
>>> ax = plt.axes(projection=ccrs.NorthPolarStereo())
>>> qplt.pcolormesh(air_temp_euro, cmap='RdBu_r')
>>> ax.coastlines('50m')
>>> ax.gridlines()
```
A real-life example
MOGREPS-G Cyclone Database

An algorithm to identify and track fronts and cyclonic features, based on:

Implementing the algorithm

• Load the phenomenon

• Regrid and interpolate data to specific to vertical levels

• Compute isolines for locating phenomenon + isosurfaces for masking phenomenon, based on thresholds from paper

• Compute intersection of isosurfaces and isolines to identify cyclonic features
• Classify cyclonic features based on phenomenon values
  • Visualise cyclonic features and the underlying diagnostics

- Barotropic Lows
- Frontal Waves
- Diminutive Waves
• Visualise fronts as a spaghetti plot
The Python ecosystem
Opportunities within Python

Recent GIS tools:

- Shapely
- Cartopy
- Fiona
- RasterIO
- QGIS

A recent publication combining shapely and Iris to assess the skill of seasonal prediction of Hurricane landfall frequencies in the North Atlantic:

Opportunities within Python

Large data manipulation:

- Cython
- Numba
- Biggus
- Dask

Tools to optimise slow for-loops using static typing and JIT compilation for C-like performance
Opportunities within Python

Large data manipulation:

- Cython
- Numba
- Biggus
- Dask

**Biggus example:**

```python
>>> print(data)
<Array shape=(80640, 4, 144, 192)
dtype=dtype('float32') size=33.22 GiB>

>>> stats = [biggus.mean(data, axis=0),
           biggus.max(data, axis=0),
           biggus.min(data, axis=0)]

>>> biggus.ndarrays(stats)

Result in ~4m45s on an Intel Xeon E5520 with 8GiB memory, bound by I/O not CPU.
```

Iris is using Biggus for many of its operations. This means that we can load, analyse and save cubes way beyond the available system memory.
Installing Iris

conda install iris --channel SciTools

Conda can be downloaded as part of “miniconda”: http://conda.pydata.org/miniconda.html
Questions

Further reading:
github.com/scitools/courses

Links from presentation:
github.com/pelson/ecmwf-vis-2015

Github: github.com/pelson
Twitter: @pypelson