Chapter 12

Plotting Berkeley Earth data (Group A, week 4)

In this week's assignment you will create plots of more Berkeley Earth data. The goals for this week are to:

- write a small program split into several modules,
- create properly structured modules,
- document code properly, and
- create the basic plots (scatter plot, line plot, histogram).

Note: properly label your plots also have a look at the hints section below.

12.1 Gathering data

Like last week we will be using Berkeley Earth data because it is nice and homogeneous. The course website has a tarball with all the necessary files.

- Download the weather data provided in a tarball on the course website `astro.mprog.nl` use one of the files in that tar ball to check that the parser you write works.
- Create a Python module called `weather_parser.py`. In this module create a function called `parse_location_data` that accepts a file name (as string) and returns a list of tuples with each tuple containing a year (as integer), a month (as integer), a temperature anomaly (as a floating point number), and the uncertainty on the temperature anomaly (again a floating point number. It is a good idea to refer to the code you wrote last week.
- Add a `if __name__ == '__main__'`: line to the module and move any code that calls the `parse_location_data` function to the block behind it. This will ensure that at import your module will not start parsing a file — in other words, that your module is safe for re-use.

12.2 Creating a line plot

You will now start creating plots using Matplotlib.

- Create a Python module called `plotter.py`, you will use this module for all the plotting tasks below. Import the `pyplot` module of Matplotlib using: `from matplotlib import pyplot`
- Import your `weather_parser` module.

- Create a `main` function that you will use to call the various plotting functions.
- Create a function called `plot_timeseries` that uses that `pyplot.plot` function to create a line plot of the temperature anomaly for one of the data files. The plot should be saved as `time-series.pdf` . As a X coordinate you are allowed to use years as floating point numbers (with the fractional part set to the appropriate value for each month) if you think using Python dates is too hard.
- Call the `plot_timeseries` function from `main` function.
- Extra credit (do this after you finish the rest of the assignment): Plot a gray background around the time-series showing the uncertainty in the data (like the plots on the Berkeley Earth website have).

12.3 Creating a scatter plot

- In your `plotter.py` module create a function `plot_scatter` that creates a scatter plot of uncertainty versus year that the data was taken. Is there any pattern you see? Call this function from `main` and save the resulting plot as `scatter.pdf`

12.4 Creating a histogram

For this part of the exercise you should download the data for all the locations in the data file.

- Create a function called `plot_histogram` in `weather_parser.py` that accepts a list of file names (for the location data files) and produces a histogram of the temperature anomalies of all the cities at January 2000.
- You will need to loop over all the file names and call the parser function for each location. Keep a list of all the temperature anomalies, use that to make the histogram.
- Save the histogram as: `histogram.pdf`

12.5 Hints

- For a plot of two variables `pyplot.plot` expects two lists of equal lengths, one containing all X values and one containing all Y values. In our case that means that you have to take the list of tuples you created and turn it into separate lists (for time values, for temperature anomalies etc.).
- If you are having problems importing your own module make sure that both modules (the one doing the importing and the one being imported) are both in the same directory.

12.6 Handing in

You should make a gzipped tarball (called `week4.tar.gz`) with the following files:

- `weather_parser.py`
- `plotter.py`
- `time-series.pdf`
- `scatter.pdf`
- `histogram.pdf`
- `report-week-4.pdf` (Only if you want to explain your methods.)