

CDS 130 - 003 -- Fall 2010
Computing for Scientists

Project 2: Global Temperature Study

Assignment Date: Nov. 23, 2010
Due Date: Dec. 07, 2010
Presentation: Dec. 07, 2010

1. Introduction

The objectives of this project are to (1) learn how to visualize a large scale, two-dimensional scientific data set; and (2) analyze that data using the methods of statistical measure, histogram and linear regression; and (3) make an effective scientific presentation.

2. Data and Data Loading

The data provided to you are global long term temperatures gathered by National Centers for Environmental Prediction (NCEP) (Credit to Dr. Donglian Sun for converting the data from native binary format to ascii format). These data specify the temperature (in degrees Celsius) at every grid point around the globe, for every latitude from South 90 degrees to North 90 degrees, and every longitude from zero degrees to 360 degrees. Therefore, there are in total $180 \times 360 = 64,800$ data points in the provided data file.

The data file can be downloaded from the class website at:

http://solar.gmu.edu/teaching/2010_CDS130/projects/proj2/temp_ascii.txt

You will need to save the file into Matlab's/Octave's working directory. In the file, the data values are stored in ASCII format in a single column. To read the data into a Matlab/Octave program:

```
data=dlmread('temp_ascii.txt','') %generate 1-D array with 64800 elements
```

To convert the 1-D data array into a 2-D data array, with rows representing latitude and columns representing longitude:

```
for i=[1:180]
    for j=[1:360]
        t(i,j)=data((i-1)*360+j);
    end
end
```

Now, you have the data in Matlab/Octave in 1-D format, stored in the variable "data", and in 2-D format, stored in the variable "t". Check the dimensional size of these two variables using Matlab's/Octave's "size" command

3. Task 1 - Visualization

You'll need to use visualization tools to visualize the global properties of the data:

1. First, you'll need to use the surface map or 3-D height plot to visualize the 2-D temperature map. The appropriate Matlab/Octave function to use is "surf"; and
2. Second, you'll need to use the Matlab/Octave function "image" to plot a colormap of the global 2-D temperature data. Make sure that your data are scaled to fit the color range of "image" method.

4. Task 2 - Statistical Measures

Find the following statistical measures of the temperature data: minimum, maximum, mean or average, variance and standard deviation.

5. Task 3- Data Analysis

1. First, use the Matlab/Octave function "hist" to plot a histogram of the data. The bin data range is -40 to 40 degrees, and each bin size is 5 degrees. From the histogram plot, find the temperature bin of highest frequency. Make sure to label the axes on your plot, and to give your plot a title.
2. Second, find the average temperature over all longitudes **for each latitude**, from South 90 degree (matrix "t" row 1) to North 90 degree (matrix "t" row 180). This results in a new 1-D data array of 180 points (say, stored in variable "t_ave"), indicating how temperature changes with latitude. Using this newly created 1-D data array, plot average temperature versus latitude. Again, make sure to label the axes on your plot, and, to give your plot a title.
3. Third, make a linear correlation analysis of temperature versus latitude. The latitude can be obtained by converting the row index I, as follows (the latitude variable's range is -90 to 90):

```
lat = [1:180]-90
```

In order to make a correlation analysis between temperature and latitude, you'll need to convert the signed latitude into an absolute latitude. This can be done as follows:

```
lat_abs = abs(lat)
```

Now, you have two new variables: "lat_abs" and "t_ave".

You are required to use the Matlab/Octave function "polyfit" to find the **linear** correlation function between the data stored in "lat_abs" and "t_ave" (linear means that $n = 1$ when using "polyfit"). Furthermore, you're also required to use the Matlab/Octave function "corrcoef" to find the correlation coefficient.

Finally, you're required to make a linear correlation plot, showing the scattering of the data points (average temperature vs. latitude), and also, the line fitted to these data that you obtained using the Matlab/Octave function "polyfit".

6. Presentation

You are required to make an oral presentation of this project to the class on Dec. 7, 2010 (Tuesday). You'll need to prepare your presentation in Microsoft PowerPoint (PPT) format, probably no more than 6 slides. The presentation time is up to 8 minutes, including questions. You'll need to (1) explain the data analysis methodology; and (2) briefly discuss the scientific meaning of your results.

7. Submission

You are required to write a project report summarizing your results, as required by tasks 1, 2 and 3 above. Images that you create must be included in the report. Your Matlab/Octave computer code also needs to be submitted with the report (either embedded in the report as text, or as a separate file ".m" file).

8. Project Grading:

This project will be worth 100 points, distributed as follows: (1) (25 pts) Overall writing of the project, including the clarity of the writing, proper labeling of the figures, sufficient discussion on your results, and the inclusion of the computer code, (2) (15 pts) the success on task 1, (3) (10 pts) the success on task 2, (4) (25 pts) the success on task 3, (5) (25 pts) the overall quality of the oral presentation.

Bonus task 1 (5 pts):

What is the average temperature in Fairfax, Virginia, as inferred from the data?

Bonus task 2 (5 pts):

Where is the location of highest average temperature, in terms of longitude and latitude?