Chapter 1

Global Climate

1.1 introduction

Informally, we can define climate as the typical weather experienced in a particular region. More rigorously, it is the statistical description, including mean and variability, of relevant variables over a specified period of time. The classical period is 30 years, defined by the World Meteorological Organization. Some commonly used climate variables include temperature, precipitation, humidity, and wind measured at Earth's surface.

In this lab we will use global surface temperature, land surface precipitation, and surface elevation data sets to investigate the broad climate patterns established by the general circulation of Earth's atmosphere.

1.2 the data

1.2.1 surface elevation and the geoid

The MATLAB globaltopo.mat contains variables storing Earth's present day surface elevation and a reconstruction of surface elevation, ice sheets and all, at the last glacial maximum (LGM). The heights are in meters, gridded to a 1° graticule. The LGM reconstruction derives from a numerical inverse model driven by observed isostatic adjustment rates and a notion of the dynamics of Earth's interior. The reference for that work is Peltier W.R., 2004, Global Glacial Isostasy and the Surface of the Ice-Age Earth: The ICE-5G(VM2) model and GRACE. *Ann. Rev. Earth. Planet. Sci.*, 32, 111-149.

The geoid is a surface along which the gravitational potential is everywhere equal and to which the direction of the acceleration of an object due to gravity is perpendicular. The geoid can be thought of as the surface a globe-covering ocean would form in response to the combined effects of Earth's mass attraction (gravitation) and the centrifugal force due to rotation. Because Earth's mass is not distributed uniformly, the geoidal surface is irregular. The geoid height is the difference between the geoid and an ellipsoid fit to Earth's polar and equatorial radii (this is called the reference ellipsoid or the ellipsoid of revolution). The geoid height can be used to determine if a particular location on Earth's surface is above or below sea level.

```
% globaltopo.mat    Earth surface elevation, meters
%   topoE       present-day 180 x 360 matrix, 1 degree grid
%   geoidE      present-day geoid height  180 x 360 matrix, 1 degree grid
%   IceAge      last glacial maximum as computed by Peltier (ICE-5G)
%               180 x 360 matrix, 1 degree grid
%   maplat maplon:    latitude and longitude, degrees
%
% reference
%  Peltier, W. R., 2004, Global Glacial Isostasy and the Surface of the
%  Ice-Age Earth, Ann. Rev. Earth. Planet. Sci., 32, pp 111-149.
```

1.2.2 climatology

The MATLAB file globalclimatology .mat contains temperature and precipitation data gridded to a 0.5° graticule.

- Ta and Ta_ann

  Ta contains mean monthly surface temperature from terrestrial observations of shelter-height air temperature and shipboard measurements in degrees Celsius. The data are stored in a multi-dimensional array with a layer for each of the 12 months so that Ta (:,:,1) stores January data, Ta (:,:,2) stores February data, and so on. Annual means have been computed for you.

  The land station records are for the years between 1920 and 1980. Median air temperatures over the oceans are taken from the Comprehensive Ocean-Atmosphere Data Set (COADS) for the years 1950-79. The reference for these data is Legates, D. R. and C. J. Willmott, 1990, Mean Seasonal and Spatial Variability Global Surface Air Temperature, *Theoretical and Applied Climatology*, 41, 11-21.

- precip & wdays

  Mean monthly precipitation in mm day^{-1}and mean monthly total days with precipitation (defined as 0.1 mm or more) for the time periods 1979 to 1999 and 1960 to 1990, respectively. The data are stored in multi-dimensional arrays with a layer for each of the 12 months, so that precip (:,:,1) stores January data, precip (:,:,2) stores February data, and so on. These data sets do not provide coverage at high latitudes because observational data are sparse in those regions.

  The precipitation data were regridded from a one degree resolution to 0.5 degree resolution to facilitate comparison with other data sets used in this lab. The precipitation data sets represent a synthesis of satellite and surface rain gauge data and thus begin in 1979. The wdays data set uses rain gauge data only.

The data references are:
Adler, R.F., G.J. Huffman, A. Chang, R. Ferraro, P. Xie, J. Janowiak, B. Rudolf, U. Schneider, S. Curtis, D. Bolvin, A. Gruber, J. Susskind, and P. Arkin (2003). The version 2 Global Precipitation Climatology Project (GPCP) monthly precipitation analysis (1979-Present). *Journal of Hydrometeorology*. Archived at the ORNL DACC `http://daac.ornl.gov/`.

New, M. G., M. Hulme and P. D. Jones, 2000, Representing 20th century space-time climate variability. II: Development of 1901-1996 monthly terrestrial climate fields, *Journal of Climate*, 13, 2217-2238. Archived at the Intergovernmental Panel on Climate Change Data Distribution Center, `http://ipcc-ddc.cru.uea.ac.uk/obs`.

The complete set of variables is as follows:

```
% Ta             monthly mean temperature in degrees C,
%                0.5 degree gridded data  360x720x12
%                layer 1: January; layer 2: February; etc.
% Ta_ann          mean annual surface air temperature, from Ta
% Tlat           grid latitudes
% Tlon           grid longitudes
% precip         mean monthly precip. rate over land and ocean
%                1979 to 1999; millimeters/day; 0.5 degree graticule
%                360x720x12
%                layer 1: January; layer 2: February; etc.
% plat           grid latitudes
% plon           grid longitudes
```

1.2.3 zonal means

Because Earth's radiation balance and atmospheric circulation have a strong zonal (varying with latitude) structure, it is often instructive to make comparisons between zonal mean values of climate variables. Several such data sets have been prepared for you using the other data sets in this lab and are stored in the data file zonalclimatology.mat.

```
% annual arrays: 180x1    monthly arrays: 180x12
% Taal_zm       zonal mean annual land surface temperature, degrees C
% Taa_zm        zonal mean annual land + ocean surface temperature, degrees C
% SST_zm        zonal mean annual sea surface temperature, degrees C
% Tal_zm        zonal mean monthly land surface temperature, degrees C
% Ta_zm         zonal mean monthly land + ocean surface temperature, degrees C
% maplat        latitudes corresponding to Taal, Tal, SST zonal data
% precipl_zm    zonal mean monthly land surface precipitation, mm/day
% precip_zm     zonal mean monthly land + ocean precipitation, mm/day
% pzmlat        latitudes corresponding to precip_zm
```

1.3 data interpretation

We will use the MATLAB® programming environment to work with the global climatology data.

1.3.1 the figure of the earth

The MATLAB script map_earthsurf.m generates several colormaps using a simple Plate Carree projection: Earth's surface elevation for the present and as reconstructed for the last glacial maximum; the difference between those two surface elevations; and the geoid height. The script requires the globaltopo.mat datafile and uses a MATLAB-native file coast.mat that contains a vector representation of the global coastine.

One example from map_earthsurf.m is provided here. All of the flat maps are produced in the same way.

```
load globaltopo.mat
load coast              % Matlab native file

wind=[min(maplon) max(maplon) min(maplat) max(maplat)];     % plot window
Cmint=-9000;            % minimum surface elevation for color
Cmaxt=7000;             % maximum surface elevation

%******
% Earth surace elevation
%******

figure(1)
clf
pcolor(maplon, maplat, topoE)
shading flat
axis equal
axis(wind)
hold on
caxis([Cmint Cmaxt])
plot(long, lat, 'k-')
colorbar('horiz')
title('Earth surface elevation (m) on a 1x1 grid')
xlabel('longitude')
ylabel('latitude')
```

The first line instructs MATLAB to open a figure window numbered 1. The next statement clears the figure window in case anything was already drawn there. The MATLAB function **pcolor** is used to render a colormap of the data stored in topoE with axis values as specified in maplon and maplat. The option flat is passed to the function **shading** to supress drawing boundaries between map grid cells. The grid tick mark spacing is set to be equal for the latitudes and longitudes and the maximum and minimum longitudes and latitudes stored in the variable wind, are used instead of whatever MATLAB produced as a default. The function **caxis** is used to define a color range for the rendering. The function plot is used to draw present-day coastlines on the map with long and lat from the MATLAB data file coast.mat. A figure title and axis labels are added as well.

Additional sections of the script repeat this procedure for IceAge, IceAge−topoE, and geoidE.

You can see that if you wanted to make very many maps you would be tying the same sets of commands over and over. The process can be simplified by writing a **function** that performs the same set of tasks every time it is invoked. A generalized map rendering function—introduced in the next section—may be downloaded via the class website and is presented in section 1.4 of this lab. It will be used in the remainder of this lab.

1.3.2 temperature

Let's begin with something simple, rendering a map of the mean annual temperature at Earth's surface. The array Ta_ann has dimension 360 x 720 in which there are 360 latitudes and 720 longitudes in the grid. The array storing the mean monthly values has an additional dimension, "layers" for the months, so it is 360 x 720 x 12. We can make a quick map at the command line in the command window:

```
>> figure(100)
>> pcolor(Tlon, Tlat, Ta_ann)
>> hold on
```

The hold is turned on so we don't accidentally replace our map. It will be impossible to view the colors on the map because grid lines have been drawn on the figure. To remove these, reset the shading

```
>> shading flat
```

Next, we correct the projection and tidy up the figure using the **axis** function

```
>> axis equal, axis tight
```

Next, we fix the color scale, draw a colorbar so we can read the map, and generate a title and axis labels

```
>> caxis([min(min(Ta_ann)) min(min(Ta_ann))])
>> colorbar
>> title('mean annual surface air temperature (^{o} C)')
>> xlabel('longitude')
>> ylabel('latitude')
```

The carat tells the MATLAB function **title** to make a superscript using whatever is contained in the adjacent brackets. An underscore can be used in the same way to create a subscript. The figure looks pretty good but it might be easier to read with the coasts of the continents drawn on top of the colormap. We can use the MATLAB-native coast.mat to draw this. The file contains 9865 x 1 arrays called lat and lon that define a global coastline.

```
>> load coast
>> plot(long, lat, 'k-')
```

We can accomplish the same thing using the function f_map. The function may be used with any of the gridded data sets. A function accepts a list of variables and uses them to perform a task or set of tasks so what matters most is that you pass information to the function in an expected sequence. The variables needed by the function may be defined at the command line or in a script. If you plan to make a large number of maps, keeping a record of your work in a script makes it easy to return to past work to make adjustments.

In the example below, all of the variables needed to render a map of the mean annual precipitation are defined and then passed to f_map. The comments are for your information, you do not need to type those when you use the function. Two options are shown for setting the value range Cax for the color bar, one in which the maximum and minimum data values are identified and used and another in which a specified range is centered around zero. It is convenient to set a fixed color range when you plan to make and compare several maps of the same variable (for example, January and July precipitation rates).

```
>> ts='1961 to 1990 mean annual temperature (^{\circ}C)'; %title
>> FN=100; %figure number
>> Cax=[min(min(Ta_ann)) max(max(Ta_ann))];  %whole range
>> Cax=[-70 70]; %set specific bounds
>> xls='longitude';
>> yls='latitude';
>>
>>  f_map(FN, Ta_ann, plat, plon, Cax, ts, xls, yls)
```

Maps are made using the mean monthly data stored in Ta in the same way, though the correct month layer must be extracted from the cube of data. In the example below, the variable mapM is used for this purpose and the layer is identified as all the rows and all the columns in layer mapM: Ta (:,:, mapM). Three color axis options are demonstrated below.

```
>> ts='1961 to 1990 mean January temperature (^{\circ}C)';
>> mapM=1;
>> FN=101;
>> Cax=[min(min(min(precip))) max(max(max(precip)))]; %whole range
>> Cax=[min(min(Ta(:,:,mapM))) max(max(Ta(:,:,mapM)))];  %just mapM
>> Cax=[-70 70]; %set specific bounds
>> xls='longitude';
>> yls='latitude';
>>
>>  f_map(FN, Ta(:,:,mapM), plat, plon, Cax, ts, xls, yls)
```

1.3.3 precipitation

1.3.3.1 precipitation rate

It is important to choose a color range that allows you to see all the features you wish to study in the rendered color map. If one region experiences very large precipitation rates compared to global means, selecting a color range that includes the very large rates will obscure more subtle features. This problem may be suppressed by setting the upper limit of the color range below the global maximum, for example Cax=[−70 70]. A few locations on the map would "saturate" at the maximum color but the rest of the map would be easier to interpret. An alternative could be to use a logarithmic color scale.

1.3.3.2 total precipitation and precipitation intensity

The precipitation rate data may be used to compute two additional useful values, the total annual precipitation and the precipitation intensity. The total precipitation is the sum of the monthly precipitation totals, which in turn are the product of monthly precipitation rate and the time interval over which they apply. This can be approximated simply:

```
ppa=precip*365/12;
precip_ann=sum(ppa,3)/1000;
```

Precipitation intensity is the quotient of the annual total precipitation and the number of wet days. This climate index allows us to differentiate regions characterized by persistent, low-rate precipitation from regions characterized by infrequent, high-rate precipitation events without searching through hundreds of daily or weekly records.

1.3.4 zonal means

A script to plot the computed monthly zonal means, plot_zonal.m is provided for you at the class website. You can also create simple plots using commands typed in at the command line. For example, you can quickly compare the mean annual zonal land surface and ocean surface temperatures in the following way:

```
>> figure(50)
>> plot(maplat, Taal_zm, 'r-')
>> hold on
>> plot(maplat, SST_zm, 'b-')
```

The script plot_zonal.m differs from the earlier programs in that it uses a **for** loop to work through the sets of monthly data, instead of plotting each month individually. The **for** statement uses a index variable n which starts with a value of 1 and increments with the (default) value of 1 up to 12. Thus, n is assigned the value 1, then 2, then 3, and so on, until it reaches its maximum possible value, 12. The maximum desired value for n is determined by measuring the second dimension of the zonal data array. MATLAB's **legend** function is used to create a legend in the figure window.

The script produces four figures and the code for two of the figures is listed below as examples.

```
load zonalclimatology

% strings to plot 12 months with different colors and line types
% and create a legend
pcs=['r-  '; 'g-  '; 'c-  '; 'm-  '; 'b-  '; 'k-  '; 'r-.'; 'g-.'; 'c-.';
    'm-.'; 'b-.'; 'k-.'];
lgs=['Jan'; 'Feb'; 'Mar'; 'Apr'; 'May'; 'Jun'; 'Jul'; 'Aug'; 'Sep'; 'Nov'; 'Dec']

%* plotting windows for Ta and precip
windt=[-90 90 floor(min(min(Tal_zm))) ceil(max(max(Tal_zm)))];
windp=[-90 90 0 ceil(max(max(precip_zm)))];


%*****
%  plot 12 months of zonal mean temperature
%*****
figure(201)
clf
axis(windt)
hold on
for n=1:size(Tal_zm)*[0 1]'
    plot(maplat, Tal_zm(:,n), pcs(n,:))
end
grid on
legend(lgs);
title('zonal mean surface air temperature over land')
xlabel('latitude')
ylabel('temperature (degrees C)')


%*****
%  plot 12 months of zonal mean precipitation
%*****
figure(203)
clf
axis(windp)
hold on
for n=1:size(precip_zm)*[0 1]'
    plot(maplat, precip_zm(:,n), pcs(n,:))
end
grid on
legend(lgs);
title('zonal mean precipitation rate over land')
xlabel('latitude')
ylabel('precipitation rate (mm/day)')
```

1.3.5 lapse rate

The lapse rate is defined as the negative of the rate of change in temperature with altitude through an atmosphere. There are three types of lapse rate,

- the environmental lapse rate, which is the observed change of temperature with altitude for the stationary atmosphere
- the wet and dry adiabatic lapse rates, which are theoretical values, for saturated and dry atmospheres, respectively.

An adiabatic process is one in which the net heat transfer to or from a fluid is zero. The wet and dry adiabatic lapse rates are pressure- and temperature-dependent.

The International Civil Aviation Organisation defines an international standard lapse rate of 6.5 °C km^{-1} from sea level to an altitude of 11 km. The global climatology and topography data sets can be used together to study the environmental lapse rate.

The lapse rate is, in essence, the slope of a line fit to the relationship between air temperature and elevation. We can plot that relationship using Ta_ann and topoE. In doing this, we must keep two things in mind:

1. The mean temperature data sets are gridded at twice the resolution of the topography data sets so we must extract data from Ta_ann to fit the resolution of topoE.
2. topoE contains surface elevations for the whole globe, which are negative below the sea surface. We should only plot the relationship for points above the sea surface.

The following commands, which you can execute in the command window or save together in a script, plot mean annual values for the entire globe. First, MATLAB's find function is used to create a list (called goodies) of elements in the surface topography array with surface elevations above sea level. This is accomplished by comparing topoE and geoidE. Next, colon notation is used to extract temperatures at a 1° spacing from the original 0.5° data set.

```
load globaltopo
load globalclimatology

goodies=find(topoE>geoidE);
Tplot=Ta_ann(1:2:360,1:2:720);

figure(100)
clf
plot(Tplot(goodies), topoE(goodies), 'b.')
hold on
title('mean annual atmospheric temperature lapse rate')
xlabel('temperature (C)')
ylabel('altitude (m)')
```

You can alter these commands to plot a month instead of the mean annual air temperature or to plot data for specific latitudes. The following lines create arrays to plot just one latitude, in this case 45N, and then plot the data on top of the figure created above. You could also open a new figure window and plot the data there if you wish to examine them separately.

```
%* plot one latitude
SL=45;                              % define the latitude
gse=find(round(maplat)==SL);        % find closest grid latitudes
gse=gse(1);                         % may find two, pick one
gst=find(round(Tlat)==SL);
gst=gst(1);

Tplot0=Ta_ann(gst,1:2:720);         % convert to same resolution
Eplot0=topoE(gse,:);
goodies2=find(Eplot0>geoidE(gse,:));          % above sea level
plot(Tplot0(goodies2), Eplot0(goodies2), 'r.')
```

The following script section plots an individual month's relationship between air temperature and elevation using the monthly temperature data array Ta. The title is constructed as a concatenation of character strings which allows you to include month number in the figure title. The Matlab function **int2str** converts an integer number, in this case 1, to a character.

The methods of identifying and using subsets of a larger data set presented in this section could also be used to examine data within specified elevation ranges or combinations of elevation and latitude, and so on.

```
%* plot one month
pT=1;         % select month number
Tplot=Ta(1:2:360,1:2:720,pT);

figure(101)
clf
plot(Tplot(goodies), topoE(goodies), 'b.')
hold on
title(['mean monthly atmospheric temperature lapse rate month: ' int2str(pT)])
xlabel('temperature (C)')
ylabel('altitude (m)')
```

1.4 a function to render flat maps

```
function f_map(FN, mdat, mlat, mlon, Cr, ttl, xlb, ylb)

%* render flat map data  f_map(FN, mdat, mlat, mlon, Cr, ttl, xlb, ylb)
%   FN       figure number
%   mdat     map data to be rendered
%   mlat     latitudes for mdat
%   mlon     longitudes for mdat
%   Cr       color axis [min max] array
%   ttl      figure title, expects character string
%   xlb      x-axis label, character string
```

```
%    ylb       y-axis label, character string

%* load a MATLAB native data set containing lat & lon of world coaslines
load coast

%* map plot window
wind=[min(mlon) max(mlon) min(mlat) max(mlat)];

%********
% flat map using data passed to the function
%********

figure(FN)                          %open figure window
clf
axis equal
axis(wind)                          % define axes for this plot
hold on
caxis(Cr)                           % define color range for this plot
pcolor(mlon, mlat, mdat)            % simple Plate Carree projection
shading flat
xlabel(xlb)
ylabel(ylb)
title(ttl)
colorbar('horiz')
hold on
plot(long, lat, 'k-')               % plot world coastlines
```

You need to define all of the variables to be passed to f_map. This can be done at the command line in the command window or you can collect all the commands together in a script that "drives" f_map. For example,

```
>> mapM=1;
>> ts='1961 to 1990 mean January precipitation (mm/day)';
>> FN=10;
>> Cax=[0 20];
>> xls='longitude';
>> yls='latitude';
>> f_map(FN, precip(:,:,mapM), plat, plon, Cax, ts, xls, yls)
```

Some additional color range options:
whole data set:

```
Cax=[min(min(min(precip))) max(max(max(precip)))];
```

just one month, where mapM has been defined:

```
Cax=[min(min(precip(:,:,mapM))) max(max(precip(:,:,mapM)))];
```

1.5 Questions

Please use the MATLAB *programs available at the course website to answer the following questions. In some cases, there is more than one way to answer the question using the data available for this lab. If the question does not specify a data set, describe what data you have used in answering each question. Short paragraphs are sufficient. You may wish to print a coastline map to use as a reference but please do not print all the colormaps. You may email electronic documents or hand in paper copies.*

1. Where were the largest changes in glacier ice between the last glacial maximum and the present-day?

2. Which geographic regions experience the largest and smallest temperature ranges (seasonality) over the course of a year? Explain the underlying causes of the patterns you identify in the seasonal temperature variation.

3. Print the zonal mean land and ocean precipitation plot. Annotate the plot to identify regions of relatively high and relatively low sea level pressure. Explain your reasoning.

4. In which geographic regions does precipitation vary the most over the course of the year? There is more than one way to use the data provided with this lab to answer this question. Describe how you used the data to answer this question.

5. Use the monthly precipitation data set to compute the total annual rainfall at each grid point and map the result. Compute and map the mean total number of wet days per year. Compare (a) Northern Europe and (b) the Gulf and Atlantic coastal plain of North America using these two values (that is, describe and explain the relationship between total precipitation and total wet days).

GEOL 559 students

6. The minimum differences between the LGM reconstruction and the present-day Earth surface elevation are negative. What does this mean and where does it happen? What process(es) are responsible for these changes in surface height?

7. The zonal mean precipitation plots contain a lot of interesting detail. For example, seasonal maxima are not symmetric across the equator, particularly in the land+ocean case. Propose a reason for that asymmetry. How might you test your hypothesis?

8. Suppose that summer melting and winter precipitation are the most important variables controlling the growth of ice sheets on land. Using the temperature and precipitation data provided in this lab, predict where the largest ice sheets should have existed at the last glacial maximum. Explain the reasoning for your prediction. Test your hypothesis using other data provided with this lab.

9. Make a plot of land surface air temperature vs. elevation following the directions in section 1.3.5. Describe and explain the dominant trends you see in the plot.