University of Colorado
Department of Computer Science

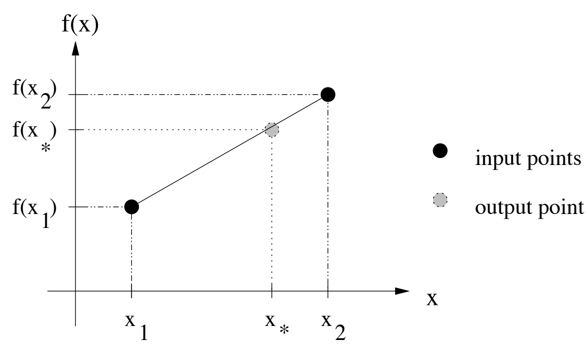
Numerical Computation CSCI 3656

Spring 2016

Problem Set 6 Solutions

1. In your favorite programming language, implement a program that:

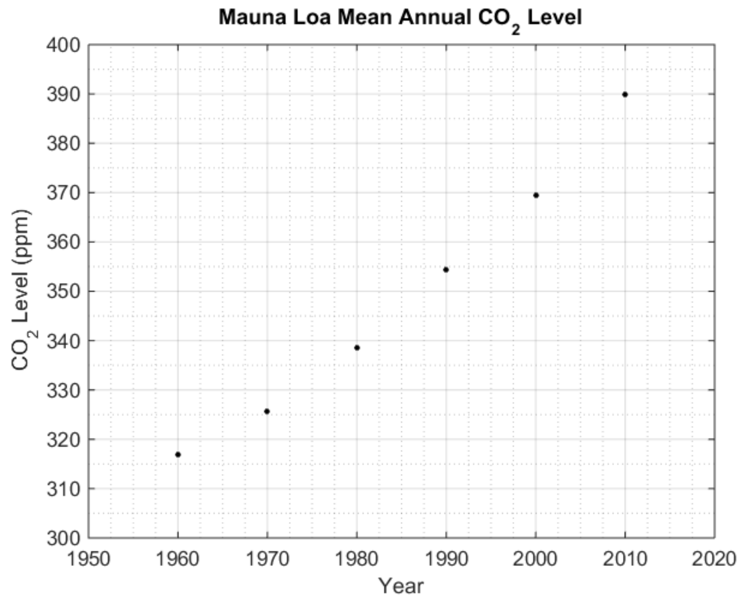
   - takes as input two points — $(x_1, f(x_1))$ and $(x_2, f(x_2))$ — and another x value x_* that is between x_1 and x_2, and
   - computes $f(x_*)$ using simple linear interpolation, as shown in the picture below:



(a) *[5 pts]* Dig around on the web and find some interesting **decadal** census data (e.g., of India, of your home city, of stray dogs in London, ...) that covers at least the last 50 years and preferably the last century. Turn in a plot of your data and a link to the webpage where you found it. Don't forget to label your axes.

The following data show the mean annual CO_2 levels measured at the top of Mauna Loa at 10 year intervals. (Yes, I know that's not population data, but the ideas are the same and this data set has an interesting property that gets a few important points across, one mathematical and one climatological.)

year	CO_2 level (ppm)
1960	316.91
1970	325.68
1980	338.68
1990	354.35
2000	369.52
2010	389.85



Here is a matlab function for linear interpolation (and extrapolation).

```
function y = lin_interp(x1, fx1, x2, fx2, xstar)
% Solve f(x*) using linear interpolation (or extrapolation)
% (x1, fx1) and (x2, fx2) are points on the function of interest
% x* is the input for which f(x) is desired
% y is the solution to the linear interpolation problem
y = (xstar-x1)*(fx2-fx1)/(x2-x1) + fx1;
```

(b) *[15 pts]* Use your interpolation code to estimate the value of your chosen quantity halfway through the most recent decade. (That is, say you know the population of Cleveland in 2000 and 2010. What was that population in 2005?) If you have access to yearly data, compare your interpolated result to the true population.

Using the `lin_interp` function on the 2000 and 2010 values, I estimated that the mean CO_2 level in 2005 was around 379.885 ppm.

2. *[10 pts]* The same idea can be applied in order to *extrapolate*: that is, to find a value for $f(x_*)$ if x_* is *not* between x_1 and x_2. Using the same data and the same code as in problem 1, estimate the size of your chosen population in 2050. This will require making an intelligent choice about what two points to feed your program, and perhaps removing any "between" error checks that you may have incorporated.

   Using the same function and the last two points, I estimated that the average CO_2 level in 2015 was around 400.015 ppm. The true value was 400.83, so the extrapolated value was 0.2% too low. In other words, the CO_2 level is increasing faster than linearly. If I extend the estimation process out to 2050, I get 471.17.

3. *[10 pts]* Use the Lagrangian interpolating polynomial approach to fit a parabola to the last three points in your data. See pp. 140 for the formula.
$P_2(x) = 354.35\frac{(x-2000)(x-2010)}{(1990-2000)(1990-2010)} + 369.52\frac{(x-1990)(x-2010)}{(2000-1990)(2000-2010)} + 389.85\frac{(x-1990)(x-2000)}{(2010-1990)(2010-2000)}$

Use that polynomial to estimate the size of your population halfway through the last decade. This polynomial yields an estimate of 379.04 ppm in 2005. Compare that to the value that you got in problem 1. Which of the two interpolated values is better, do you think? Would that depend on the shape of the data?

Generally, the (parabolic) Lagrangian polynomial will be better than a simple linear interpolation, but not in this case. (I know that because I know the true value at 2005.) This is because of the shape of the data, which is a really good match to a plain old line. Probably your population data weren't linear, in which case your Lagrangian polynomial produced a better answer.