UCSB, CS111, Midterm Exam, Feb. 7-14, 2006

This is a one-week long project, due in class on Tuesday, Feb. 14. If you can not make it to that class, put your work in the CS111 homework box in the CS office, or email it to `moler@cs.ucsb.edu`, before 3:00 pm, Feb. 14.

Do your own work.

This project was motivated by an article headlined *"Come October, Baby Will Make 300 Million or So"* that appeared the New York Times on January 13. A copy of the article is available at:

http://www.cs.ucsb.edu/~moler/NYT_Census.doc

Your ultimate goal is to produce a more detailed version of the first plot in the graphic that accompanied the article:

http://www.cs.ucsb.edu/~moler/NYT_Census.jpg

The U. S. Census Bureau has a "Population Clock" on their Web home page:

http://www.census.gov

On Feb. 7, 2006, the clock claims the United States population (in millions of people) is about :

298.062

This project considers how one might compute the numbers in the NYT article and the Population Clock. The population data for the project is the same as that is used in `censusgui.m`:

```
p =
   75.995
   91.972
  105.711
  123.203
  131.669
  150.697
  179.323
  203.212
  226.505
  249.633
  281.422
```

The MATLAB date functions `datenum` and `datestr` work with time measured in units of days, rather than years. The Census Bureau has decided that their data corresponds to April 1 of the decennial years, so the time variable t can be generated by

```
t = datenum((1900:10:2000)',4,1)
```

The corresponding centered and scaled time variable s is

```
s = (t - mean(t))/std(t)
```

We can fit the census data by any of the following five models, each of which is a function of s or t:

spline	Use `splinetx(t,p,...)`
"pchip"	Use `pchiptx(t,p,...)`
quadratic	$p \approx c_1 s^2 + c_2 s + c_3$
cubic	$p \approx c_1 s^3 + c_2 s^2 + c_3 s + c_4$
exponential	$p \approx \alpha e^{\beta t}$

The first two models interpolate the data, as described in chapter 3. The last three models are least squares approximations to the data, as described in chapter 5. The exponential model can be transformed to a model that is linear in its coefficients by taking logarithms,

$$\log p \approx c_1 t + c_2, \text{where } c_1 = \beta, c_2 = \log \alpha$$

(Part 1) Use each of the five models to extrapolate to February 7, 2006, that is

```
T = datenum(2006,2,7)
```

Find the resulting values of the population. Which one is closest to the Population Clock estimate? Figures 1 and 2 illustrate this computation. You do not have to reproduce these figures, you should just compute the p-axis coordinates of the five black dots.

(Part 2) Some, but not all, of the models use the MATLAB backslash operator to solve, or approximately solve, square or rectangular systems of simultaneous linear equations. Find the size and condition number of each of the coefficient matrices.

(Part 3) Which one of the models does not make use of all the data when extrapolating past the year 2000? How much of the data does it use?

(Part 4) At what time do each of the models predict the population will reach 300 million people? There are two ways to approach this problem, direct interpolation and reverse interpolation. They are described briefly in section 4.9 of the text, but we will not use `fzerotx`. Direct interpolation determinines t by solving the equation

$$p(t) = 300$$

Each of the models involves a polynomial or piecewise polynomial. Once you have found the coefficients of the relevant polynomial, use the MATLAB function `roots` to solve equations like

$$c_1 s^3 + c_2 s^2 + c_3 s + c_4 - 300 = 0$$

Figures 3 and 4 illustrate direct interpolation. Again, you do not have to reproduce these figures, you should just compute the t-axis coordinates of the five black dots.

(Part 5) Reverse interpolation involves interchanging the roles of t and p. The five models are

spline	Use `splinetx(p,t,...)`
"pchip"	Use `pchiptx(p,t,...)`
quadratic	$t \approx c_1 p^2 + c_2 s + c_3$
cubic	$t \approx c_1 p^3 + c_2 p^2 + c_3 p + c_4$
logarithmic	$t \approx c_1 \log p + c_2$

Once you have found the coefficients in each model, you simply evaluate the model at $p = 300$. Root finders such as `fzero` or `roots` are not required. Figures 5 and 6 illustrate reverse interpolation.

(Part 6) Make your own version of the first plot in the NYT graphic. Choose what you think is the best model and use either direct or reverse interpolation. Make a MATLAB plot that shows the time when the population reaches 300 million. You do not have to try to reproduce the shading and other details of the NYT graphic.

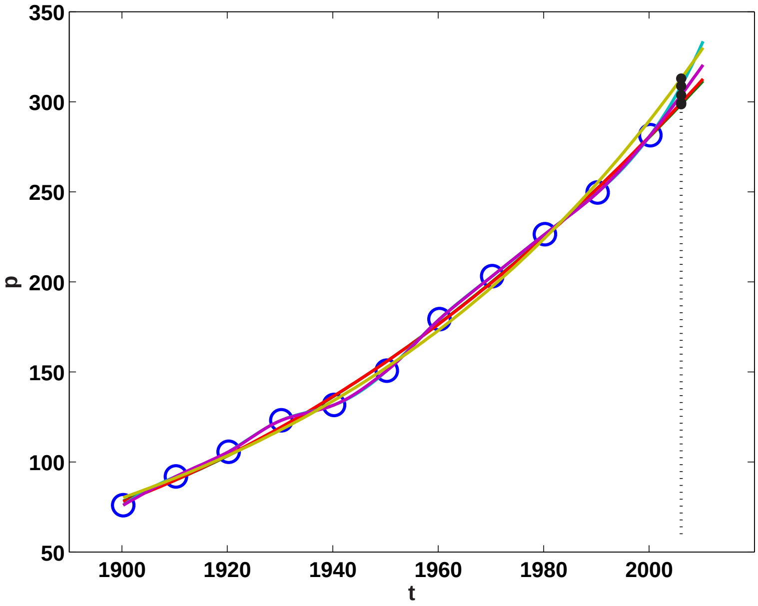

Figure 1: Extrapolation

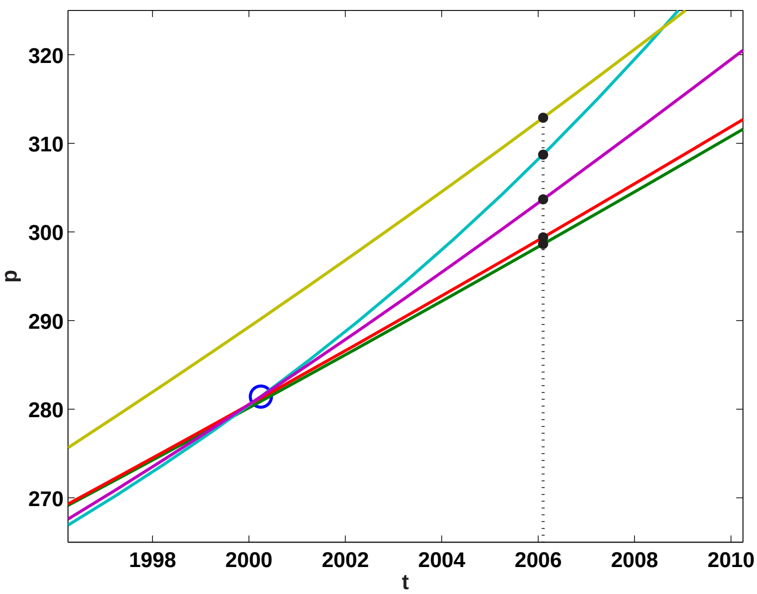

Figure 2: Extrapolation, detail

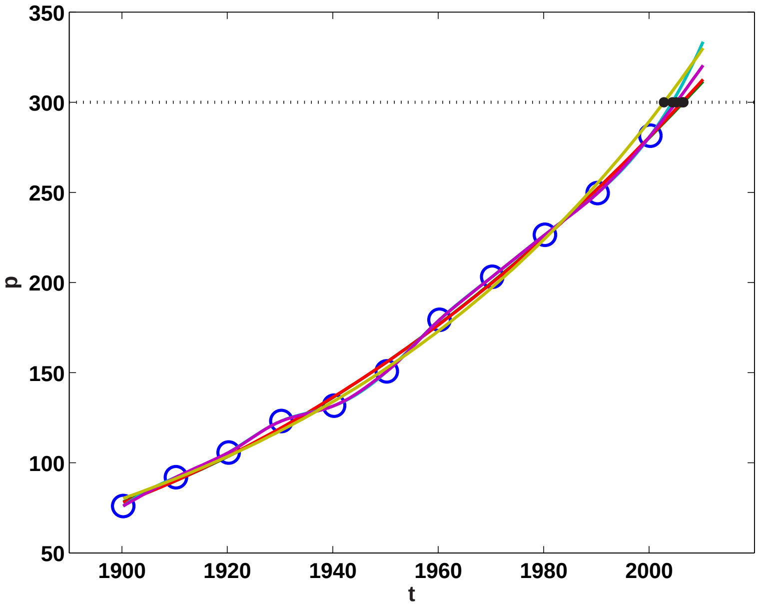

Figure 3: Target population, direct interpolation

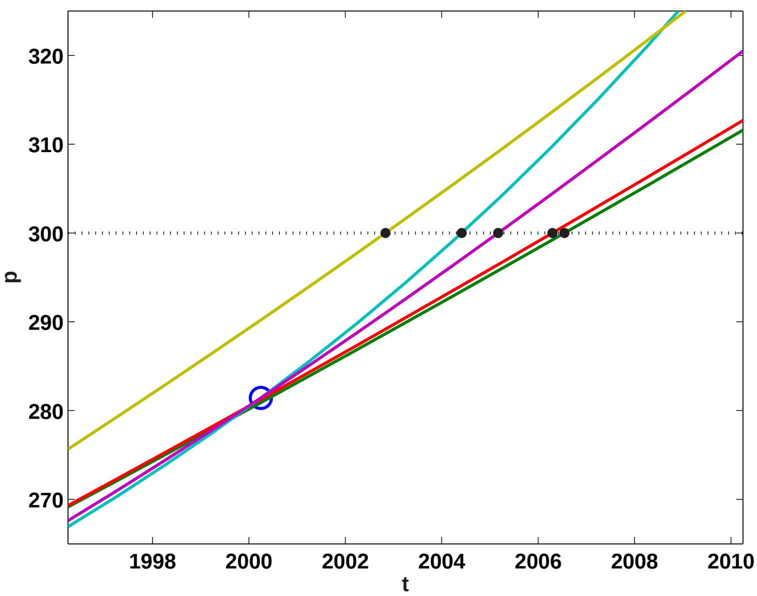

Figure 4: Direct interpolation, detail

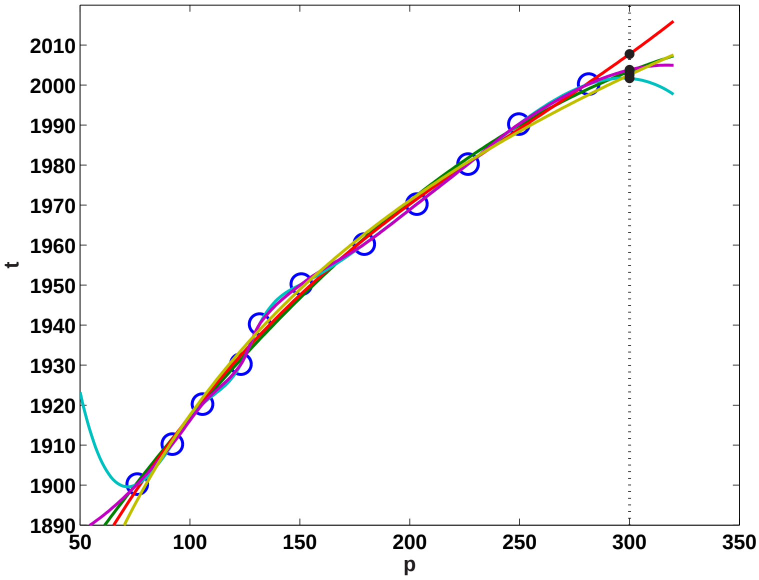

Figure 5: Target population, reverse interpolation

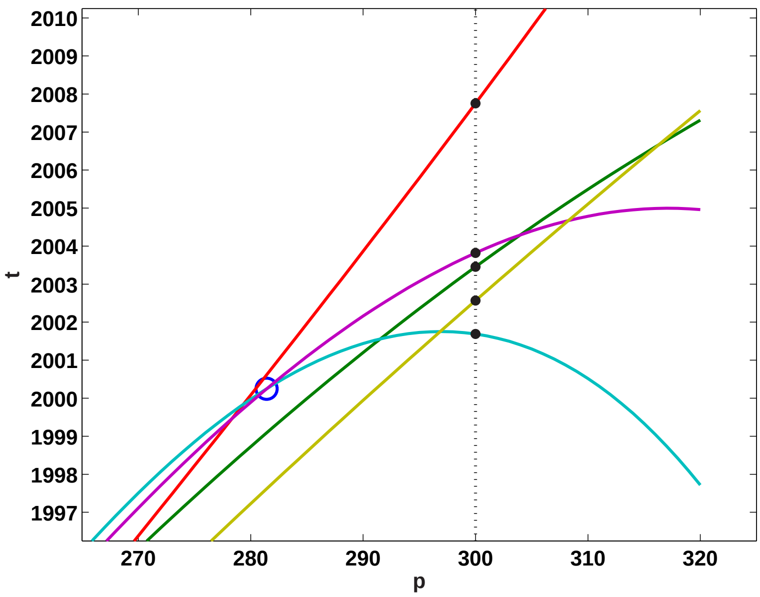

Figure 6: Reverse interpolation, detail