Astronomy 575: Project $^{\#}2$ (Monte Carlo Simulations)

Week 1:
Create a new managed C/C++ project within the Eclipse IDE. Within the project, write a C program (mcs01.c) which generates sets of random numbers based on the following algorithm. This is a mixed linear-congruential generator (LCG), and, like all random number generators, will create pseudo-random numbers. For various values of multiplier a, additive constant b, modulus m, and seed value r_0,

$$r_i = (a \times r_{i-1} + b) \bmod m.$$

where a and b are non-negative, m is a positive integer, and the n ($n \leq m$) values of r_i are integers between 0 and $m - 1$.

Examine the sequences of random numbers generated in the following cases. For each case, determine the maximum number of unique elements which can be generated, and the effect of changing the seed value. Explore both periodicity and serial correlation, taking into account how randomly distributed the values of r_i are and how correlated successive values may be. You will, for example, want to print them out in binary notation to look for patterns. You should also plot successive values against each other (r_i versus r_{i+1}, for the $n - 1$ overlapping and the $n/2$ non-overlapping values), and create histograms showing the distribution of values over fractions of a full period.

Case 1: $a = 25,173$, $b = 13849$, $m = 2^{16}$, and $r_0 = 1$.
Case 2: $a = 1,229$, $b = 1$, $m = 2^{11}$, and $r_0 = 1$.
Case 3: $a = 65$, $b = 1$, $m = 2^{11}$, and $r_0 = 1$.

Here is a C function which you may use to convert decimal to binary notation.

```
/* Dec2Bin: convert decimal value to binary notation */
/* For example, 5 => 101, and 44 => 101100           */

void Dec2Bin(int decimalval, char *stringval, int strsize)
{
  int i;

  for (i = strsize-1; i >= 0; --i)  {
    stringval[i] = (decimalval & 0x00000001) ? '1' : '0';
    decimalval >>= 1;
  }
  stringval[strsize] = 0x00;

  return;
}
```

Analyze the results of the random number generators defined above. What are the particular weaknesses of each one, and what are their strengths? Discuss your results in a LaTex project report which incorporates key Supermongo plots to aid in your discussion. Please be explicit about which Additional Tasks you have explored, and your conclusions. When your project report is complete, attach a tarfile containing your C source file (mcs01.c) and output files (mcs_01.out through mcs_03.out), your Makefile, your Supermongo macro file (mcs01.mac), and your project report (report01.pdf) to an e-mail (Subject: Project02-Week 1 Materials from Your Name), and submit it.

Week 2:
Utilize the LCG of your choice to model the effects of Malmquist-type Bias on local galaxy samples limited by magnitude and volume cuts. Begin by using a uniform distribution to randomly place galaxies in x, y, and z throughout a cubic volume 200 Mpc on each side. (Be sure to use different seeds when computing x, y, and z values, so that your coordinate components are not correlated.) If you assume a local density

function of 0.25 bright galaxies per cubic Mpc, roughly how many random elements do you need? Derive equilateral coordinates (right ascension in hours, declination in degrees, and also radial distance in Mpc) for each galaxy. Next, truncate the sample by removing objects more than 100 Mpc from the centerpoint of the distribution. Finally, test that your spatial distribution is correct by plotting histograms of declination, right ascension, and distance with justified model curves of the expected distributions superimposed, for various numbers of initial elements (for densities less than or equal to the rough density estimate).

Now generate model luminosities for the sample. Create two sets of uniformly distributed random numbers, ranging between -1 and 1. Then use a Box-Muller transform to create Gaussian distributions from them. For n-dimensional uniform arrays u_1 and u_2, we can thus form m-dimensional (where $m \leq n$) normally distributed arrays g_1 and g_2.

$$r = u_1^2 + u_2^2$$

$$\begin{aligned} v_0 &= r \quad \texttt{if } (0 < r \ \&\& \ r < 1) \\ v_1 &= u_1 \ \texttt{if } (0 < r \ \&\& \ r < 1) \\ v_2 &= u_2 \ \texttt{if } (0 < r \ \&\& \ r < 1) \end{aligned}$$

$$v_3 = \sqrt{\frac{-2 \ln(v_0)}{v_0}}$$

$$\begin{aligned} g_1 &= v_1 \times v_3 \\ g_2 &= v_2 \times v_3 \end{aligned}$$

Scale the Gaussian arrays to a mean value of -20.0 magnitudes, with a standard deviation of 1.5, to form absolute magnitudes for a sample of bright spiral galaxies. You can then determine the apparent magnitude of each galaxy as viewed from Earth, by combining its intrinsic luminosity and distance.

Construct a set of three "observational" samples by imposing magnitude limits of 12, 14, and 16 magnitudes upon the initial data set. Construct and overlay histograms of key galaxy properties. Calculate the observed average luminosity, the standard deviation of the luminosity, and the average distance to the observed galaxies, for each of the four data sets. How complete are your observed samples for galaxies at a distance of, for example, the Virgo cluster versus the Cancer cluster?

Note that we have chosen to explore a very local volume and to apply quite stringent sensitivity limits. This allows us to avoid two distance-dependent issues. First, at cosmological distances Newtonian calculations of distance and magnitude are no longer applicable. Second, only in the local Universe can we assume that there are no evolutionary changes in galaxy properties across our survey volume. This allows us to concentrate upon the effect of survey sensitivity on the observed, versus complete, samples.

Generate a set of velocity widths V_{full} for the sample galaxies, assuming

$$M = m \,(\lg V_{full} - 2.5) + b,$$

where $m = -7.27$ and $b = -20$ (V_{full} is simply twice the circular velocity V_{circ}). Then generate revised absolute magnitudes M' by adding a normally distributed noise component σ with a standard deviation of 0.3 magnitudes.

$$M' = m \,(\lg V_{full} - 2.5) + b + \sigma,$$

Examine the distribution of $\Delta M \equiv M - M'$ versus distance and absolute magnitude, and then re-apply the magnitude limits to recreate the three observational data sets, based upon revised observed magnitudes.

Now perform y on x and x on y least squares fits on the four data sets, and compare your extracted slopes and y-intercepts, and associated errors, with the underlying relation used to generate V_{full} values. What do you conclude about techniques for fitting relations on magnitude limited versus volume limited samples?

Discuss your results in a LaTeX project report which incorporates key SUPERMONGO plots to aid in your discussion. Please be explicit about which Additional Tasks you have explored, and your conclusions. When your project report is complete, attach a tarfile containing your C source file (mcs02.c) and output files, your MAKEFILE, your SUPERMONGO macro file (mcs02.mac), and your project report (report02.pdf) to an e-mail (SUBJECT: PROJECT02-WEEK 2 MATERIALS FROM YOUR NAME), and submit it.

Week 3:
Model the stellar populations which make up a galaxy spectrum by combining a set of spectral templates drawn from optical spectra of Main Sequence and Giant stars of various spectral classes (Silva & Cornell 1992, ApJS, 81, 865).

Write a C program (mcs03.c) which reads in a set of spectral templates found on the ASTR575 class web pages (stellar_spectra.dat) and a spectrum of the galaxy NGC 3512 (ngc3512.dat). You will scale each spectral template independently, to create a combined best-fit model spectrum which matches that of the galaxy. As we have far more luminosity data points than spectral templates, the system is over-constrained. We can thus use the Moore-Penrose inversion technique to derive a least squares solution, as sketched below.

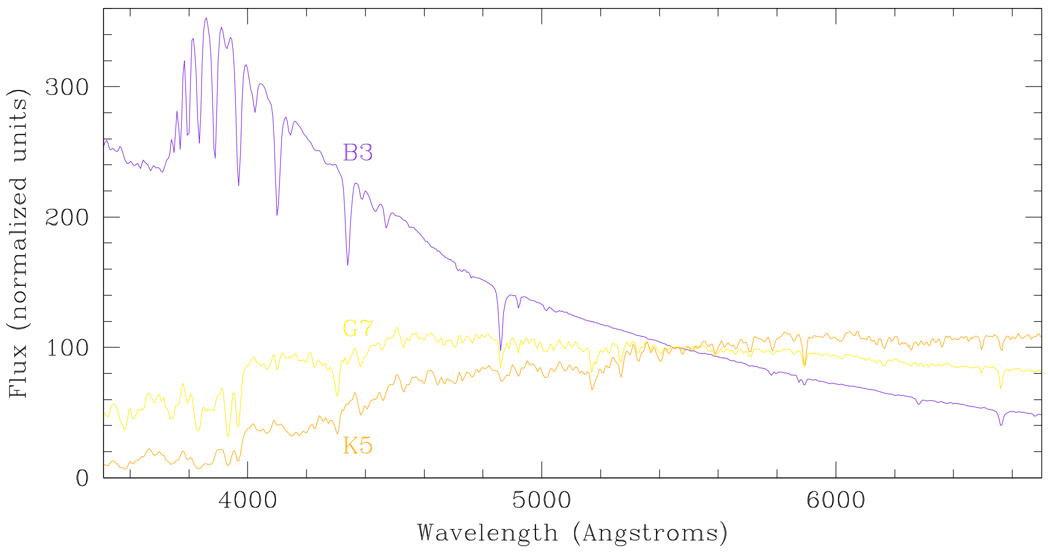


Figure 1: Three of the ten Silva & Cornell spectral templates, covering a range of spectral classes.

For a galaxy spectrum F_λ, we can fit the scalar amplitudes s_j to the stellar templates T_j^λ as follows. For the sake of example, we will utilize a restricted set of three templates: B3-4, G6-8, K5. The selected templates are shown in Figure 1; note that they are all arbitrarily scaled to the same amplitude at $5445\AA$. To achieve a best-fit match,

$$F_\lambda = \sum_{j=0}^{2} s_j T_j^\lambda$$

where F is a 639×1 matrix of galaxy fluxes with elements

$$F = \begin{bmatrix} 61.327 \\ 62.303 \\ 62.428 \\ \dots \\ 94.632 \end{bmatrix}$$

and 3×1 matrix s has the form

$$s = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix}$$

while for dwarf spectral templates B3-4, G6-8, and K5, 639×3 matrix T of stellar template fluxes can be written as

$$T = \begin{bmatrix} 253.810 & 49.472 & 9.367 \\ 257.281 & 48.610 & 9.879 \\ 258.936 & 47.363 & 9.643 \\ \dots & \dots & \dots \\ 48.316 & 81.007 & 107.513 \end{bmatrix}.$$

We can solve this in the following form

$$s = \left(T^t\,T\right)^{-1}\,T^t\,F,$$

where $\left(T^t\,T\right)^{-1}$ is the Moore-Penrose pseudoinverse of T and T^t is the transpose of T, with size 3×639 and content

$$T^t = \begin{bmatrix} 253.810 & 257.281 & 258.936 & \dots & 48.316 \\ 49.472 & 48.610 & 47.363 & \dots & 81.007 \\ 9.367 & 9.879 & 9.643 & \dots & 107.513 \end{bmatrix}.$$

The quantity $(T^t\,T)$ is a 3×3 matrix of the form

$$T^t\,T = \begin{bmatrix} 253.810 & 257.281 & 258.936 & \dots & 48.316 \\ 49.472 & 48.610 & 47.363 & \dots & 81.007 \\ 9.367 & 9.879 & 9.643 & \dots & 107.513 \end{bmatrix} \begin{bmatrix} 253.810 & 49.472 & 9.367 \\ 257.281 & 48.610 & 9.879 \\ 258.936 & 47.363 & 9.643 \\ \dots & \dots & \dots \\ 48.316 & 81.007 & 107.513 \end{bmatrix}$$

$$= \begin{bmatrix} 1.87241 \times 10^7 & 8.03748 \times 10^6 & 5.50450 \times 10^6 \\ 8.03748 \times 10^6 & 5.21978 \times 10^6 & 4.52050 \times 10^6 \\ 5.50450 \times 10^6 & 4.52050 \times 10^6 & 4.34237 \times 10^6 \end{bmatrix}$$

where individual elements have the form

$$\left(T^t\,T\right)[i,j] = \sum_{k=0}^{638} T^t[k,j] \times T[i,k].$$

Note that the first index listed here in brackets is the column number and the second is the row number, so that element $T[2,3]$ is located in the third column of the fourth row of T (with rows and columns both starting at zero, not one), not the third row of the fourth column.

We can invert this matrix to find

$$\left(T^t\,T\right)^{-1} = \begin{bmatrix} 7.20930 \times 10^{-7} & -3.23702 \times 10^{-6} & 2.45594 \times 10^{-6} \\ -3.23702 \times 10^{-6} & 1.64806 \times 10^{-5} & -1.30533 \times 10^{-5} \\ 2.45594 \times 10^{-6} & -1.30533 \times 10^{-5} & 1.07058 \times 10^{-5} \end{bmatrix},$$

using the following pseudo-code algorithm to perform the inversion in place.

```
for $i$ = 0, 2 {
   $r0$ = $(T^t\,T)\,[i,i]$
   $(T^t\,T)\,[i,i]$ = 1
   for $j$ = 0, 2 {
      $(T^t\,T)\,[i,j]$ = $(T^t\,T)\,[i,j]$ / $r0$
   }
```

```
      for $j$ = 0, 2 {
        if ($i \neq j$) {
          $r0$ = $(T^t T)[j,i]$
          $(T^t T)[j,i]$ = 0
          for $k$ = 0, 2 {
            $(T^t T)[j,k]$ = $(T^t T)[j,k]$ - $r0$ $\times$ $(T^t T)[i,k]$
          }
        }
      }
    }
```

The next step is to multiply T^t by F:

$$T^t F = \begin{bmatrix} 253.810 & 257.281 & 258.936 & \ldots & 48.316 \\ 49.472 & 48.610 & 47.363 & \ldots & 81.007 \\ 9.367 & 9.879 & 9.643 & \ldots & 107.513 \end{bmatrix} \cdot \begin{bmatrix} 61.327 \\ 62.303 \\ 62.428 \\ \ldots \\ 94.632 \end{bmatrix}$$

$$= \begin{bmatrix} 8.12358 \times 10^6 \\ 5.24518 \times 10^6 \\ 4.59580 \times 10^6 \end{bmatrix}.$$

Combining these two products, we observe that

$$\left(T^t T\right)^{-1} T^t F = \begin{bmatrix} 7.20930 \times 10^{-7} & -3.23702 \times 10^{-6} & 2.45594 \times 10^{-6} \\ -3.23702 \times 10^{-6} & 1.64806 \times 10^{-5} & -1.30533 \times 10^{-5} \\ 2.45594 \times 10^{-6} & -1.30533 \times 10^{-5} & 1.07058 \times 10^{-5} \end{bmatrix} \begin{bmatrix} 8.12358 \times 10^6 \\ 5.24518 \times 10^6 \\ 4.59580 \times 10^6 \end{bmatrix}$$

$$.$$

$$= \begin{bmatrix} 0.164770 \\ 0.157051 \\ 0.685993 \end{bmatrix} \equiv s.$$

Figure 2 shows the results of our fit, building a green summed spectral template which is 16% B stars, 16% G stars, and 68% K stars. Recall, however, that these percentages are luminosity-weighted, not number-weighted.

We can calculate a reduced χ^2 value for our fit, as shown in Figure 3. In addition, we can examine the wavelengths of the residuals of the fit to determine the primary sources of error.

A pleasant online review of the relevant linear algebra can be found at

`www.matrixanalysis.com.`

Re-conduct the fit with the three spectral templates above, to verify the accuracy of your matrix manipulation algorithms. Then explore the effect of using various subsets (and all) of the stellar templates. How does the use of Giant stars compare to the Main Sequence components? Are there any non-physical repercussions to our choice of analysis technique?

Are there particular galaxy features which are consistently poorly, or well, fit by the stellar templates? Do your fits, and the derived reduced χ^2 values, make intuitive sense, based on your knowledge of stellar populations? Discuss roughly extrapolated number-weighted counts of various populations, based on the luminosity-weighted amplitudes derived.

Utilize the LCG of your choice to add small amounts of uniformly distributed noise to the spectrum for NGC 3512, and then test your fitting technique for sensitivity to help understand the drivers of the variations in template amplitude.

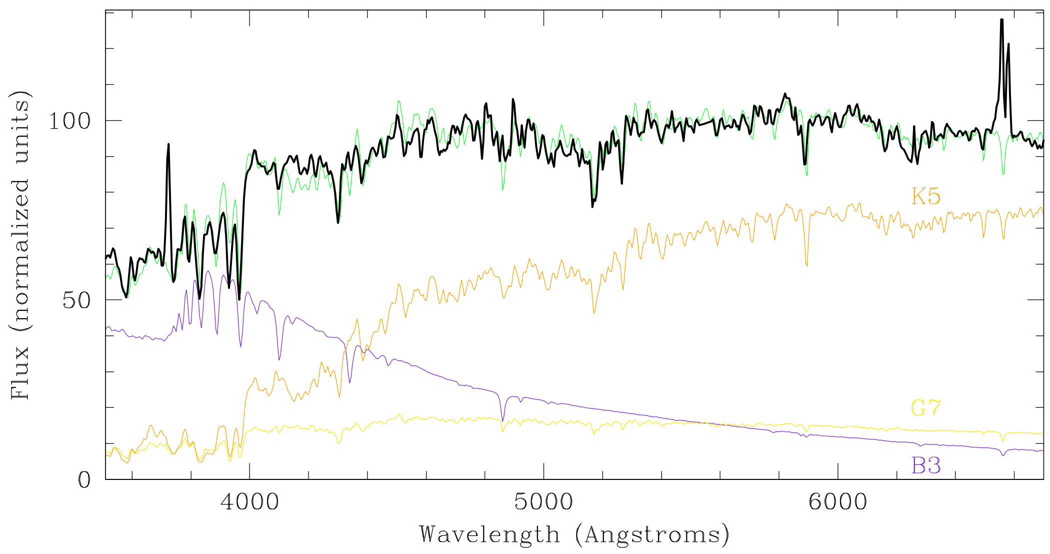


Figure 2: The three spectral templates, scaled in amplitude according to our fit. The black spectrum is of NGC 3512, while the green represents the sum contribution of our three components.

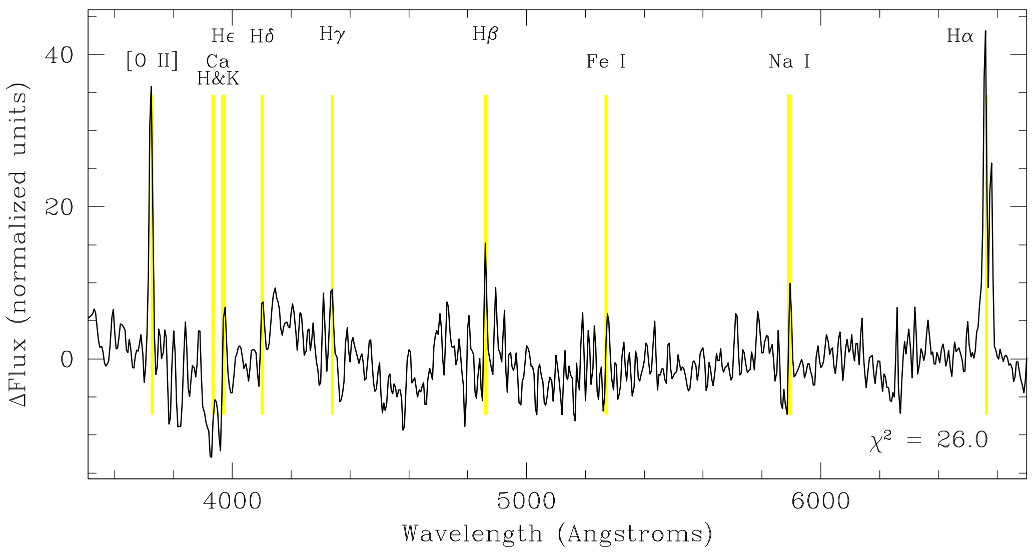


Figure 3: The residuals of the model fit to the galaxy spectrum for NGC 3512, as a function of wavelength. Selected optical features are marked in yellow.

Discuss your results in a LaTeX project report which incorporates key SUPERMONGO plots (like those shown here) to aid in your discussion. Please be explicit about which Additional Tasks you have explored, and your conclusions. When your project report is complete, attach a tarfile containing your C source file (mcs03.c) and output files, your MAKEFILE, your SUPERMONGO macro file (mcs03.mac), and your project report (report03.pdf) to an e-mail (SUBJECT: PROJECT02-WEEK 3 MATERIALS FROM YOUR NAME), and submit it.

Additional Project Tasks (for Week 1 through Week 3):

1. Explore the capabilities and limitations of the random number generator module within C.

2. Explore the capabilities and limitations of the random number generator module within SUPERMONGO.

3. Explore the attributes of the following additional LCG:

   Case 4: $a = 7^5$, $b = 0$, $m = 2^{31} - 1$, and $r_0 = 1$.

   This problem explores the question of how to evaluate an LCG for which the product of a and r_{i-1} is greater than the largest integer allowed on the system. One solution is to use 64-bit and unsigned integers. Alternatively, one can utilize the following identity:

   $$a\,x \bmod m \;=\; g(x) \;+\; m\,h(x)$$

   where

   $$g(x) \;=\; a(x \bmod q) \;-\; r(x \operatorname{div} q)$$

   and

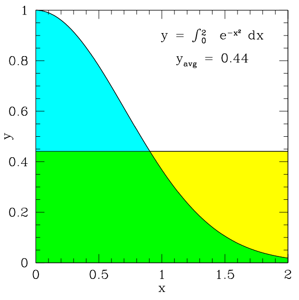
   $$h(x) \;=\; (x \operatorname{div} q) \;-\; (a\,x \operatorname{div} m),$$

   for $q = m \operatorname{div} a$ and $r \;=\; m \bmod a$.

4. Design a Monte Carlo simulation to perform numerical integration of the following function:

   $$y = \int_0^2 e^{-x^2}\,dx.$$

   Utilize the LCG of your choice to select uniformly distributed random numbers which sample the full range in x, and then calculate the corresponding y values and average them in order to then evaluate the integral. How large must the sample be before successive cumulative estimates converge to within 0.0001% (one part in a million)? For contrast, sample the full range in x linearly and compare how



   quickly the cumulative solution converges. Under what conditions would the Monte Carlo simulation be preferable?

5. Create a model sample of spiral galaxies placed at random inclination and position angles on the sky, and solve numerically for the average inclination angle expected in a large set of such objects.

6. Replace the Hubble-style Gaussian distribution of galaxy luminosities with the more modern Schechter luminosity function, and re-perform your Week 2 sample bias analysis. For galaxy luminosity L, where the number of galaxies with luminosities between L and $L + dL$ is $\phi(L)\,dL$,

$$\phi(L)\,dL \;=\; \phi^* \left(\frac{L}{L^*}\right)^{\alpha} \exp\left(-\frac{L}{L^*}\right)\frac{dL}{L^*}.$$

   Select appropriate scaling values ϕ^*, L^*, and α from the literature.

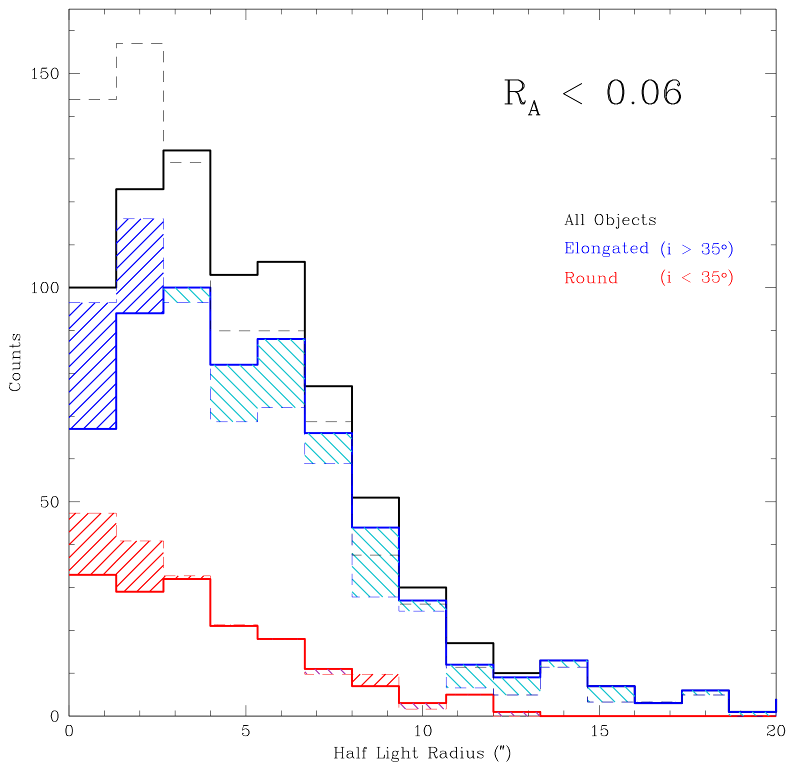
7. Explore the effect of using a single seed for all randomly generated sequences for the Week 2 model, or of using an inappropriate random number generator. What sort of spurious trends can be introduced into the derived data sets?

8. You should notice the absence of a key galaxy constituent when fitting stellar templates to the NGC 3512 spectrum. Use random uniform and normal distributions to create an additional template, with noise, to satisfy this unmet need appropriately.

9. Search the NED and SIMBAD astronomical databases to find additional optical galaxy spectra to fit with the stellar templates.

10. Explore using Tikhonov Regularization rather than the Moore-Penrose inversion to solve for stellar template amplitudes for NGC 3512.

11. Recreate the following histogram, comparing half light radii from real observations of distant galaxies with random values drawn systematically (with restrictions on galaxy asymmetry) from large imaging catalogs, via SUPERMONGO. Data point values can be found in the file sample02.dat.



Additional Project Tasks (reprise):

The following Additional Tasks from our first project are of good general utility. If you did not complete them previously, you may investigate them as a part of this project.

1. Add a time stamp to your output files indicating when they were created.
2. Replace a set of connected individual variables with a single STRUCT, so that they can all be passed to and from the MAIN program as a single argument.
3. Use MALLOC within your C program to dynamically allocate memory. You might either allocate memory in a series of blocks, or allocate the exact amount needed after first reading an input file once.
4. The basic MAKEFILE provided in this class is rather simplistic, because it recompiles the C source file whether or not it has been updated since the last compilation. Adapt the file so that it only runs `gcc` and `latex` when actually necessary.
5. Create a SUPERMONGO plot of data, including error bars, in which some points are solid and others are hollow. Be sure that the point centers for the hollow points are not obscured by associated error bars.