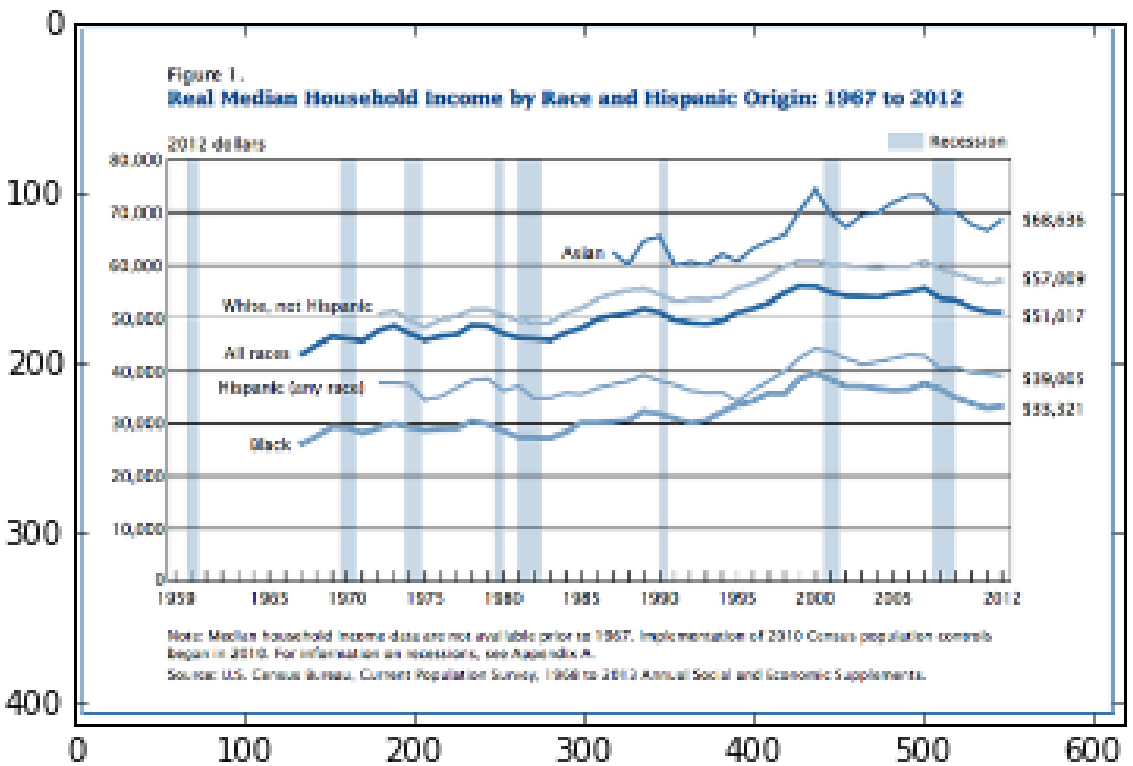
Simulating Income Distributions

Thomas M. Breuel

```
1 from pylab import *
2 from urllib2 import urlopen
3 from bisect import bisect
```

Income Distributions and Political Debates

```
1 imshow(imread(urlopen("http://imgick.cleveland.com/home/cleve-media
    /width620/img/datacentral/photo/13425050-mmmain.png")))
```



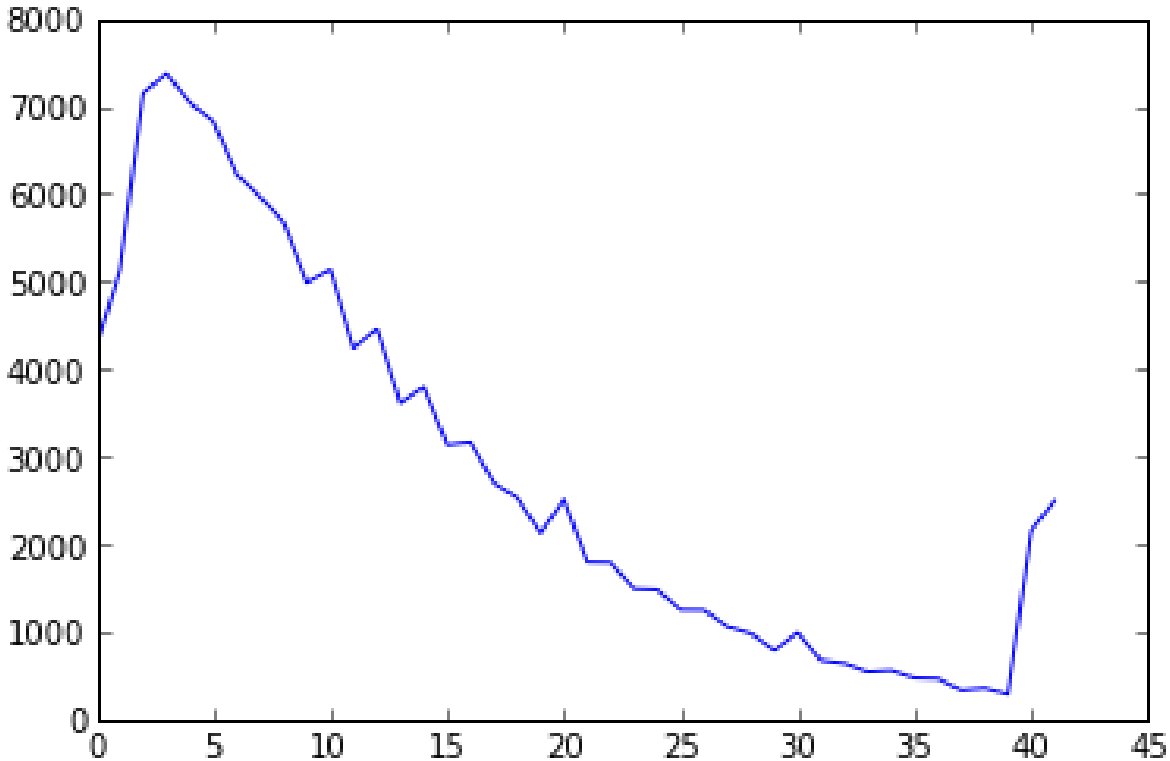
Reading the Income Distribution

```
1 data = genfromtxt("usincome.csv",skip_header=2,delimiter=",",names=
      "lo,hi,n,mean,stddev")
2 for i in range(5): print data[i]
```

```
(0.0, 4999.0, 4245.0, 1249.0, 50.0)
(5000.0, 9999.0, 5128.0, 7923.0, 30.0)
(10000.0, 14999.0, 7149.0, 12389.0, 28.0)
(15000.0, 19999.0, 7370.0, 17278.0, 26.0)
(20000.0, 24999.0, 7037.0, 22162.0, 27.0)
```

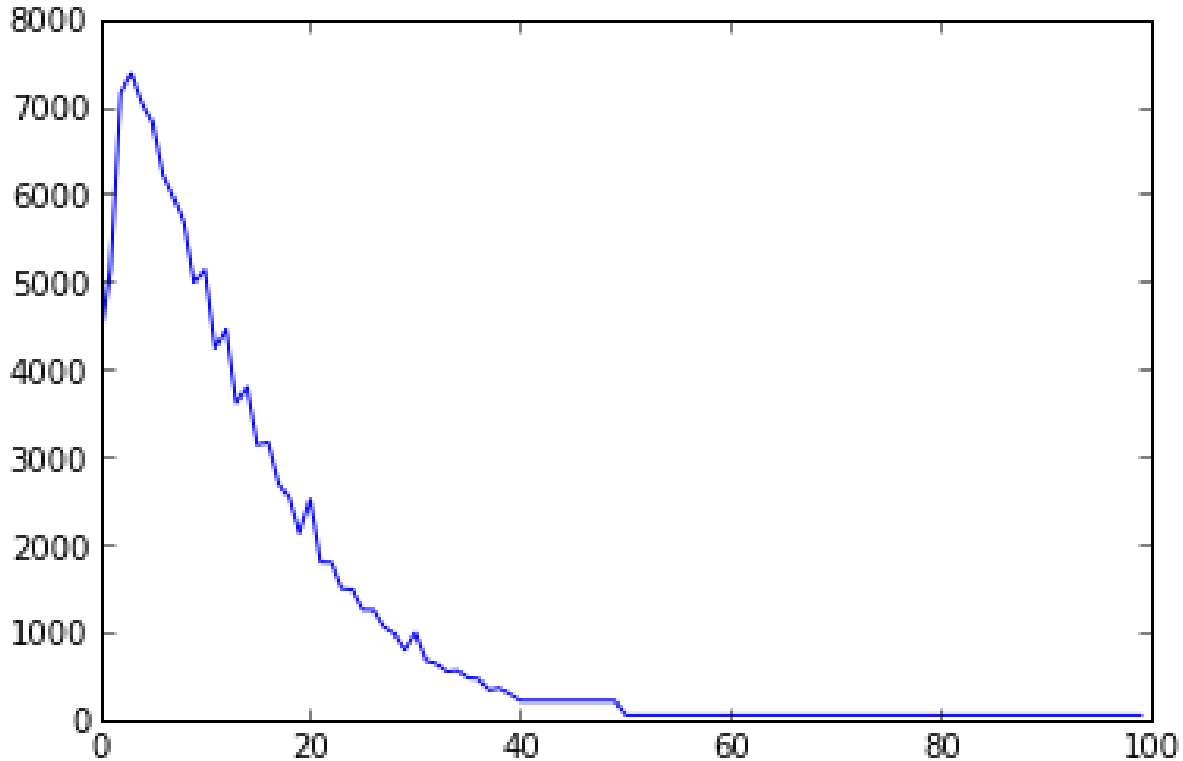
```
1 data = array([list(x) for x in data])
```

```
1 plot(data[:,2])
```


8000
7000
6000
5000
4000
3000
2000
1000
0
0
5
10
15
20
25
30
35
40
45

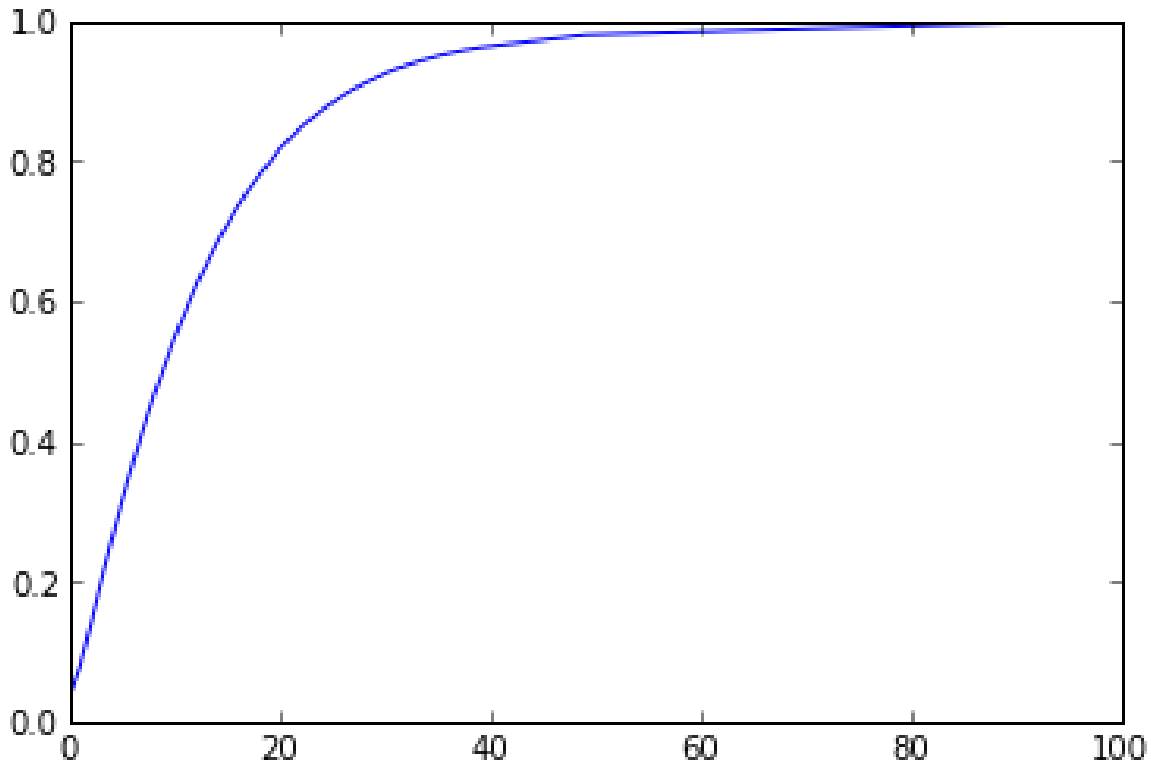
```
1 idist = concatenate([data[:-2,2],[data[-2,2]/10]*10,[data
      [-1,2]/50]*50])
```

```
1 plot(idist)
```



Random Samples from the Income Distribution

```
1 cdist = add.accumulate(idist)
2 cdist = cdist*1.0/amax(cdist)
3 plot(cdist)
```



```
1 bisect(cdist,rand())
```

```
4
```

```
def empsample(dist,n=1):
    """Given an array representing a histogram, return a random bin
        number
    according to that histogram."""
    cdist = add.accumulate(dist)
    cdist = cdist*1.0/amax(cdist)
    result = [bisect(cdist,rand()) for i in range(n)]
    if n==1: return result[0]
    else: return array(result,'f')
```

Let's generate a population of 100000 individuals and their incomes.

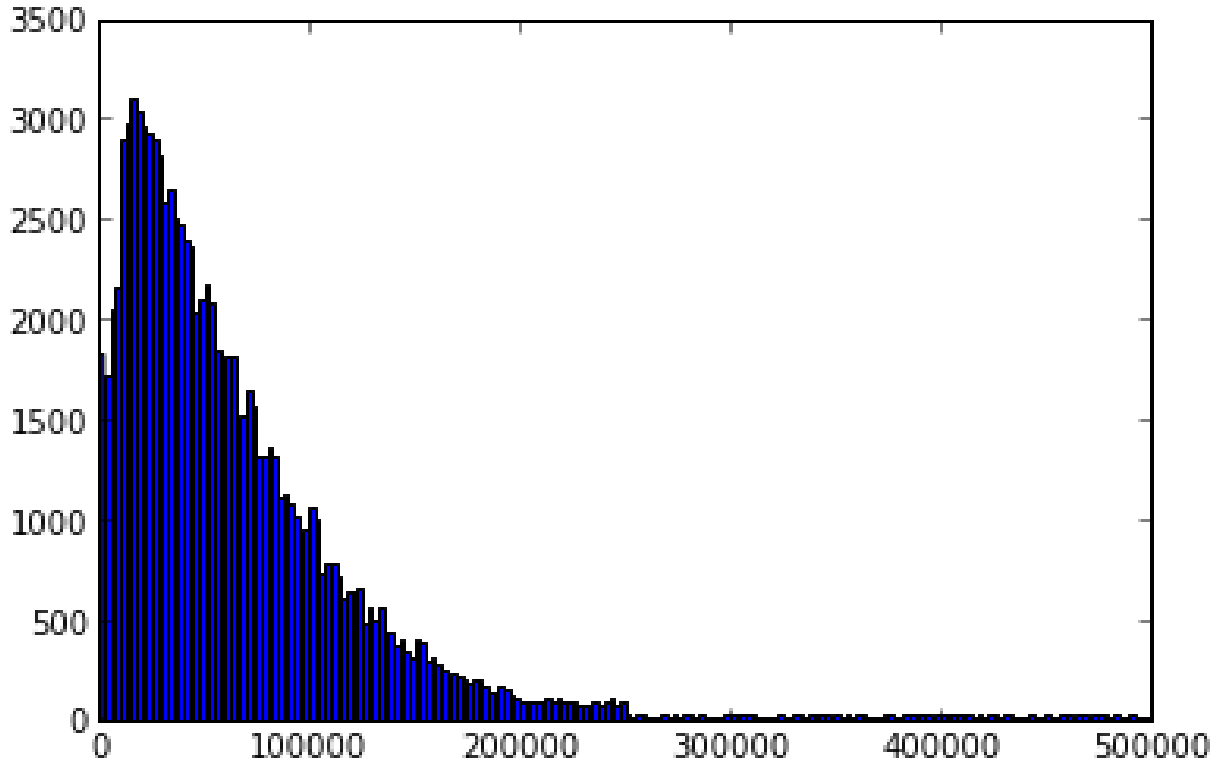
```
N = 100000
incomes = 5000*(empsample(idist,n=N)+rand(100000))
```

The income statistics match what they should be pretty well. We ought to check this because we had to guess in the construction of the histogram at the top end.

```
1 print mean(incomes)
2 print median(incomes)
```

```
67354.6278518
49389.2793994
```

```
1 _=hist(incomes,bins=200)
```



Lorenz Curve and Gini Coefficient

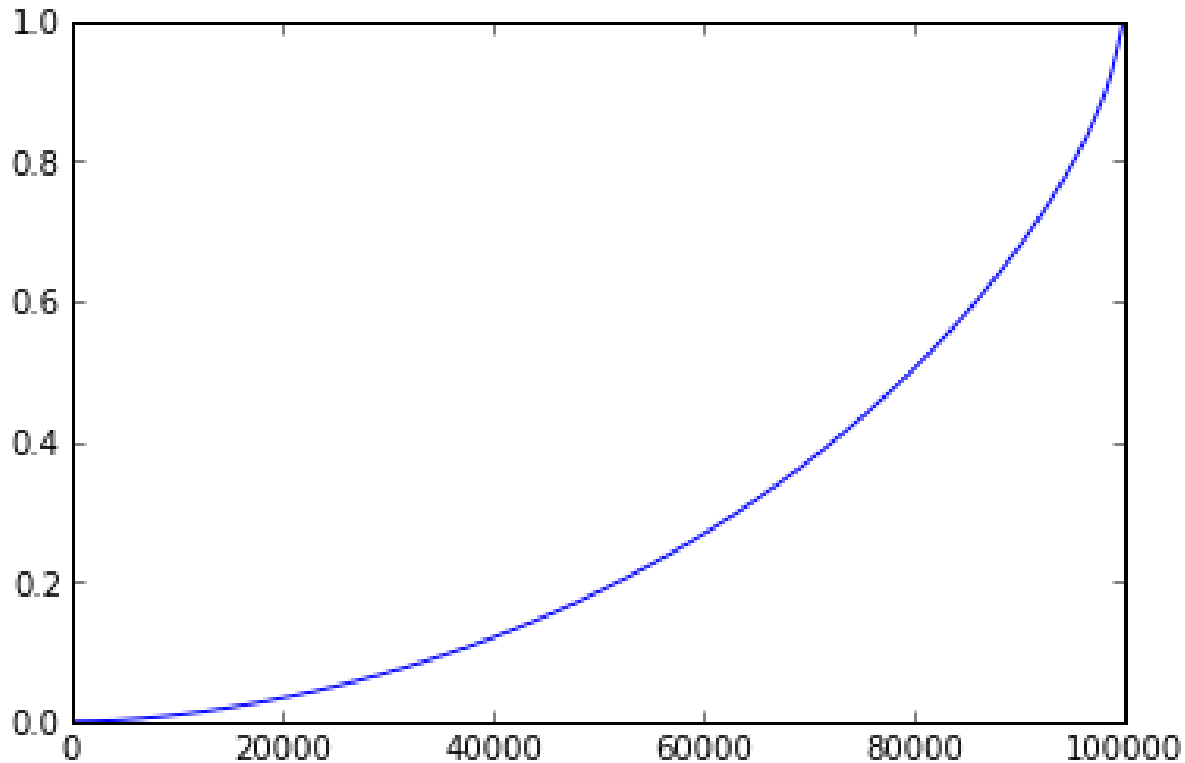
Usually, what we look at in order to measure inequality is the *Gini index*.

This is defined in terms of the Lorenz curve.

The Lorenz curve gives the cumulative fraction of the total wealth/income of all the individuals at the given income rank or below.

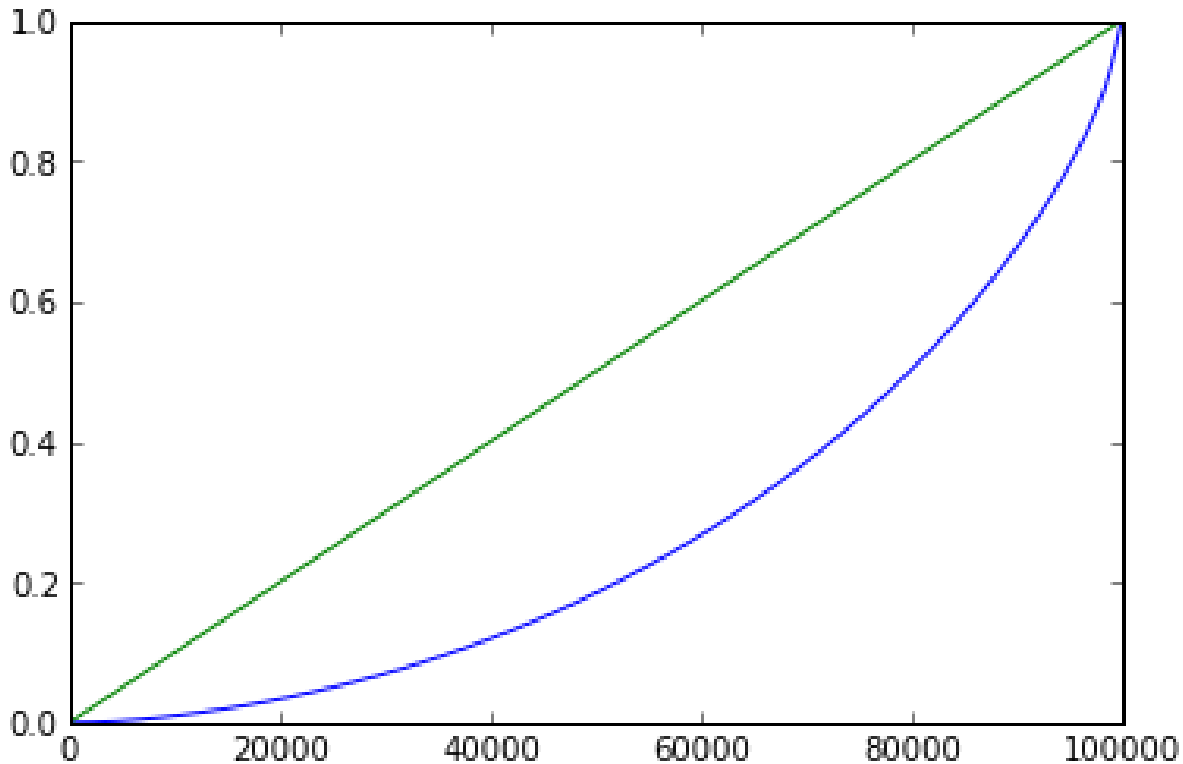
That is, we sort the individuals by income, accumulate the incomes, and normalize by the total.

```
1 lorenz_curve = add.accumulate(sorted(incomes))
2 lorenz_curve /= amax(lorenz_curve)
3 plot(lorenz_curve)
```



If everybody makes about the same income, the Lorenz curve is a straight line.

```
uniform_lorenz = add.accumulate(sorted(66000.0+0.0001*rand(100000))
    )
uniform_lorenz /= amax(uniform_lorenz)
plot(lorenz_curve)
plot(uniform_lorenz)
```



The Gini index is defined as a ratio of the areas on the Lorenz curve diagram. If the area between the line of perfect equality and the Lorenz curve is A, and the area under the Lorenz curve is B, then the Gini index is A / (A + B). Since A + B = 0.5, the Gini index is G = 2 * A or G = 1 2 B.

```
1 gini = 2*sum(linspace(0.0,1.0,len(lorenz_curve))-lorenz_curve)/len(
      lorenz_curve)
2 print gini
```

```
0.463659397039
```

That's about right; the US Gini coefficient in 2007 was 0.45.

```
def gini(samples):
    lorenz_curve = add.accumulate(1.0*numpy.sort(samples))
    lorenz_curve /= amax(lorenz_curve)
    return 2*sum(linspace(0.0,1.0,len(lorenz_curve))-lorenz_curve)/
        len(lorenz_curve)
```

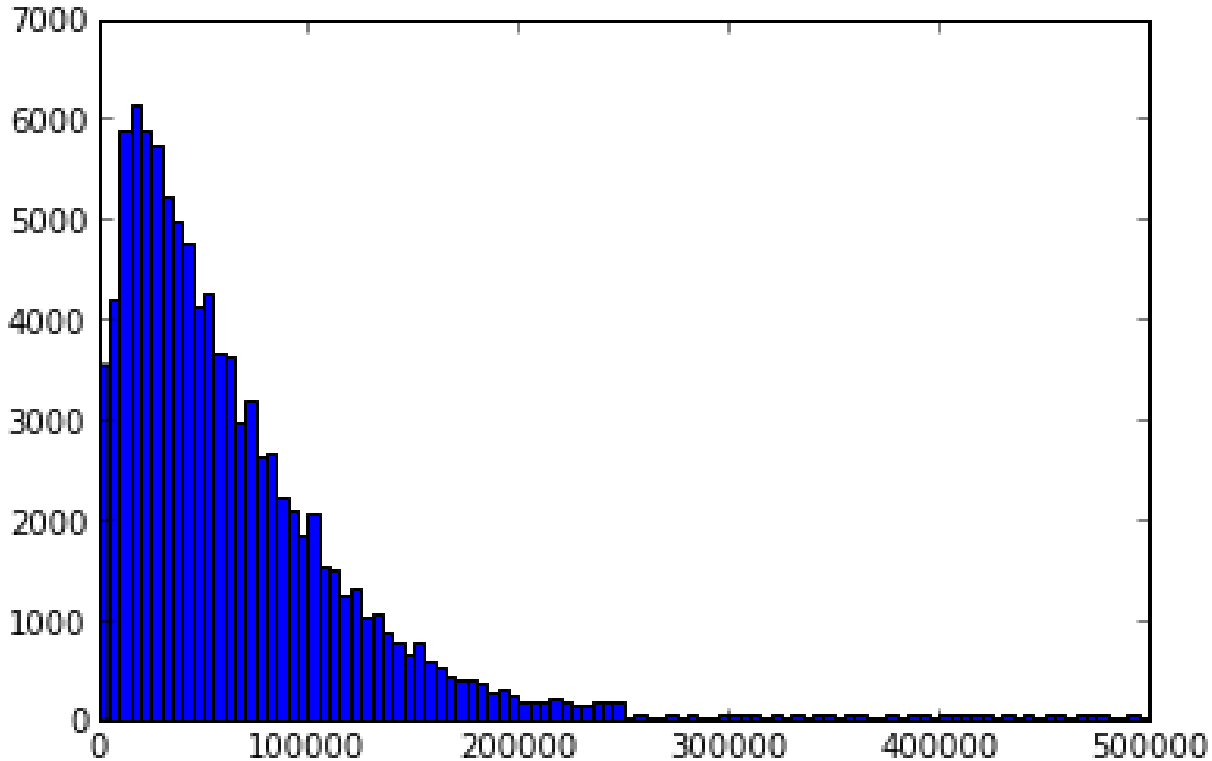
```
1 gini(incomes)
```

```
0.4636593970393926 3
```

Parametric Income Distributions

Let's look at the income distribution again.

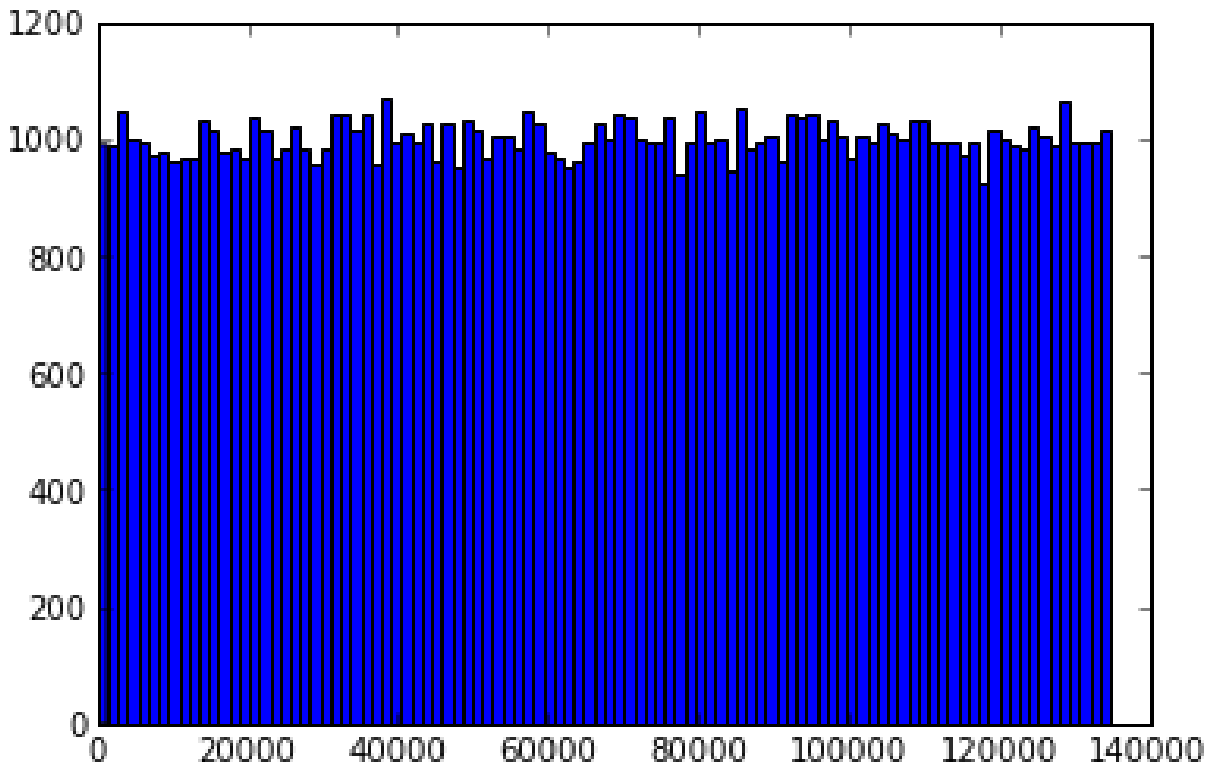
```
1 _=hist(incomes,bins=100)
```



Can we match this with a parametric distribution?

The uniform distribution doesn't work; its shape is all wrong.

```
uincomes = rand(N)*2*mean(incomes)
_=hist(uincomes,bins=100)
```



It produces a variance and a Gini coefficient that are too small. We can't even reproduce the US income with that.

```
1 print var(uincomes)**.5
2 print gini(uincomes)
```

```
38846.4648883
0.332483069236
```

```
1 print mean(incomes)
2 print median(incomes)
3 print var(incomes)**.5
```

```
67354.6278518
49389.2793994
66577.3126328
```

It turns out that the log-normal distribution is a fairly good match for income distributions, except at the high end, where it produces too many outliers.

Log-normal distribution:

$$X = e^{\mu+\sigma Z} \tag{1}$$

We want to match the parameters of the log-normal distribution to the actual parameters of the US income distribution. We can do that using these formulas.

mean: $e^{\mu+\sigma^2/2}$

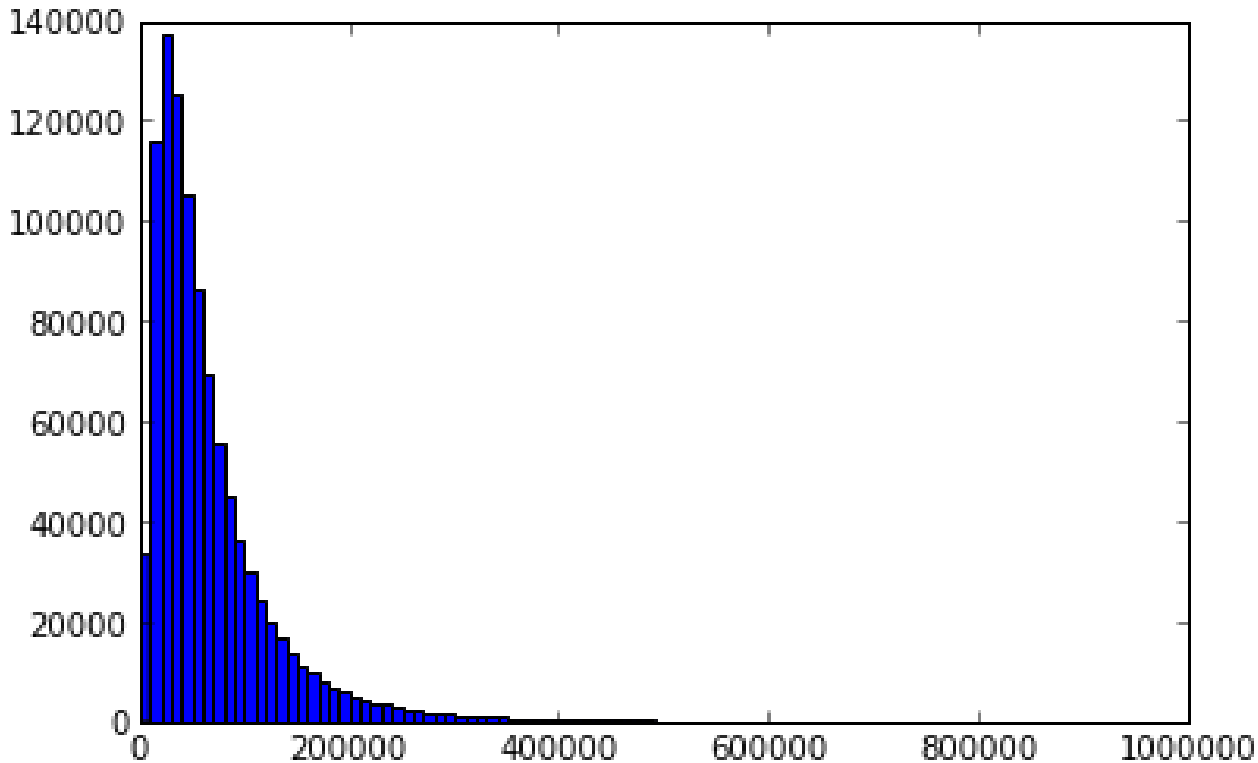
median: e^{μ}

variance: $(e^{\sigma^2}-1)\ e^{2\mu+\sigma^2}$

We cap incomes at USD 1m to avoid the outliers.

```
mu = 10.8
sigma = 0.82
lnincomes = minimum(1e6,exp(randn(1000000)*sigma+mu))
```

```
1 _=hist(lnincomes,bins=100)
```



```
1 print mean(lnincomes)
2 print median(lnincomes)
3 print var(lnincomes)**.5
4 print gini(lnincomes)
```

```
68662.9122598
49108.5031911
66890.4497484
0.438028713623
```

Why log-normal?

- normal distribution from central limit theorem
- large number of independent additive contributions
- log-normal distribution also from central limit theorem
- large number of independent *factors*

$$r = \prod_i (1 + \epsilon_i)$$

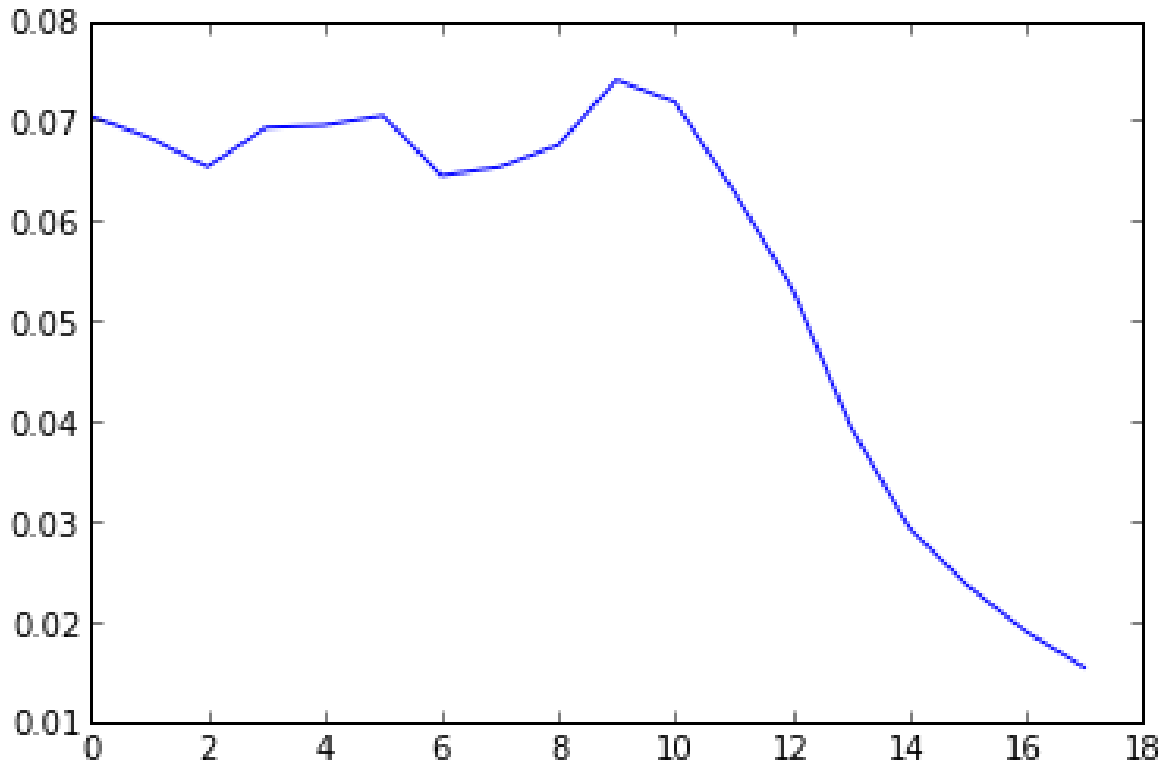
Log-normal tails are different from actual income distribution, but accounts for about 99

Age Distribution

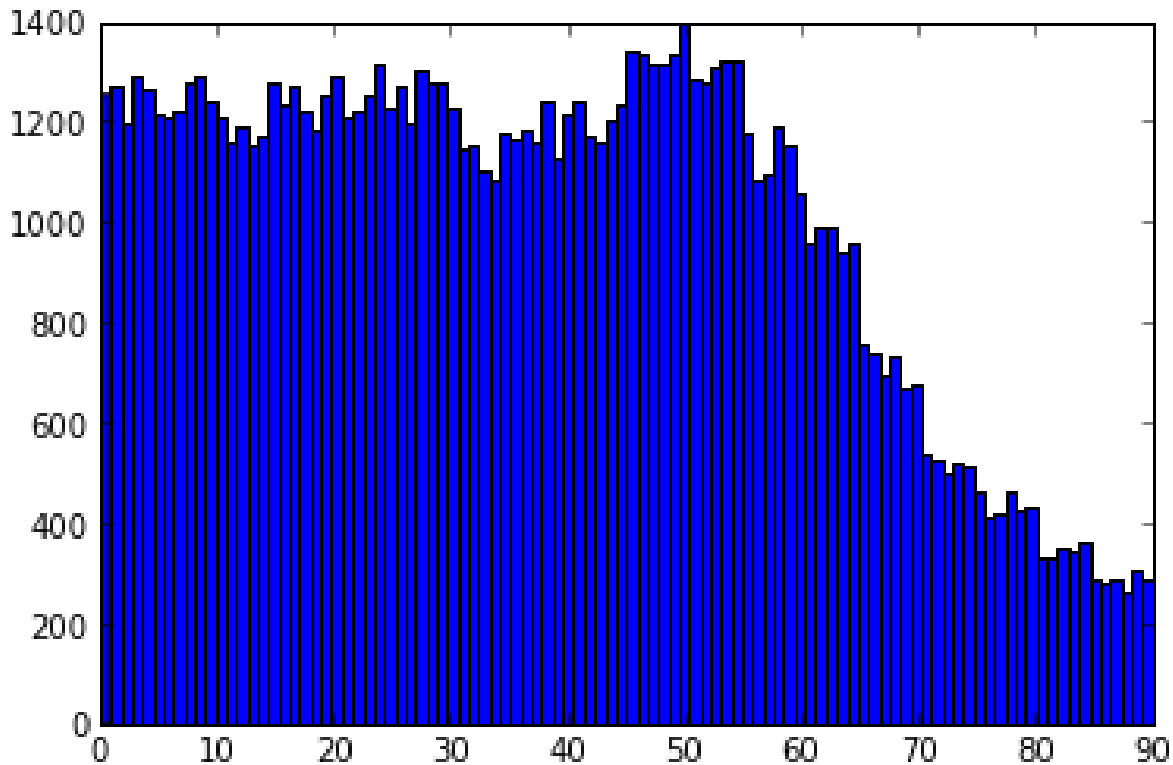
```
data = genfromtxt("usage.csv",skip_header=2,delimiter=",",names="lo
    ,both,m,f")
for i in range(5): print data[i]
```

```
(0.0, 21434.0, 10955.0, 10479.0)
(5.0, 20785.0, 10624.0, 10162.0)
(10.0, 19893.0, 10178.0, 9714.0)
(15.0, 21086.0, 10719.0, 10367.0)
(20.0, 21154.0, 10684.0, 10470.0)
```

```
1 adist = [row[1] for row in data]
2 adist /= sum(adist)
3 plot(adist)
```



```
ages = (empsample(adist,n=N)+rand(N))*5.0
_=hist(ages,bins=100)
```



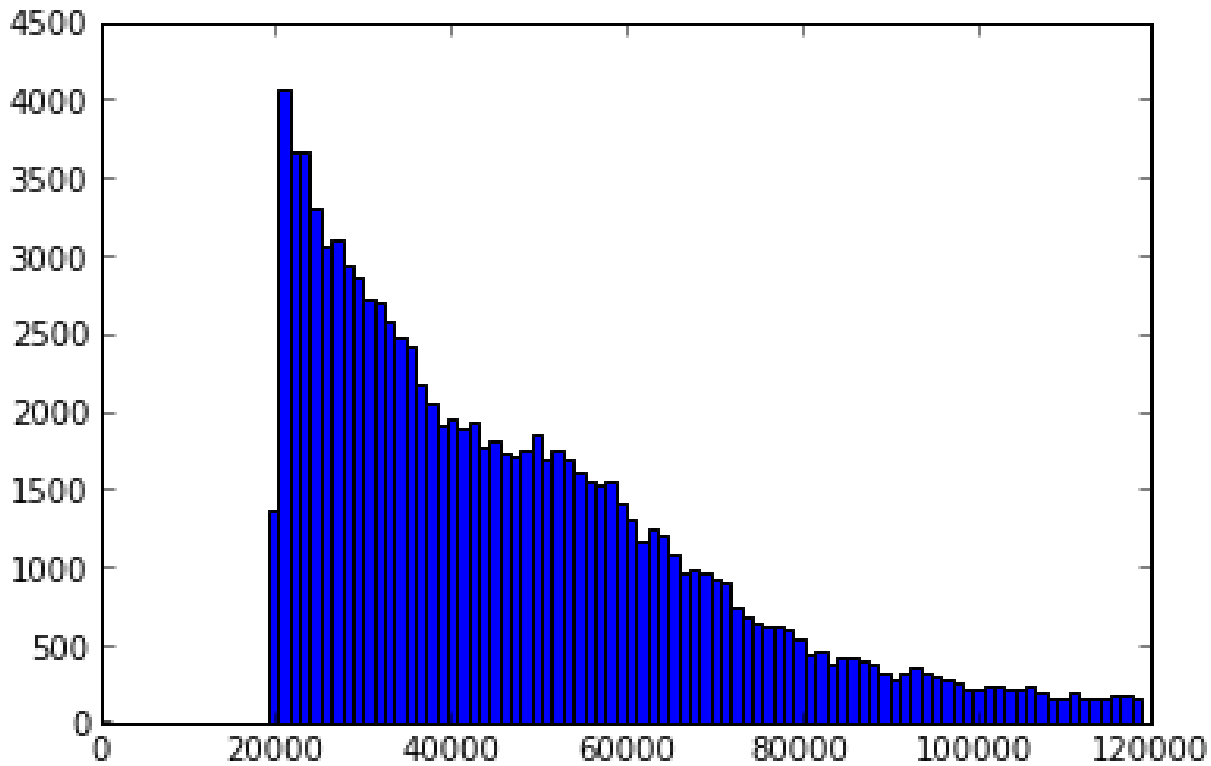
Gini Index with Equal Starting Conditions and Growth

Let's look at the Gini coefficient of income distributions in the presence of growth.

- everybody starts with the same income
- everybody gets the same raise every year

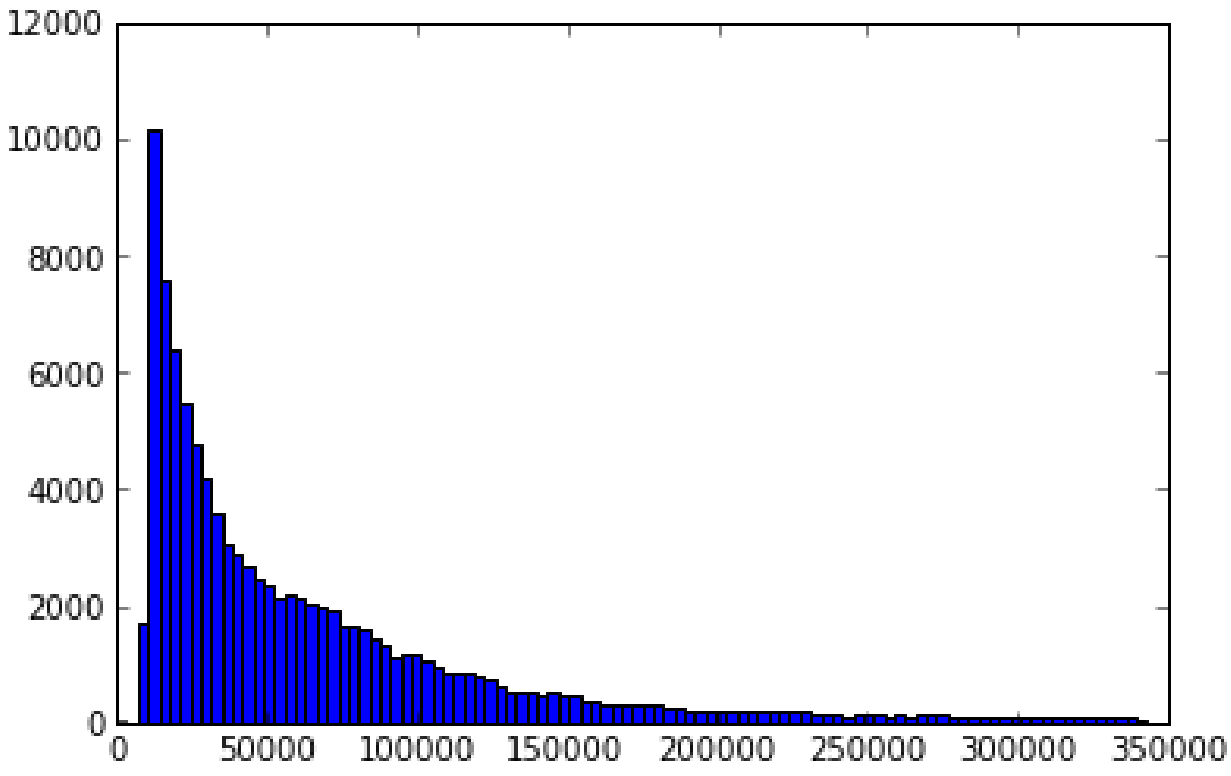
```
1 gincomes = 21000*1.02**ages
2 print mean(gincomes)
3 print gini(gincomes)
4 _=hist(20000*1.02**ages,bins=100,range=(0,120000))
```

```
48855.8947279
0.256307522736
```



```
1 gincomes = 10000*1.04**ages
2 print mean(gincomes)
3 print gini(gincomes)
4 _=hist(gincomes,bins=100,range=(0,350000))
```

```
64618.4564894
0.47489228661
```



US historical GDP rate:

1950: USD 293.7b, population: 151m

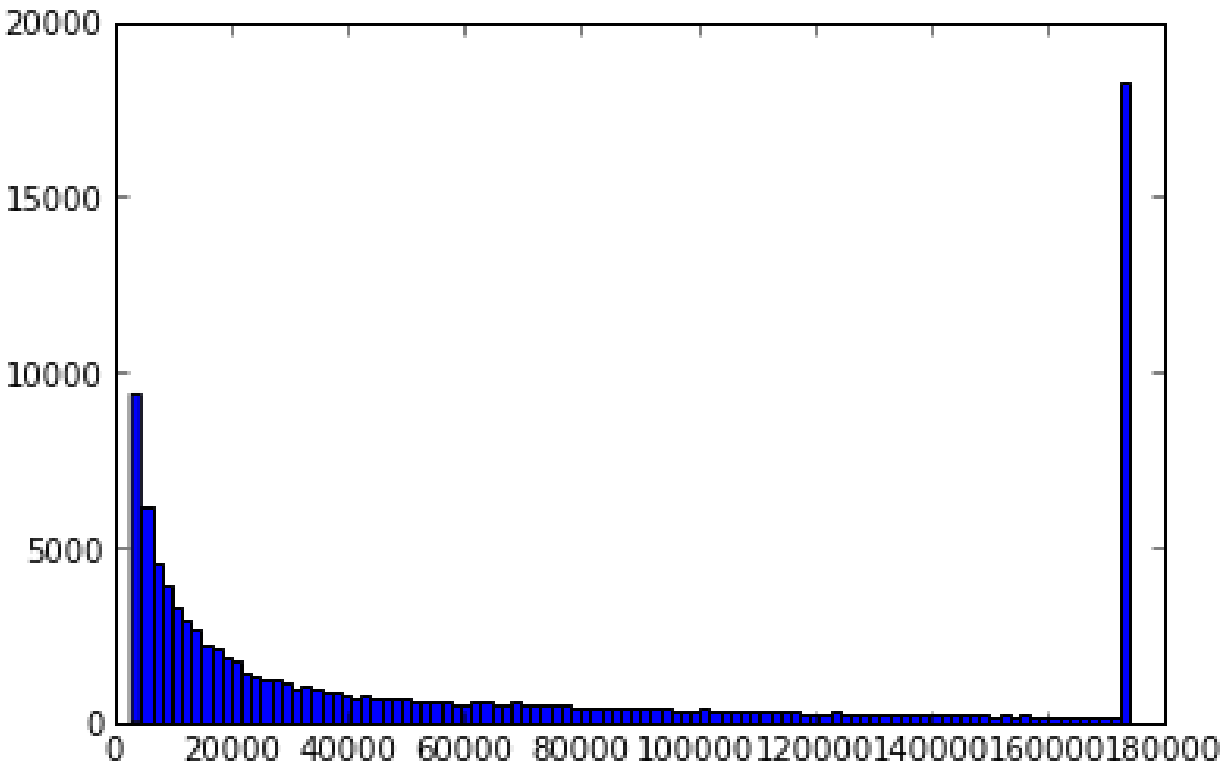
2010: USD 14498b, population: 308m

```
1 100*((14498/293.7)**(1.0/60)-1.0)
```

```
6.7144922854040083
```

```
gincomes = 3000*1.07**minimum(ages,60.0)
print mean(gincomes)
print gini(gincomes)
_=hist(gincomes,bins=100)
```

```
65355.6737054
0.53052393153
```



Equality and Fairness

- everybody started out with the same income
- everybody was subject to the same growth rate

Is this fair? Should older workers get more money due to growth?

Income growth itself is usually considered fair, but it produces inequality.

Economics and Social Sciences View

This model is similar to a "random walk" model used in economics; economists usually are looking for more complicated explanations:

- it may not exactly describe the tails of the distribution
- economists and social scientists prefer theories involving actual choice and behavior

Important Conclusion

Large differences in Gini coefficients can result from simple differences in overall economic success.

Differences in Gini coefficients between societies/groups do not imply lack of fairness or equal treatment.

Low growth rates imply lower Gini coefficients in simple economies.