In [8]:

```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

The Problem

Here we're going to look at some census data from http://www.census.gov/hhes/socdemo/education/data/cps/historical/index.html (http://www.census.gov/hhes/socdemo/education/data/cps/historical/index.html)

Specifically we'll look at the **percent of adults over the age of 25 with a college degree by year**

Is participation growing and at what rate?

A full report is here: http://www.census.gov/prod/2012pubs/p20-566.pdf (http://www.census.gov/prod/2012pubs/p20-566.pdf)

Load some data

In [24]:

```
d = np.loadtxt('year.txt')
year = d[:,0]
participation_all = d[:,1]
```

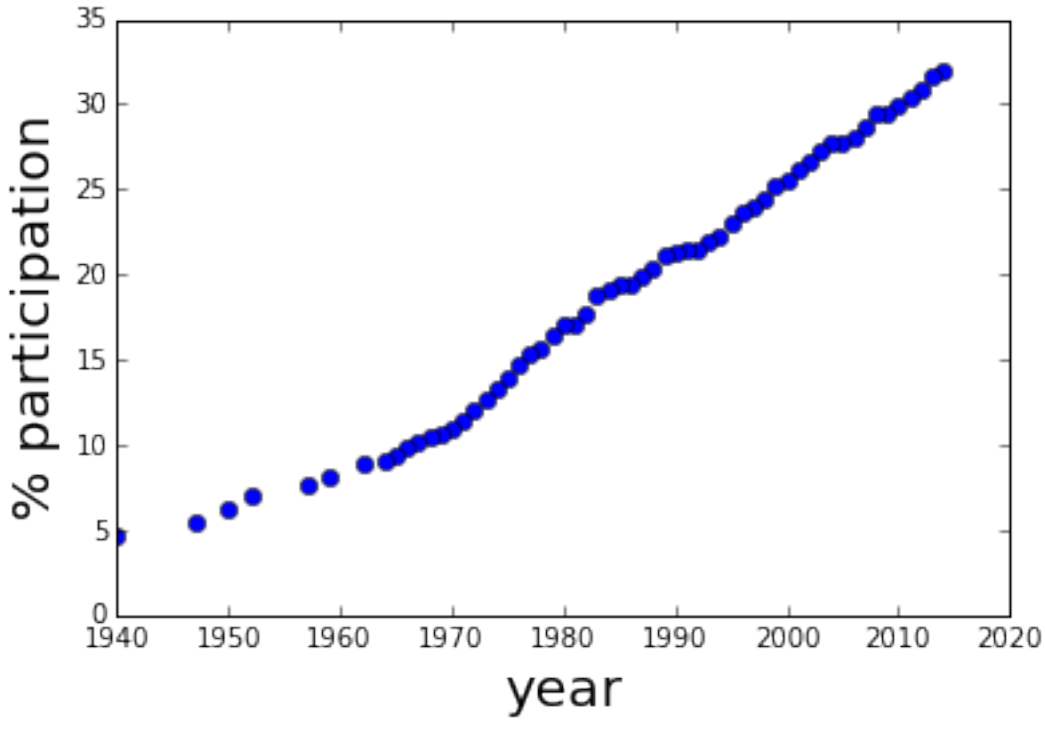
Plot year versus participation

In [29]:

```
plt.plot(year, participation_all, 'o')
plt.xlabel('year', fontsize=20)
plt.ylabel('% participation', fontsize=20)
```

Out[29]:

```
<matplotlib.text.Text at 0x112d35e10>
```



Here we see the trend *looks* linear. Let's try to fit the data to make some observations

To do this, let's let `t` be time and participation `b`. If we assume the data behaves like:

$$b_i = x_0 + x_1 t_i$$

for each year i, then we're assuming the growth is linear in time.

What are x_0 and x_1 in this case?

In [30]:

```
n = len(perc)
A = np.ones((n,2))
A[:,1] = year
b = participation_all
```

We have a big system:

$$Ax = b$$

where b is the participation and x are the parameters that determine the shape of the linear growth. We can solve this with

1. pseudo-inverse (bad idea) $x = (A^T A)^{-1} A^T b$
2. QR factorization (hold on!)

In [31]:

```
x = np.linalg.solve(A.T.dot(A), A.T.dot(b))
print(x)
```

```
[ -8.15531563e+02   4.20432737e-01]
```

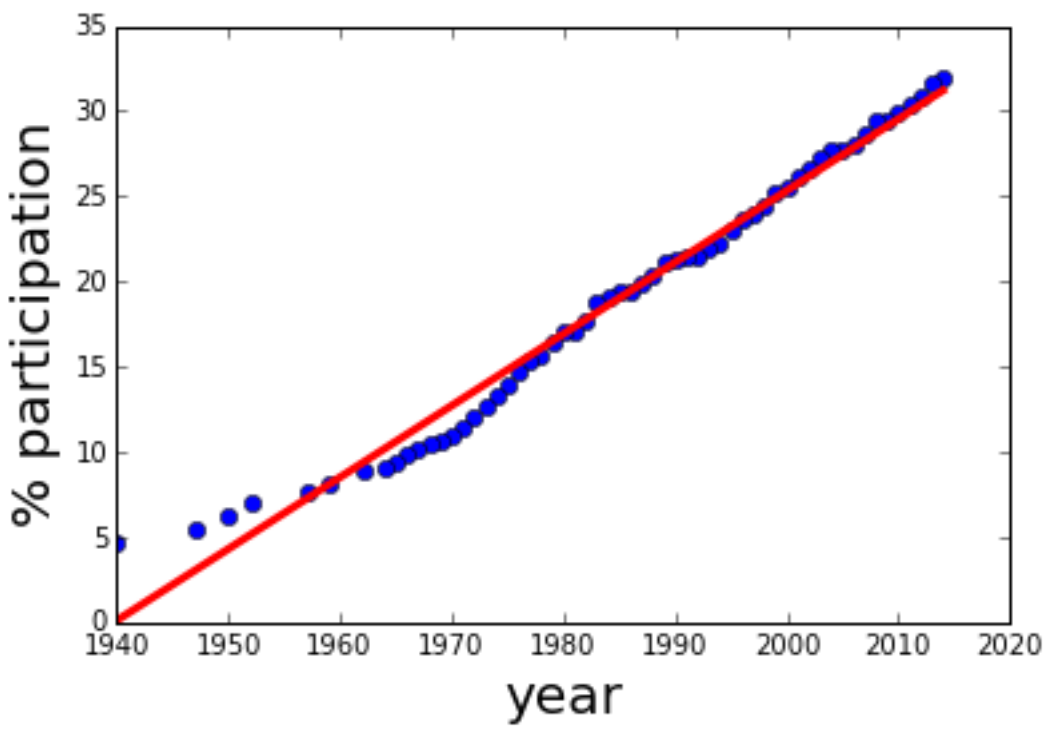
Now let's plot the line to see if it matches up:

In [34]:

```
plt.plot(year, participation_all, 'o')
t = np.linspace(year.min(), year.max(), 100)
plt.plot(t, x[0] + x[1]*t, 'r-', lw=3)
plt.xlabel('year', fontsize=20)
plt.ylabel('% participation', fontsize=20)
```

Out[34]:

```
<matplotlib.text.Text at 0x112e39048>
```



In [ ]: