Lab 6: Pareto and the 1 percent - continued!

Stat 597A

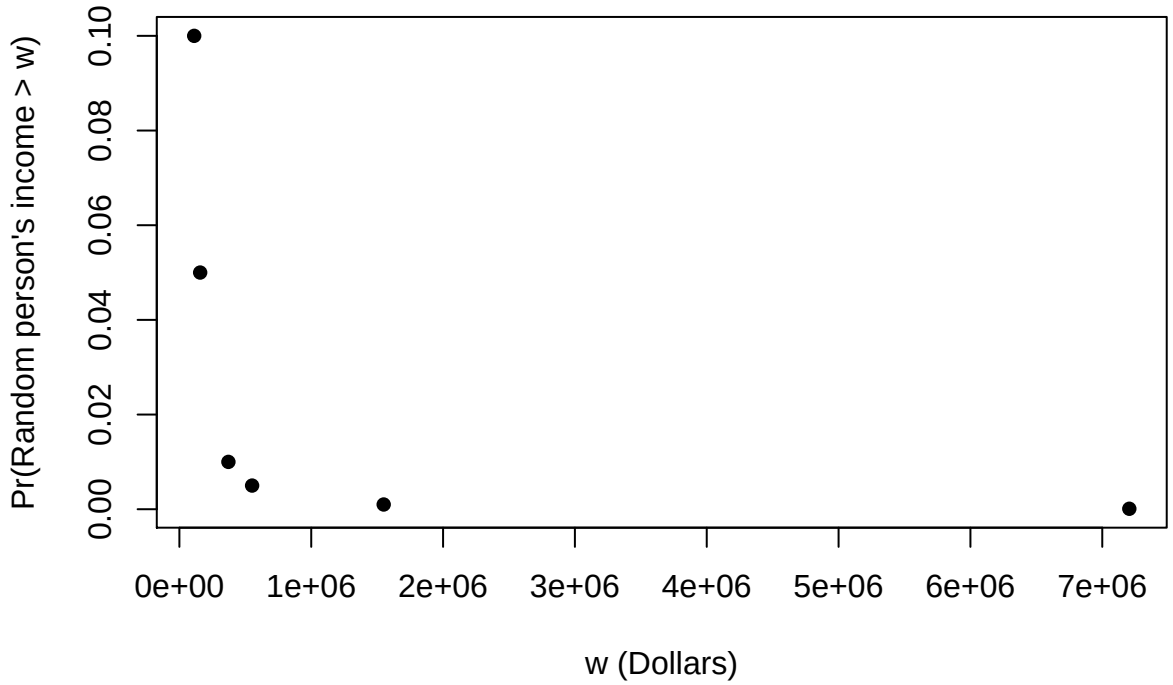
Friday, 16 October 2015

There have been a few questions about the previous lab and how it relates to statistics. Today's lab will work to try to clear up those issues. First let's load the data, make it a little easier to work with, and plot it.

```
# read in income data
data <- read.csv("http://people.math.umass.edu/~jstauden/wtid-report.csv")
data <- data[data$Year == "2012", ]
data.2012 <- data.frame(percentile = c(0.1, 0.05, 0.01, 0.005, 0.001, 1e-04),
    income = unlist(data[-(1:2)]))
data.2012
```

```
##                          percentile  income
## P90.income.threshold          1e-01  112200
## P95.income.threshold          5e-02  157510
## P99.income.threshold          1e-02  371689
## P99.5.income.threshold        5e-03  551622
## P99.9.income.threshold        1e-03 1549616
## P99.99.income.threshold       1e-04 7205236
```

```
plot(data.2012$income, data.2012$percentile, xlab = "w (Dollars)", ylab = "Pr(Random person's income > w
    type = "n")
points(data.2012$income, data.2012$percentile, pch = 16)
```



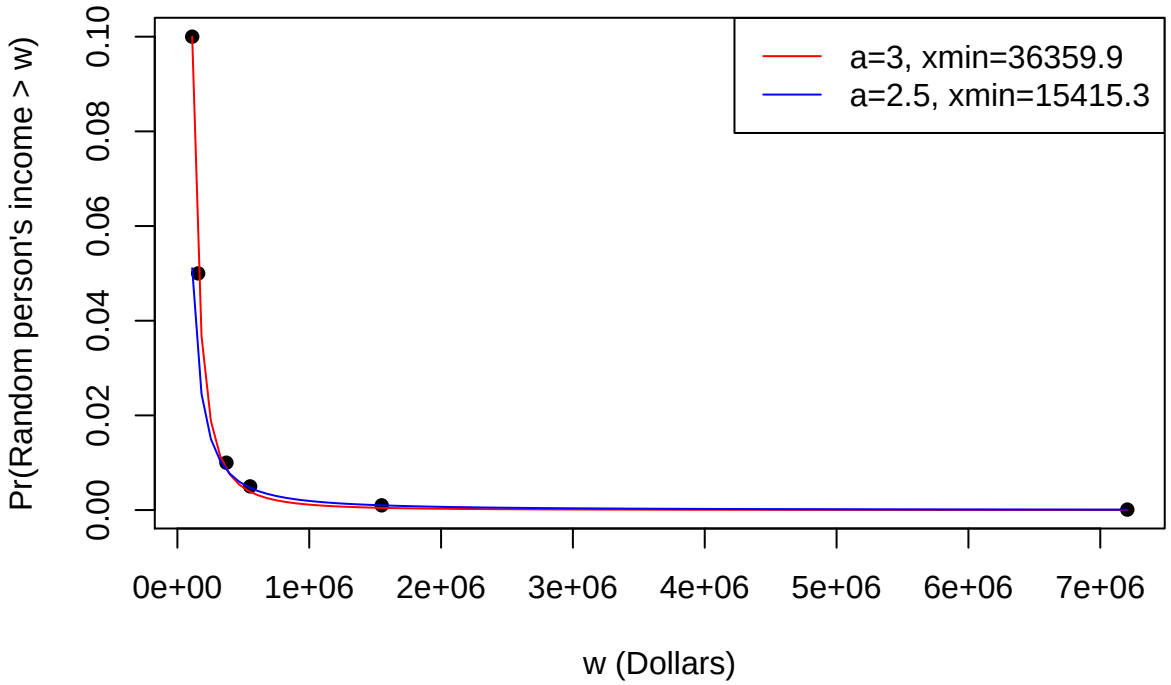
If the incomes have a pareto distribution, then the points in that plot should lie on a curve that has the form:

$$Pr(X \geq w) = \left(\frac{w}{x_{min}}\right)^{-a+1}. \quad (1)$$

Note that w is the independent variable in the function, $Pr(X \geq w)$ ("percentile") is the dependent variable, and a and x_{min} are unknown parameters. In order to actually draw that curve over the points, we need to estimate a and x_{min}. Luckily, we're statisticians, and we can estimate parameters from data!

Last lab described one way to estimate those parameters. Below are functions to do that. After that we apply the functions to the data, estimate a and x_{min} two different ways, and plot the results.

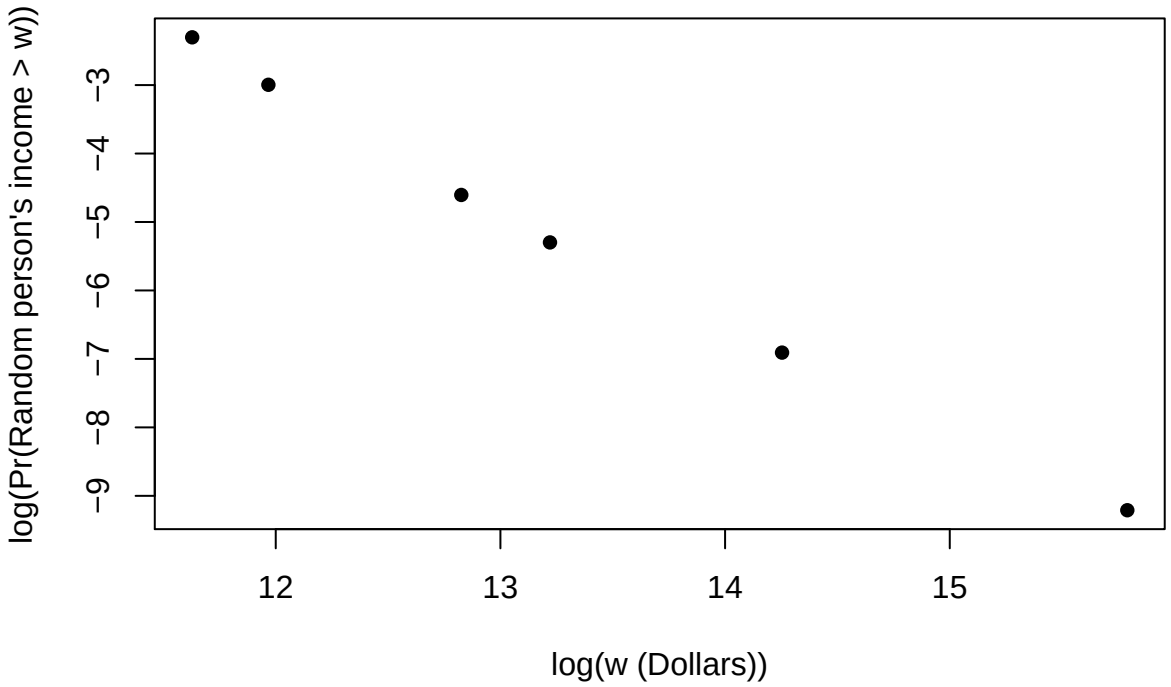
```
# below is a more general version of the a estimation method
estimate.a <- function(inc.1, inc.2, p.1, p.2) {
    # required: inc.1 < inc.2, so p.1<p.2
    a <- 1 - (log(p.1/p.2)/log(inc.1/inc.2))
    return(a)
}

# this function uses the method I described in class.
estimate.xmin <- function(inc, p, a) {
    xmin <- inc/(p^(1/(-a + 1)))
    return(xmin)
}

# use some of the data to make two estimates of a
a.1 <- estimate.a(data.2012$income[1], data.2012$income[2], data.2012$percentile[1],
    data.2012$percentile[2])
a.2 <- estimate.a(data.2012$income[5], data.2012$income[6], data.2012$percentile[5],
    data.2012$percentile[6])


# use some of the data to make two estimates of x_{min}
xmin.1 <- estimate.xmin(data.2012$income[1], data.2012$percentile[1], a.1)
xmin.2 <- estimate.xmin(data.2012$income[6], data.2012$percentile[6], a.2)

# here's the Pr(X>x) function according to the pareto distribution
tail.prob <- function(x, a, xmin) {
    prob <- (x/xmin)^(-a + 1)
    prob[x < xmin] <- 0
    return(prob)
}


# plot the data and the estimated functions
plot(data.2012$income, data.2012$percentile, xlab = "w (Dollars)", ylab = "Pr(Random person's income > w
    type = "n")
points(data.2012$income, data.2012$percentile, pch = 16)
curve(tail.prob(x, a.1, xmin.1), from = min(data.2012$income), to = max(data.2012$income),
    add = T, col = "red")
curve(tail.prob(x, a.2, xmin.2), from = min(data.2012$income), to = max(data.2012$income),
    add = T, col = "blue")
legend("topright", lty = c(1, 1), col = c("red", "blue"), legend = c(paste("a=",
    round(a.1, 1), ", xmin=", round(xmin.1, 1), sep = ""), paste("a=", round(a.2,
    1), ", xmin=", round(xmin.2, 1), sep = "")))
```



It is unsatisfying that we get a different curve depending on which parts of the data we use to estimate a and x_{min}! It would be nice to use all the data and get one estimate. Below is one way to do that.

Use the data to plot of log(income) vs log(percentile).

```
plot(log(data.2012$income), log(data.2012$percentile), xlab = "log(w (Dollars))",
    ylab = "log(Pr(Random person's income > w))", type = "n")
points(log(data.2012$income), log(data.2012$percentile), pch = 16)
```



This suggests that there might be a linear relation ship between log(w) and log(Pr(X>w)). Let's see if that's true for the pareto model.

Recall that the pareto model says that when $w > x_{min}$,

$$Pr(X \geq w) = \left(\frac{w}{x_{min}}\right)^{-a+1}, \tag{2}$$

This means that

$$\log\left[(Pr(X \geq w)\right] = (-a+1)\left[\log(w) - log(x_{min})\right] \tag{3}$$

which means

$$\log\left[(Pr(X \geq w)\right] = -log(x_{min})(-a+1) + (-a+1)\log(w). \tag{4}$$

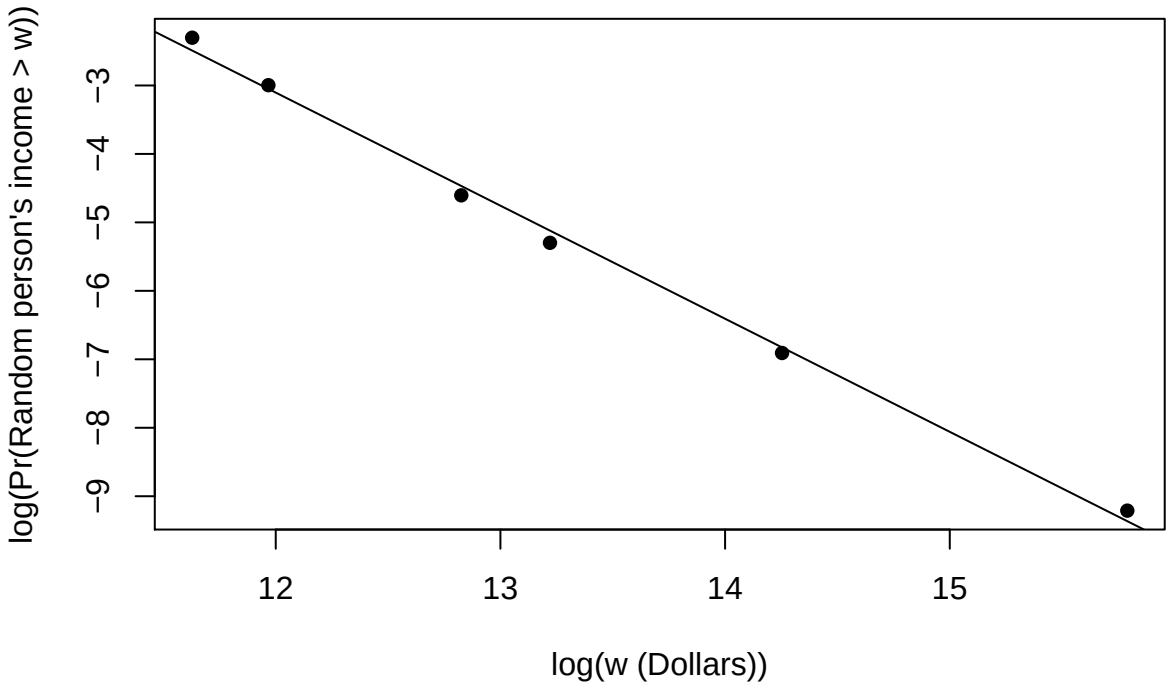
This means the that the pareto model says that log(percentile) is a linear function of log(income)! That suggests that we can use linear regression to estimate a and x_{min}.

$$\log\left[(Pr(X \geq w)\right] = \beta_0 + \beta_1 \log(w). \tag{5}$$

with $\beta_1 = (-a+1)$ and $\beta_0 = -(-a+1)\log(x_{min})$ which means that $a = 1 - \beta_1$, $\log(x_{min}) = -\beta_0/\beta_1$, and $x_{min} = \exp(-\beta_0/\beta_1)$.

Code below fits the linear regression, extracts the parameters and draws the line.

```
plot(log(data.2012$income), log(data.2012$percentile), xlab = "log(w (Dollars))",
    ylab = "log(Pr(Random person's income > w))", type = "n")
points(log(data.2012$income), log(data.2012$percentile), pch = 16)
fit <- lm(log(data.2012$percentile) ~ log(data.2012$income))
beta.0 <- coef(fit)[1]
beta.1 <- coef(fit)[2]
abline(fit)
```



Your only task for today (other than reading all this!) is to make a plot similar to the second one above where you use linear regression to estimate a and x_{min}.