Log Transformation

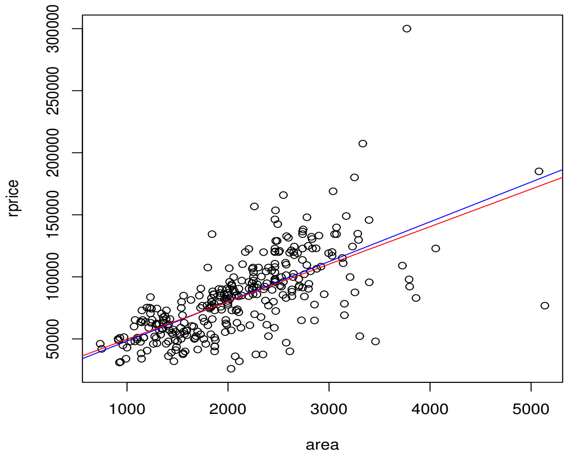
(Jing Li, Miami University)

1. Log transformation is very common in data analysis, and you may wonder why. This note discusses several cases that call on log transformation. First of all, I use log to denote natural log, that is, log is ln you see in other books. Recall that log function $y = \log(x)$ is defined for $x > 0$; as x rises, the function increases (because $\frac{dy}{dx} = \frac{1}{x} > 0$) at decreasing rate (because $\frac{d^2y}{dx^2} = \frac{-1}{x^2} < 0$). The log function has no turning point because the first order equation $\frac{1}{x} = 0$ does not have a solution.

2. Notice that $\log(10000) = 9.21034 < 10000$. So log transformation is able to reduce the impact of a large value. Given this property, we can

   (a) use log transformation to downplay outliers (extreme values). The OLS method is sensitive to outliers because the presence of outliers can greatly affect line fitting. I use house data as an illustration. In the scatter plot of rprice against area shown below, we see there is an outlier—a house with price greater than 250000. Two OLS lines fitted with and without that outlier are included in the scatter plot

```
> ad = "https://www.fsb.miamioh.edu/lij14/400_house.txt"
> data = read.table(url(ad), header=T)
> attach(data)
> plot(area,rprice)
> abline(lm(rprice~area),col="blue")
> abline(lm(rprice[rprice<250000]~area[rprice<250000]),col="red")
```



Now we can compare OLS results with and without outliers

```
> summary(lm(rprice~area))$coef
               Estimate Std. Error   t value     Pr(>|t|)
(Intercept) 16225.83153 4381.25679  3.703465 2.505499e-04
area           32.03807    1.97527 16.219593 1.448699e-43
> summary(lm(rprice[rprice<250000]~area[rprice<250000]))$coef
                         Estimate  Std. Error   t value     Pr(>|t|)
(Intercept)           19481.95234 4091.739406  4.761289 2.926629e-06
area[rprice < 250000]    30.24629    1.850211 16.347485 4.997755e-44
> (32-30)/32
[1] 0.0625
```

We see the difference in the slope coefficient with and without the outlier is about 6.25 percent. By contrast, after taking log of rprice and running the log-level model, the difference in the slope coefficient with and without outliers reduces to 2.14 percent. The takeaway is, after log transformation, the issue of outliers becomes secondary

```
> summary(lm(log(rprice)~area))$coef
                Estimate   Std. Error   t value     Pr(>|t|)
(Intercept) 1.047457e+01 5.151353e-02 203.33629 0.000000e+00
area        3.734732e-04 2.322464e-05  16.08091 4.990752e-43
> summary(lm(log(rprice)[rprice<250000]~area[rprice<250000]))$coef
                          Estimate   Std. Error   t value     Pr(>|t|)
(Intercept)           1.048913e+01 5.137758e-02 204.15776 0.000000e+00
area[rprice < 250000] 3.654595e-04 2.323202e-05  15.73086 1.208208e-41
> (3.73-3.65)/3.73
[1] 0.02144772
```

(b) use log to mitigate heteroskedasticity. The standard error, t value, and p-value reported by R lm function assume homoskedasticity (constant variance). However, we see evidence of heteroskedasticity in house data—the variance of rprice when area is less than 3000 is 775143694, smaller than the variance 2531633752 when area is greater than 3000. The variance ratio is 3.266019. Actually in the scatter plot we see that the rprice becomes more spread out as area rises. This finding implies heteroskedasticity, which invalidates the conventional standard error, t value, and p-value (we need to use heteroskedasticity-robust standard error)

```
> var(rprice[area<3000])
[1] 775143694
> var(rprice[area>3000])
[1] 2531633752
> var(rprice[area>3000])/var(rprice[area<3000])
[1] 3.266019
> var(log(rprice)[area>3000])/var(log(rprice)[area<3000])
[1] 1.130154
```

After taking log, the variance ratio drops to 1.130154, close to unity. So the data become almost homoskedastic after taking log. Put differently, the reported standard error, t value, and p-value in the log level model are more likely not to suffer from heteroskedasticity.

(c) For the method of maximum likelihood and Bayesian inference, log transformation can mitigate the issue of overflowing.

```
> prod(rprice)
[1] Inf
> sum(log(rprice))
[1] 3614.902
```

For instance, instead of directly computing

$$y_1 y_2 \cdots y_n,$$

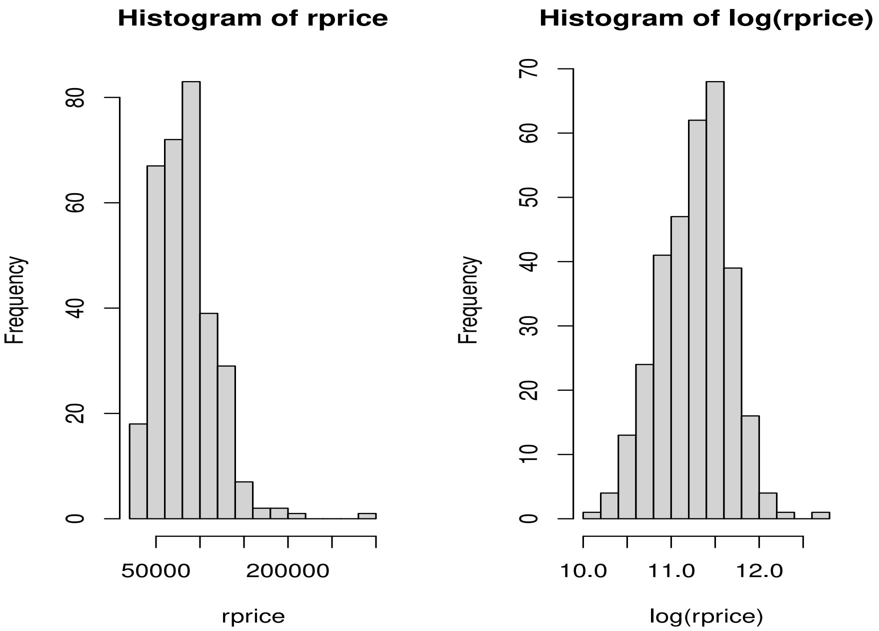
which can be extremely large if y is house price (R reports the product as infinity Inf, an example of data overflowing), we can consider the log transformation

$$y_1 y_2 \cdots y_n = e^{\log(y_1 y_2 \cdots y_n)} = e^{\sum_i \log(y_i)} = e^{3614.902}$$

which clearly is not infinity, so the overflowing issue is prevented

3. If y follows a skewed log-normal distribution, then its log follows symmetric normal distribution.

```
> par(mfrow = c(1, 2))
> hist(rprice)
> hist(log(rprice))
```



We see the histogram of rprice becomes more symmetric or more bell-shaped after taking log. To see why, suppose every year the price of a house grows by a random percentage

$$p_t = p_0 g_1 g_2 ... g_t \Rightarrow \log(p_t) = \log(p_0) + \sum_{i=1}^{t} \log(g_i) \sim Normal\ distribution \tag{1}$$

where p_0 is the initial price, and g_i is the i-th year growth rate. The last step follows because the central limit theorem implies that the process of adding random values leads to a normal distribution. Similarly, that explains why the log of wage is often less skewed than wage. To sum up, the log transformation can reduce not only variance, but also skewness.

4. Since log function is nonlinear, we can capture nonlinearity between y and x by taking log. For instance, we may believe that as area rises the price will rise, but at decreasing rate (similar to diminishing marginal utility). In other words, a nonlinear level-log model may be more appropriate than the linear level-level model.

```
> mean(rprice[area>2000&area<3000])-mean(rprice[area>1000&area<2000])
[1] 34016.91
> mean(rprice[area>3000&area<4000])-mean(rprice[area>2000&area<3000])
[1] 25552.21
```

Indeed we see that as area changes from the range of $(1000, 2000)$ to $(2000, 3000)$, the average price increases by a bigger amount than area changes from $(2000, 3000)$ to $(3000, 4000)$

5. If we think y changes by a constant percentage rather than a constant amount, then the log level model is suitable

   $$\log(y) = \beta x + u \Rightarrow \beta = \frac{d\log(y)}{dx} = \textit{constant percent change}$$

   In short, log transformation can provide percent interpretation

6. Finally, keep mind that log transformation cannot be applied to negative values and 0!