Regression with Nonlinear Transformations

Joel S Steele
Portland State University

Abstract

Gaussian Likelihood

When data are drawn from a Normal distribution, $\sim \mathcal{N}(\mu, \sigma^2)$, we can use the Gaussian distribution function to describe the probability of the data.

$$f(x_i|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\left(-\frac{(x_i-\mu)^2}{2\sigma^2}\right)} \tag{1}$$

This specifications represents how to compute the probability for a single value x_i. That means, we can get the value of the function for any particular input, x_i, if we supply the parameters μ and σ^2.

A quick aside

You may be wondering why we have discussed probabilities but we are interested in likelihoods? Well, the terms are often used interchangeably, which is a shame. In our application however, we will say that, if we know the parameter values for a distribution, we can compute a probability of any observation we obtain. The result tells us the probability (or how likely we are to see) a value like that given the distribution that we have at hand.

If, however, we don't know the exact parameters or our distribution, but instead we have a set of observations, we must figure out which values of the parameters would result in the largest probability. In this case, we are going *backwards* and using the data along with the hypothesized shape of the probability distribution, in order to find the parameters that we believe produced our observations. In this latter case, we are interested in finding the parameters which maximize the likelihood that our observations are distributed a particular way.

Likelihood of a set of values

The function specification changes when we are dealing with an entire set of observations. From basic probability, we know that, if the observations are independent, their *joint* probability is the product of their individual probabilities. So, for our set of observations, we compute the probability value of each point, and then multiply them all together to get

the probability of the entire sample. What does this mean? Well, we literally multiply each obtained value from the function. The result is,

$$\begin{aligned} f(x_1, x_2, \ldots, x_n|\mu, \sigma^2) &= \prod_i^n f(x_i|\mu, \sigma^2) \\ &= f(x_1|\mu, \sigma^2) \times f(x_2|\mu, \sigma^2) \times \ldots \times f(x_n|\mu, \sigma^2) \end{aligned} \tag{2}$$

Using our Gaussian function this translates to,

$$\begin{aligned} &= \prod_i^n \frac{1}{\sqrt{2\pi\sigma^2}} e^{\left(-\frac{(x_i-\mu)^2}{2\sigma^2}\right)} \\ &= \frac{1}{\sqrt{2\pi\sigma^2}} e^{\left(-\frac{(x_1-\mu)^2}{2\sigma^2}\right)} \times \frac{1}{\sqrt{2\pi\sigma^2}} e^{\left(-\frac{(x_2-\mu)^2}{2\sigma^2}\right)} \times \ldots \times \frac{1}{\sqrt{2\pi\sigma^2}} e^{\left(-\frac{(x_n-\mu)^2}{2\sigma^2}\right)}. \end{aligned} \tag{3}$$

This product can be simplified somewhat. To help illustrate we will take advantage of the fact that the product operator, $\prod_i^n$, can be distributed algebraically.

$$\prod_i^n \frac{1}{\sqrt{2\pi\sigma^2}} e^{\left(-\frac{(x_i-\mu)^2}{2\sigma^2}\right)} = \prod_i^n \left[\frac{1}{\sqrt{2\pi\sigma^2}}\right] \times \prod_i^n \left[e^{\left(-\frac{(x_i-\mu)^2}{2\sigma^2}\right)}\right] \tag{4}$$

Thus, we can deal with each portion one at a time.

Quick and dirty Power rules. Raising a number, say a to a power, b, then raising that quantity to the power c is the same as multiplying the powers together, thus $(a^b)^c = a^{bc}$. For example,

$$(2^2)^3 = 2^{2\times 3} = 2^6 = (2 \times 2) \times (2 \times 2) \times (2 \times 2) = 64.$$

Also of note, is that the product of the same value, or base, let's say 2, raised to different powers is equal to the base raised to the sum of the powers. For example,

$$2^2 \times 2^3 = 2^{2+3} = 2^5 = (2 \times 2) \times (2 \times 2 \times 2) = 32.$$

Okay, back to each portion of equation 4.

First portion

First, we see that the $\frac{1}{\sqrt{2\pi\sigma^2}}$ term does not involve the observation x_i, which makes it a constant. We also know that taking the product of a constant, is equivalent to having the constant multiplied by itself a number of times. In this case n times. So, we can express the first portion of the joint probability as,

$$\prod_i^n \frac{1}{\sqrt{2\pi\sigma^2}} = \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n. \tag{5}$$

Alternatively, we can re-express the fraction,

$$\frac{1}{\sqrt{2\pi\sigma^2}} = (2\pi\sigma^2)^{-\frac{1}{2}}. \tag{6}$$

This is helpful, since we remember that raising a power to a power is the same as multiplying the powers together, $(a^b)^c = a^{bc}$. This means that the product of the first term can be simplified as the fraction to the power n. Again, this is the same as multiplying the powers together. The result is,

$$\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n = \left((2\pi\sigma^2)^{-\frac{1}{2}}\right)^n = (2\pi\sigma^2)^{(-\frac{1}{2})\times(n)} = (2\pi\sigma^2)^{(-\frac{n}{2})}. \tag{7}$$

Second portion

Okay, on to the $e^{\left(-\frac{(x_i-\mu)^2}{2\sigma^2}\right)}$ part. First, we can re-express the entire power portion as $(-\frac{1}{2\sigma^2}) \times (x_i - \mu)^2$, so this can be rewritten as $e^{\left(-\frac{1}{2\sigma^2}(x_i-\mu)^2\right)}$.

It is important to recognize that if we have a base number, raised to a power, multiplied by the same base number, raised to a different power, this is equal to the base raised to the sum of the two powers. For example

$$2^2 \times 2^3 = (2 \times 2) \times (2 \times 2 \times 2) = 2^{2+3} = 2^5 = 32. \tag{8}$$

We can take the product of our exponential part and sum over x_i because,

$$\begin{aligned} \textstyle\prod_i^n e^{\left(-\frac{1}{2\sigma^2}(x_i-\mu)^2\right)} &= e^{\left(-\frac{1}{2\sigma^2}(x_1-\mu)^2\right)} \times e^{\left(-\frac{1}{2\sigma^2}(x_2-\mu)^2\right)} \times \ldots \times e^{\left(-\frac{1}{2\sigma^2}(x_n-\mu)^2\right)} \\ &= e^{\left[-\frac{1}{2\sigma^2}(x_1-\mu)^2 + -\frac{1}{2\sigma^2}(x_2-\mu)^2 + \ldots + -\frac{1}{2\sigma^2}(x_n-\mu)^2\right]} \end{aligned} \tag{9}$$

Now, factor out the common multiple $-\frac{1}{2\sigma^2}$ to simplify the expression as,

$$\begin{aligned} &= e^{\left(-\frac{1}{2\sigma^2}\left[(x_1-\mu)^2 + (x_2-\mu)^2 + \ldots + (x_n-\mu)^2\right]\right)} \\ &= e^{\left(-\frac{1}{2\sigma^2}\sum_i^n (x_i-\mu)^2\right)} \end{aligned} \tag{10}$$

Knowing all of this, we can express the *joint probability* of all our observations using the *Gaussian* distribution function as,

$$\begin{aligned} f(x_1, x_2, \ldots, x_n|\mu, \sigma^2) &= \textstyle\prod_i^n (2\pi\sigma^2)^{-\frac{1}{2}} e^{\left(-\frac{1}{2\sigma^2}(x_i-\mu)^2\right)} \\ &= (2\pi\sigma^2)^{-\frac{n}{2}} e^{\left(-\frac{1}{2\sigma^2}\sum_i^n (x_i-\mu)^2\right)}. \end{aligned} \tag{11}$$

But as you can imagine, if the probabilities are less than 1, then the product of a bunch of these is going to be **SUPER** small. It's not that big of a deal for the math, at least symbolically, but dealing with repeated multiplication of small things is *tedious* and *error prone*, for both humans and computers alike. Practically speaking, a computer has a limit on how small it can represent things and still be accurate.

Fortunately, we may, or may not, remember a special property of *logs*, that the *log* function can turn a product into sum—this will be illustrated below. So, by taking the *log* of the probability function we can make the computation much easier while still keeping the same functional relations among the parameter in our original probability function.

Quick and dirty logs. Just a refresher, *logs* are meant to show the number of times a number, the *base*, is to be multiplied by itself to get a particular value. As the YouTuber ***Vihart*** put it, if we were counting in a "times the base sort of way",[1] how many steps would we need to go to get to the answer. So, the answer of the *log* function represents what power of the *base* is needed to get the input value. For example, if the base is 10, and input value is 10, then the answer of the *log* function is 1, because $10^1 = 10$, and so $log_{10}(10) = 1$. Additionally, counting in a "times ten" sort of way, how many steps to get to 100? The answer is 2.

In order to show some of the other properties of *logs* we will work with an easy example. We will use 100, which can be expressed the following **equivalent** ways.

$$\begin{aligned} 100 &= 10^2 \\ &= 10 \times 10 \\ &= 1000/10 \end{aligned} \tag{12}$$

So, let's work with *log* with a base of 10, this means we are interested in what power to raise 10 to in order to produce the result of 100.

$$\begin{aligned} \text{if} \quad & 10^2 = 100 \\ \text{then} \quad & log_{10}(100) = 2 \end{aligned} \tag{13}$$

As we can see, 2 is the answer for base 10. Below we present 3 of the basic properties of *logs*. These are not all of the properties, just the ones that are important for our illustration.

We assume base 10 for the following rules:

power rule

$log(A^n) = n \times log(A)$

- $log(10^2) = 2 \times log(10) = 2 \times 1 = 2$

product rule

$log(A \times B) = log(A) + log(B)$

- $log(10 \times 10) = log(10) + log(10) = 1 + 1 = 2$

quotient rule

$log(\frac{A}{B}) = log(A) - log(B)$

- $log(\frac{1000}{10}) = log(1000) - log(10) = 3 - 1 = 2$

Log likelihood derivation

So, why does this matter? Well, because we are interested in fitting our previous function of the likelihood of a set of data, but we don't want to cause our computer to start to smoke computing very small numbers. If we take the *log* of the likelihood function we get another function that preserves our main properties, but that will also turn our product into a sum.

[1] Check out Vihart's video "How I Feel About Logarithms" for a great explanation.

We will take the *log* of this joint probability version from above. In this case it is easiest to use a base of e for the log of the likelihood, or *natural log*, ln which equals log_e—and remember, this means $ln(e) = 1$. This makes the exponential part much easier to understand. Here are the steps for expressing the new log-likelihood function,

$$\begin{aligned} ln(f(x_1, x_2, \ldots, x_n|\mu, \sigma^2)) &= ln\left[(2\pi\sigma^2)^{-\frac{n}{2}} e^{\left(-\frac{1}{2\sigma^2}\sum_i^n (x_i-\mu)^2\right)}\right] \\ \text{by the } \textbf{product rule} &= ln\left[(2\pi\sigma^2)^{-\frac{n}{2}}\right] + ln\left[e^{\left(-\frac{1}{2\sigma^2}\sum_i^n (x_i-\mu)^2\right)}\right] \\ \text{by the } \textbf{power rule} &= \left[(-\tfrac{n}{2})\, ln(2\pi\sigma^2)\right] + \left[\left(-\tfrac{1}{2\sigma^2}\textstyle\sum_i^n (x_i-\mu)^2\right) ln(e)\right] \end{aligned} \tag{14}$$

simplify and we get

$$\mathcal{L}(X|\mu, \sigma^2) = -\left(\tfrac{n}{2}\right) ln(2\pi\sigma^2) - \tfrac{1}{2\sigma^2}\textstyle\sum_i^n (x_i-\mu)^2$$

Minus 2 of the log of the likelihood

$$-2\mathcal{L}(X|\mu, \sigma^2) = n(ln(\pi\sigma^2)) + \frac{1}{\sigma^2}\sum_{i=1}^{n}(x_i-\mu)^2 \tag{15}$$

Maximum Likelihood

Analytic solution

In this section we will work to solve for the specific parameters that will maximize our observed data. Again we are basing this on the distribution that we believe generated our data, in this case the Gaussian probability function. Below we need to solve for the parameters μ and σ^2 in terms of the observed data $X \in \{x_i, x_2, \ldots, x_n\}$.

To do this we will use calculus to find the maximum of the above function with regard to each parameter. First we will express the function in terms of the specific parameter, then take the derivative of the function with respect to the parameter to isolate its influence on the function overall. This step helps us understand how the function changes with respect to the parameter of interest. We set the partial derivative equal to zero and solve for the parameter to get where the changes in the function reach a maximum.

A quick note about derivatives. This next section assumes that you have some familiarity with the ideas of differentiation. If you do not, fear not, the examples below are pretty simple once you understand what exactly is going on. In short, derivatives communicate how a funciton changes. So, imagine a line, let's say defined by the following equation,

$$y = 0.15x + 3.$$

As we can see, the slope is 0.15 and represents how the overall function—here describing y—changes for a unit change in x. If we just focused on how the function changes,

we see that it does not differ no matter what x value we supply, the diffference is always 0.15. In calculus terms, the first derivative of the function above is 0.15.

This is different for a more complex model like,

$$y = 3x^2 + 4.$$

In this case we have a parabola, and the changes are not the same for every value of x. In some cases the changes are going in the negative direction, in other they are going in the positive direction, at the turning point, the changes are flat! The point here is that if we focus on the changes, we will see that they are dependent on the value of x that you supply, thus if we are trying to decribe the change, we would expect it to involve the value or x. As it turns out there is a rule called the *power rule* that describes how to figure out the derivative for this sort of equation. In short we take the power of the variable x, multiply it by the existing coefficient then reduce the power by 1. This would mean that in our parabola example, the derivative would be

$$3(2) \times x^{2-1} = 6x,$$

Notice that the constant 4 was not included, why? Well because it doesn't influence x in any direct way. So it can be safely omitted.

Here is the general formula

$$\frac{d}{dx} = n(x^{n-1})$$

Now, we will see partial derivatives below. In this case we are only intersted in doing differentiation for terms in our equation that include the parameter of interest.

Admittedly, there is WAY more to differentiation than this, but above is the essential information for what is to follow.

Partial derivative wrt μ. So, let's start by restating our loglikelihood specification in equation 15.

$$-2\mathcal{L}(X|\mu, \sigma^2) = n(ln(\pi\sigma^2)) + \frac{1}{\sigma^2}\sum_{i=1}^{n}(x_i - \mu)^2$$

Now, only focusing on the terms that involve the μ parameter we get,

$$\begin{aligned}
\mathcal{L} \text{ wrt } \mu &= \tfrac{1}{\sigma^2}\textstyle\sum_{i=1}^{n}(x_i - \mu)^2 \\
&= \tfrac{1}{\sigma^2}\left[\textstyle\sum_{i=1}^{n} x_i^2 - 2x_i\mu + \mu^2\right] \\
&= \tfrac{1}{\sigma^2}\left[\textstyle\sum_{i=1}^{n} x_i^2 - 2\sum_{i=1}^{n} x_i\mu + \sum_{i=1}^{n}\mu^2\right] \qquad (16) \\
&= \tfrac{1}{\sigma^2}\left[\textstyle\sum_{i=1}^{n} x_i^2 - 2\mu\sum_{i=1}^{n} x_i + n\mu^2\right] \\
\tfrac{\partial\mathcal{L}}{\partial\mu} &= \tfrac{1}{\sigma^2}\left[-2\textstyle\sum_{i=1}^{n} x_i + 2n\mu\right]
\end{aligned}$$

Set the result equal to zero and solve

$$\begin{aligned} 0 &= \tfrac{1}{\sigma^2}\left[-2\textstyle\sum_{i=1}^{n} x_i + 2n\mu\right] \\ &= -2\textstyle\sum_{i=1}^{n} x_i + 2n\mu \\ 2\textstyle\sum_{i=1}^{n} x_i &= 2n\mu \\ \tfrac{\sum_{i=1}^{n} x_i}{n} &= \mu \end{aligned} \tag{17}$$

Partial derivative wrt σ^2

Substitute $\sigma^2 = u$

$$\begin{aligned} \mathcal{L} \text{ wrt } u &= n(ln(\pi u)) + \tfrac{1}{u}\textstyle\sum_{i=1}^{n}(x_i-\mu)^2 \\ &= n\left[ln(\pi) + ln(u)\right] + \tfrac{1}{u}\textstyle\sum_{i=1}^{n}(x_i-\mu)^2 \\ &= nln(\pi) + nln(u) + \tfrac{1}{u}\textstyle\sum_{i=1}^{n}(x_i-\mu)^2 \\ \tfrac{\partial \mathcal{L}}{\partial u} &= n\left(\tfrac{1}{u}\right) - \tfrac{1}{u^2}\textstyle\sum_{i=1}^{n}(x_i-\mu)^2 \end{aligned} \tag{18}$$

Set the result equal to zero and solve

$$\begin{aligned} 0 &= n\left(\tfrac{1}{u}\right) - \tfrac{1}{u^2}\textstyle\sum_{i=1}^{n}(x_i-\mu)^2 \\ \tfrac{1}{u^2}\textstyle\sum_{i=1}^{n}(x_i-\mu)^2 &= n\left(\tfrac{1}{u}\right) \\ \tfrac{\sum_{i=1}^{n}(x_i-\mu)^2}{u^2} &= \tfrac{n}{u} \\ \textstyle\sum_{i=1}^{n}(x_i-\mu)^2 &= \tfrac{nu^2}{u} \\ \textstyle\sum_{i=1}^{n}(x_i-\mu)^2 &= nu \\ \tfrac{\sum_{i=1}^{n}(x_i-\mu)^2}{n} &= u \end{aligned} \tag{19}$$

back substitute for σ^2

$$\tfrac{\sum_{i=1}^{n}(x_i-\mu)^2}{n} = \sigma^2$$

Regression through Calculus with Partial Derivatives

Minimizing a loss function

In this section we will explore the use of this same basic approach to solve for the parameters in a different equation. In this case we will look to solve for the parameters of a linear model expressed as,

$$y = b_0 + b_1 x, \tag{20}$$

where b_0 is the y intercept and b_1 is the slope.

Our goal here is to solve for each of these parameters and express these solutions in terms of our observed data $y_i \in \{y_1, y_2, \ldots, y_n\}$ and $x_i \in \{x_1, x_2, \ldots, x_n\}$.

Loss function

Now, if we are interested in solving for the parameters in equation 20, we need observations. If we have as many observations as we have parameters to solve, in our models there are 2, b_0 and b_1 respectively, there is assumed to be one unique solution that will satisfy the equation. This is exactly the same task as solving simultaneous equation from Algebra, keep in mind that each observation is thought to follow the same model in equation 20. For example if we had two observations $(x_1, y_1) = (0, 3)$ and $(x_2, y_2) = (1, 3.5)$ we could construct the following lines,

$$\begin{array}{rl} 3 & = b_0 + b_1 0 \\ 3.5 & = b_0 + b_1 1 \end{array}. \tag{21}$$

Solving for each parameter above we would find that $b_0 = 3$ and $b_1 = .5$.

The situation becomes more difficult when we have more observations than we have parameters. In practice this is most common, but what we need to do is find a solution that is "close enough" to the data that we are satisfied. We will call the values that are computed from this "close enough" solution our estimates and designate them as $\hat{y}_i$. These values represent a direct application of our model in equations 20,

$$\hat{y}_i = b_0 + b_1 x_{1i} \tag{22}$$

Our estimates however are not perfect, and most times will be off by some amount. This amount will be different for each estimate we generate, so it's helpful to think about capturing this error. We will define our error term as,

$$\begin{array}{l} y_i = \hat{y}_i + \epsilon_{y_i} \\ \epsilon_{y_i} = y_i - \hat{y}_i \end{array}. \tag{23}$$

Since our goal is to find the parameters, b_0 and b_1 that are "close enough" we will try to minimize the errors expressed in equation 23. Specifically, we will look to minimize the variance of these errors.

Error variance. To compute the variance of the errors we need to start with the sum of squared errors,

$$\sum_{i=1}^{n} \epsilon_i^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2. \tag{24}$$

Substituting equation 20 for our $\hat{y}_i$ term we get,

$$\begin{aligned}
\textstyle\sum_{i=1}^{n} \epsilon_i^2 &= \textstyle\sum_{i=1}^{n} (y_i - b_0 - b_1 x_i)^2 \\
&= \textstyle\sum_{i=1}^{n} (y_i - b_0 - b_1 x_i)(y_i - b_0 - b_1 x_i) \\
&= \textstyle\sum_{i=1}^{n} (y_i^2 - b_0 y_i - b_1 x_i y_i) \\
&\quad + (-y_i b_0 + b_0^2 + b_0 b_1 x_i) \\
&\quad + (-y_i b_1 x_i + b_0 b_1 x_i + b_1^2 x_i^2) \\
&= \textstyle\sum_{i=1}^{n} (y_i^2 + b_0^2 + b_1^2 x_i^2) \\
&\quad + (-y_i b_0 - y_i b_0) \\
&\quad + (-b_1 x_i y_i - b_1 x_i y_i) \\
&\quad + (b_0 b_1 x_i + b_0 b_1 x_i) \\
&= \textstyle\sum_{i=1}^{n} (y_i^2 + b_0^2 + b_1^2 x_i^2 - 2 y_i b_0 - 2 b_1 x_i y_i + 2 b_0 b_1 x_i)
\end{aligned} \tag{25}$$

We distribute the summation operator to get,

$$\begin{aligned}
&\textstyle\sum_{i=1}^{n} (y_i^2 + b_0^2 + b_1^2 x_i^2 - 2 y_i b_0 - 2 b_1 x_i y_i + 2 b_0 b_1 x_i) \\
&\textstyle\sum_{i=1}^{n} y_i^2 + \sum_{i=1}^{n} b_0^2 + \sum_{i=1}^{n} b_1^2 x_i^2 \\
&\qquad - \textstyle\sum_{i=1}^{n} 2 y_i b_0 - \sum_{i=1}^{n} 2 b_1 x_i y_i + \sum_{i=1}^{n} 2 b_0 b_1 x_i \\
&\textstyle\sum_{i=1}^{n} (y_i^2) + n b_0^2 + b_1^2 \sum_{i=1}^{n} (x_i^2) \\
&\qquad - 2 b_0 \textstyle\sum_{i=1}^{n} (y_i) - 2 b_1 \sum_{i=1}^{n} (x_i y_i) + 2 b_0 b_1 \sum_{i=1}^{n} (x_i)
\end{aligned} \tag{26}$$

We will simplify the above expression by making the following substitutions,

- $SS_y = \sum_{i=1}^{n} (y_i^2)$
- $SS_x = \sum_{i=1}^{n} (x_i^2)$
- $SCP_{xy} = \sum_{i=1}^{n} (x_i y_i)$
- $SS_\epsilon = \sum_{i=1}^{n} \epsilon_i^2$.

Using these definitions our expression in equation 26 becomes,

$$SS_\epsilon = SS_y + n b_0^2 + b_1^2 SS_x - 2 b_0 \sum_{i=1}^{n} (y_i) - 2 b_1 SCP_{xy} + 2 b_0 b_1 \sum_{i=1}^{n} (x_i). \tag{27}$$

Solving for the parameters

From here we will perform our partial differentiation in order to solve for the parameters b_0 and b_1.

The Intercept b_0.

$$SS_e \text{ w.r.t. } b_0 = nb_0^2 - 2b_0 \sum_{i=1}^{n}(y_i) + 2b_0 b_1 \sum_{i=1}^{n}(x_i) \tag{28}$$

Next we take the partial derivative of the above with respect to b_0, set it to zero and solve for b_0,

$$\begin{aligned} \frac{\partial SS_e}{\partial b_0} &= 2nb_0 - 2\textstyle\sum_{i=1}^{n}(y_i) + 2b_1 \sum_{i=1}^{n}(x_i) \\ 0 &= 2nb_0 - 2\textstyle\sum_{i=1}^{n}(y_i) + 2b_1 \sum_{i=1}^{n}(x_i) \\ 2nb_0 &= 2\textstyle\sum_{i=1}^{n}(y_i) - 2b_1 \sum_{i=1}^{n}(x_i) \\ b_0 &= \frac{2\sum_{i=1}^{n}(y_i) - 2b_1 \sum_{i=1}^{n}(x_i)}{2n} \\ b_0 &= \frac{\sum_{i=1}^{n}(y_i)}{n} - b_1 \frac{\sum_{i=1}^{n}(x_i)}{n} \end{aligned} \tag{29}$$

We can use the fact that $\bar{x} = \frac{\sum_{i=1}^{n}(x_i)}{n}$ and $\bar{y} = \frac{\sum_{i=1}^{n}(y_i)}{n}$, thus turning the result of equation 29 into,

$$b_0 = \bar{y} - b_1 \bar{x} \tag{30}$$

The Slope b_1.

$$SS_e \text{ w.r.t. } b_1 = b_1^2 SS_x - 2b_1 SCP_{xy} + 2b_0 b_1 \sum_{i=1}^{n}(x_i). \tag{31}$$

Next we take the partial derivative of the above with respect to b_1, set it to zero and solve for b_1.

$$\frac{\partial SS_e}{\partial b_1} = 2b_1 SS_x - 2SCP_{xy} + 2b_0 \sum_{i=1}^{n}(x_i) \tag{32}$$

Before we do the algebra to find the solution, we will take advantage of our solution for b_0 in equation 30 and use this to transform all terms in our expression in equation 32 to be expressed as functions of the b_1 parameter,

$$\begin{aligned} 0 &= 2b_1 SS_x - 2SCP_{xy} + 2(\frac{\sum_{i=1}^{n}(y_i)}{n} - b_1 \frac{\sum_{i=1}^{n}(x_i)}{n}) \textstyle\sum_{i=1}^{n}(x_i) \\ &= 2b_1 SS_x - 2SCP_{xy} + \frac{2}{n}\textstyle\sum_{i=1}^{n}(y_i x_i) - b_1 \frac{2}{n}\sum_{i=1}^{n}(x_i x_i) \\ &= 2b_1 SS_x - 2SCP_{xy} + \frac{2}{n} SCP_{xy} - b_1 \frac{2}{n} SS_x \end{aligned} \quad . \tag{33}$$

Move all common terms to one side

$$-2b_1 SS_x + b_1 \frac{2}{n} SS_x = -2SCP_{xy} + \frac{2}{n} SCP_{xy} \ . \tag{34}$$

Next, multiply through by $-\frac{1}{2}$

$$\begin{aligned} b_1 SS_x - b_1 \tfrac{1}{n} SS_x &= SCP_{xy} - \tfrac{1}{n} SCP_{xy} \\ b_1 (SS_x - \tfrac{1}{n} SS_x) &= (SCP_{xy} - \tfrac{1}{n} SCP_{xy}) \\ b_1 [SS_x (1 - \tfrac{1}{n})] &= [SCP_{xy} (1 - \tfrac{1}{n})] \\ b_1 &= \tfrac{SCP_{xy}(1-\frac{1}{n})}{SS_x(1-\frac{1}{n})} \\ b_1 &= \tfrac{SCP_{xy}}{SS_x} \times \tfrac{(1-\frac{1}{n})}{(1-\frac{1}{n})} \\ b_1 &= \tfrac{SCP_{xy}}{SS_x} \end{aligned} \tag{35}$$

Thus our maximum likelihood estimates for the parameters b_0 and b_1 from equation 20 are,

$$b_0 = \bar{y} - b_1 \bar{x} \tag{36}$$

and

$$b_1 = \frac{SCP_{xy}}{SS_x} \tag{37}$$

Why do we care? Well, another way to think about regression coefficients are as the partial derivatives with respect to each input. So in the equation,

$$GPA_{HS} \sim b_0 + b_1 ED_{mom} + \epsilon$$

The b_1 coefficient is equal to

$$\frac{\partial GPA_{HS}}{\partial ED_{mom}} = b_1,$$

So for an increase of one unit in a mother's level of education there would be a corresponding increase of b_1 in GPA_{HS}. This type of regression is sometimes referred to as *Level-Level* regression, because, a change in the level of the input x results in a change in the level of the output y, while holding everything else (that is the other inputs) constant. We will see other types of regression below, but first a bit of a review.

Nonlinear models

From here out, we will be looking at nonlinear trends and some of the ways that we approach modeling them. It is important to keep in mind that one of the major assumptions of regression is that the variables are linearly related. For the most part we will be trying to transform relations of the form,

$$y = x^{\beta} + \epsilon \tag{38}$$

into something that look more like the lines that we know, specifically $y = ax + b$. Figure 1 illustrates three possible trajectories for different powers.

Log-Level regression example

For this model the outcome will be transformed in order to make the model linear. In particular the hypothesized model takes the the form

$$y \sim \alpha e^{\beta x}. \tag{39}$$

Based on our knowledge of logs, in particular the **power rule** from above, we can take the log of both sides of Equation 39 to get [2],

$$\ln(y) \sim \ln(\alpha) + \beta x. \tag{40}$$

Table 1 contains example data that will be used to present this model [3].

Table 1
Log-level example data

	x	y
1	45.00	33.00
2	99.00	72.00
3	31.00	19.00
4	57.00	27.00
5	37.00	23.00
6	85.00	62.00
7	21.00	24.00
8	64.00	32.00
9	17.00	18.00
10	41.00	36.00
11	103.00	76.00

Table 2
Log-level regression results

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	2.64	0.12	21.83	0.00
x	0.02	0.00	8.19	0.00

Now the question is how to interpret the resulting estimates. At first glace we are dealing with log changes in the outcome y for corresponding unit changes in x. This is where the term *Log-Level* comes from, and put simply, we can expect a 0.02% change in y for a 1 unit change in x.

A better understanding, at least in terms of raw scale units can be gained from *back transformation*. In particular, we must exponentiate our estimates if we want to get back to raw score levels. Thus, our equation estimates

$$\ln(y) \sim 2.64 + 0.02x, \tag{41}$$

[2] Here we take the natural log, or log base e.
[3] These example data, and others from http://www.real-statistics.com/regression/

would need to be transformed back as

$$\begin{aligned} y &\sim e^{2.64+0.02x} \\ &\quad e^{2.64} \times e^{0.02x} \end{aligned} \tag{42}$$

$$y \sim 14.0132 \times 1.0202^{x} \tag{43}$$

Now, the intercept term is the expected level of $\ln(y)$ when $x = 0$. In our equation above, the value is 14.0132, however the mean of our outcome is actually 38.3636. Let's see what happens when we mean center our predictor.

Table 3
Log-level with mean centered predictor

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	3.52	0.06	62.55	0.00
x_c	0.02	0.00	8.19	0.00

Based on this model the expected mean of the outcome is 33.7982. This is closer to our reported mean of 38.3636 but not exact... why? This is because here we are dealing with what is called the *Geometric* mean, rather than the mean we normally use. The *Geometric* mean is computed as,

$$\left(\prod_{i=1}^{N} x_i \right)^{1/N}. \tag{44}$$

When we compute the geometric mean of y we get 33.7982, which matches our estimate based on the model. Figure 2 compares the untransformed trajectory and the transformed trajectory.

Log-Log regression example

In this model, both the outcome and the predictor are log-transformed. That is because the predictor is raised to the power of a parameter, specifically

$$y = \alpha x^{\beta} \tag{45}$$

Thus, taking the log of both sides results in,

$$\ln(y) = \ln(\alpha) + \beta \ln(x) \tag{46}$$

Parameter interpretation

Interpretation of the model estimates in Table 5 is pretty straight-forward. We are dealing with percent changes in both the outcome and the predictor. In economics this is referred to as *elasticity*. So, based on the model a 1% change in x would result in an 0.23% change in y. Figure 3 compares the untransformed trajectory and the transformed trajectory.

Table 4
Log-Log example data

	x	y
1	8.10	33.00
2	69.90	49.00
3	4.20	19.00
4	14.10	27.00
5	5.60	23.00
6	52.10	51.00
7	44.60	34.00
8	19.60	32.00
9	33.00	28.00
10	6.70	36.00
11	30.10	43.00

Table 5
Log-Log regression results

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	2.81	0.21	13.65	0.00
log(x)	0.23	0.07	3.44	0.01

A caution about nonlinear transformations

Above, we've discussed power-*ish* transformations, notice that in Figure 1 the basic equation was $y = x^{\beta}$. The transformations just illustrated only really make sense if the ratio of largest to smallest value on the raw scale is large. If it's not, then something like the natural log will have little effect on the relation.

Also, if the values are negative, it will be necessary to add a constant before taking the log of the values.

Lastly, these transformations should only be used when the relationship is *monotonic*, meaning it passes the horizontal line test, or is a *one-to-one* function. This means that quadratic trends or trends that oscillate are not good candidates for transformation. When relations are *monotonic* these transformations will not change the *rank-order* of the observations, just the spaces in between successive values.

Proportions and the Binomial models

There is a special case involved around **binary** outcomes. In general power transformations don't work well when the data values are near 0 or 1, which is exactly the case for binary data. Think about coding *pass-fail* or *True-False* outcomes. In this case we need to develop a way in which to transform these 0/1 values into something manageable.

The Logistic curve. We will be using a function called the *logistic curve* which has the functional specification of,

$$y = \frac{e^{\theta}}{1 + e^{\theta}} \tag{47}$$

Where θ represents all of the possible inputs of interest for predicting y. From our *level-level* regression example, *theta* may be equal to $\beta_0 + \beta_1 x_1 + \ldots + \beta_k x_k$ thus making the equation

$$y = \frac{e^{(\beta_0+\beta_1 x_1+\ldots+\beta_k x_k)}}{1 + e^{(\beta_0+\beta_1 x_1+\ldots+\beta_k x_k)}} \tag{48}$$

With this approach we are modeling proportions. So, instead of trying to use either 0 or 1 as an outcome directly, we will be looking at the total number of 1s out of all responses. A little later we will specify this as $Pr(Y = 1)$ or the probability of scoring a 1 on the outcome. For binary data this follows our logistic curve.

However, we can still model proportions directly as well, as shown below.

Dose-Response example

These data are a reproduction of data from:

C.I. Bliss (1935). The calculation of the dosage-mortality curve. *Annals of Applied Biology, 22*(1), 134-167. [4]

The Data Beetles were exposed to carbon disulphide at varying concentrations for 5 hours.

- dose = mf/L concentration of CS2
- nexp = number of beetles exposed
- ndied = number of beetles killed
- prop = proportion of dead to exposed beetles

Table 6
Beetle data

	dose	nexp	ndied	prop	nalive
1	49.10	59.00	6.00	0.10	53.00
2	53.00	60.00	13.00	0.22	47.00
3	56.90	62.00	18.00	0.29	44.00
4	60.80	56.00	28.00	0.50	28.00
5	64.80	63.00	52.00	0.82	11.00
6	68.70	59.00	53.00	0.90	6.00
7	72.60	62.00	61.00	0.98	1.00
8	76.50	60.00	60.00	1.00	0.00

The Logistic Model

Run a logistic regression of the proportion of dead to living beetles as a function of the dose of CS2 gas. Our model specification is,

$$\frac{n_{died}}{n_{alive}} \sim b_0 + b_1 dose \tag{49}$$

Table 7
Logistic model results

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-14.8230	1.2896	-11.49	0.0000
dose	0.2494	0.0214	11.66	0.0000

we may be interested in finding the concentration of CS2 gas that is lethal 50% of the time, the **LD50**.

Note that if we have a function with multiple predictors we can solve for each variable using something similar. For example if,

$$y \sim b_0 + b_1(x_1) + b_2(x_2) + b_3(x_3)$$

is the model. Then to find a specific value for one of the predictors $(x1, x2, x3)$ that corresponds to a desired probability y.

- $x_1 = (-b_0 - b_2 - b_3 + log\left(\frac{-y}{(y-1)}\right))/b1$
- $x_2 = (-b_0 - b_1 - b_3 + log\left(\frac{-y}{(y-1)}\right))/b2$
- $x_3 = (-b_0 - b_1 - b_2 + log\left(\frac{-y}{(y-1)}\right))/b3$

[4] Many thanks to Thaddeus Tarpey at Wright University. Check out his cite for this and more http://www.wright.edu/ thaddeus.tarpey/

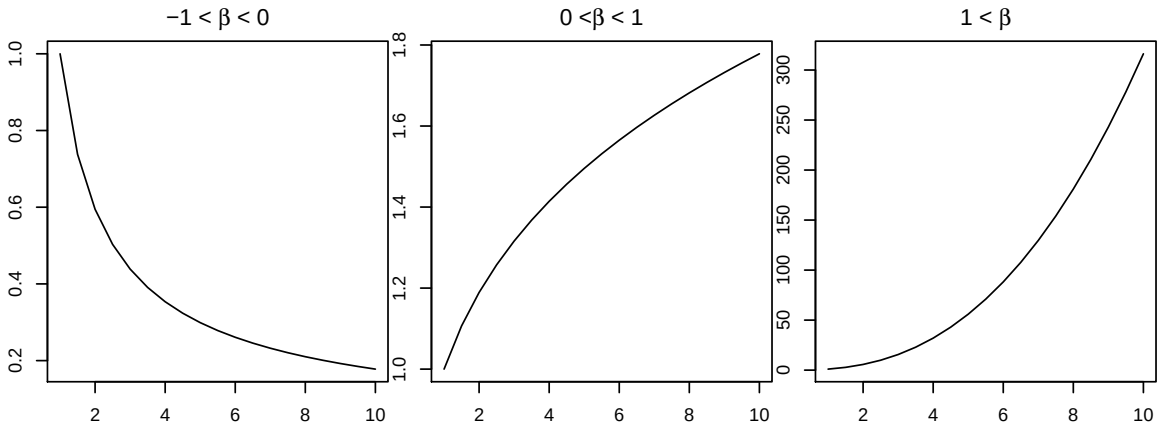

Figure 1. Nonlinear trajectories for different powers

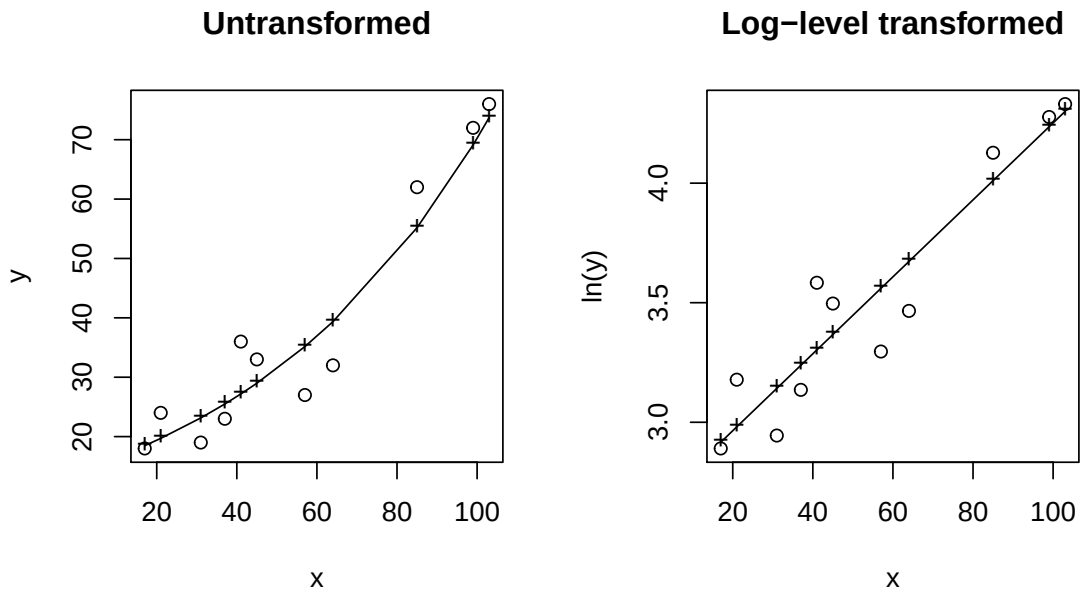

Figure 2. Log-level Observed vs predicted

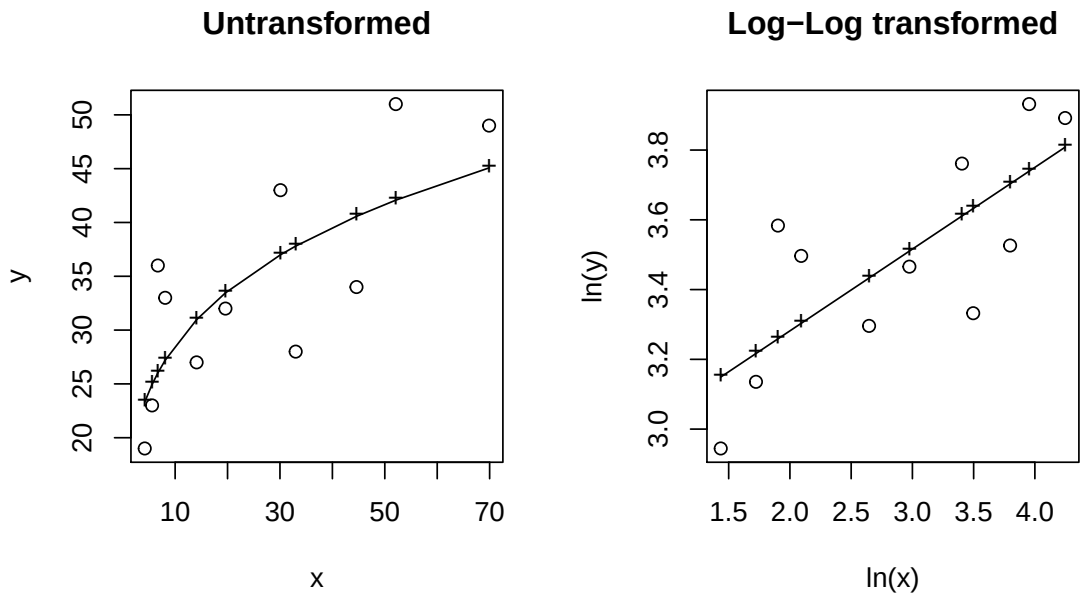

Figure 3. Log-Log Observed vs predicted

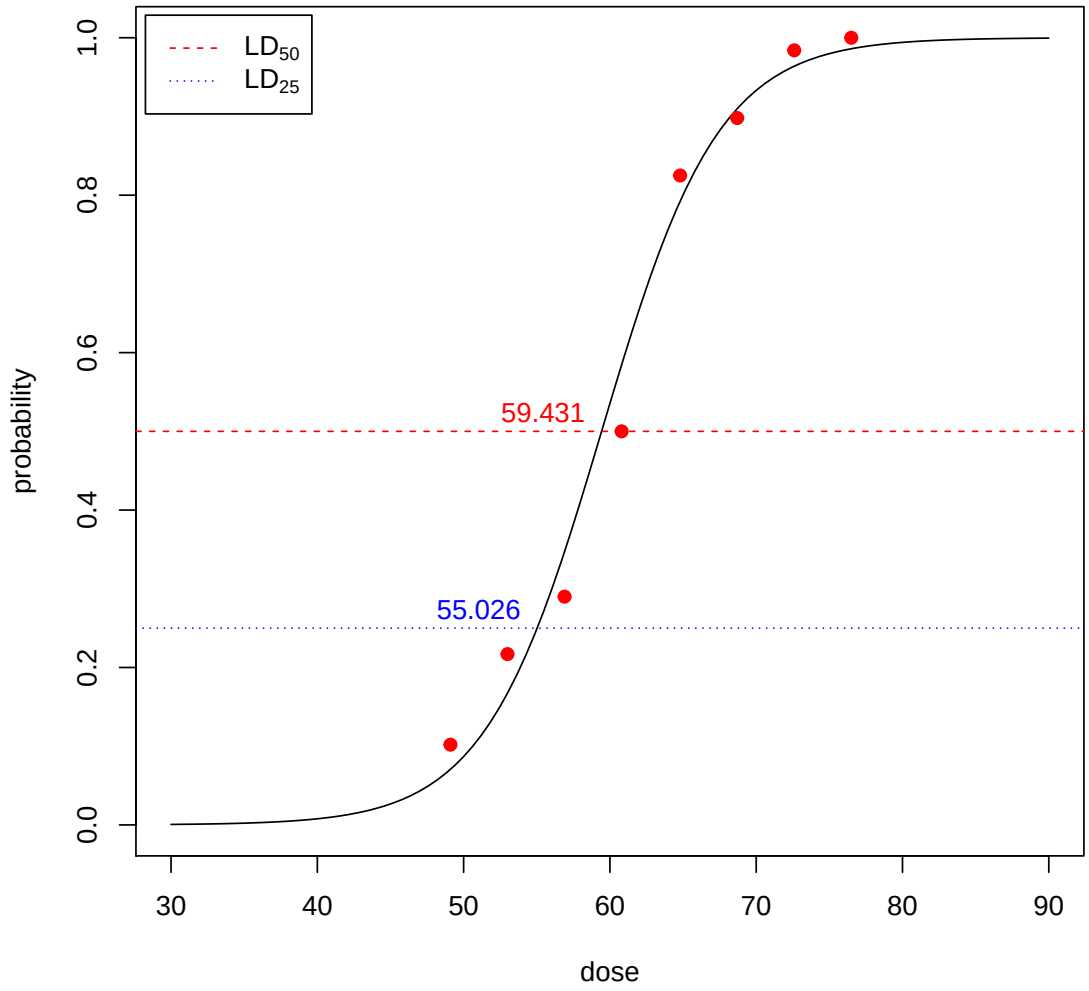

Figure 4. Example Dose-Response curve

Below is a table to help you understand the different types of transformations available and how to interpret them.

Name	Outcome	Input	Form	β_1	interpretation
Level-Level	Y	X	$y \sim \beta_0 + \beta_1 x + \epsilon$	$\Delta y = \beta_1 \Delta x$	1 unit change in x give β_1 unit change in y
Level-Log	Y	$ln(X)$	$y \sim \beta_0 + \beta_1 ln(x) + \epsilon$	$\Delta y = \beta_1 \%\Delta x$	1% change in x give β_1 unit change in y
Log-Level	$ln(Y)$	X	$ln(y) \sim \beta_0 + \beta_1 x + \epsilon$	$\%\Delta y = \beta_1 \Delta x$	1 unit change in x gives β_1% change in y
Log-Log	$ln(Y)$	$ln(X)$	$ln(y) \sim \beta_0 + \beta_1 ln(x) + \epsilon$	$\%\Delta y = \beta_1 \%\Delta x$	1% change in x give β_1% change in y

(50)