

# NON-LINEAR OLS

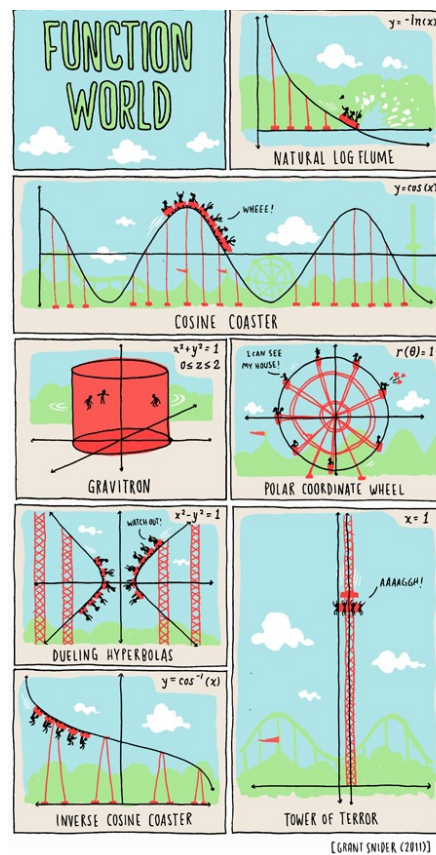
Research Methods in Psychology I & II • Department of Psychology • Colorado State University

## BY THE END OF THIS UNIT YOU WILL:

1. Understand that nonlinear relationships can be fit within an OLS framework.
2. Know how to apply non-linear transformation(s) to  $x$ ,  $y$ , or both to achieve linearity.
3. Know how to apply polynomial regression to model non-linear relationships between a particular  $x$  variable and an outcome.

## Can we model a non-linear relationship in an OLS framework?

Yes! We will explore two different techniques that can be used to model a non-linear relationship in an OLS regression. The first involves non-linear transformation of  $x$ ,  $y$ , or perhaps both. The second involves adding squared, cubic, and even higher order terms for certain  $x$  variables to allow for a curvilinear relationship between  $x$  and  $y$ .



## Prepare a New Notebook

Create a new notebook called `Nonlinear_Notebook` in your `MyClassActivities` Folder. Then add the following code chunk.

```
---  
title: "R Notebook for Nonlinear OLS regression"  
output: html_notebook  
---  
  
# Part 1: Nonlinear OLS models via variable transformation  
  
## Load libraries  
```{r, message = FALSE}  
  
library(tidyverse)  
library(olsrr)  
library(modelr)  
library(psych)  
library(ggally)  
```
```

## A Brief Primer on Transformations

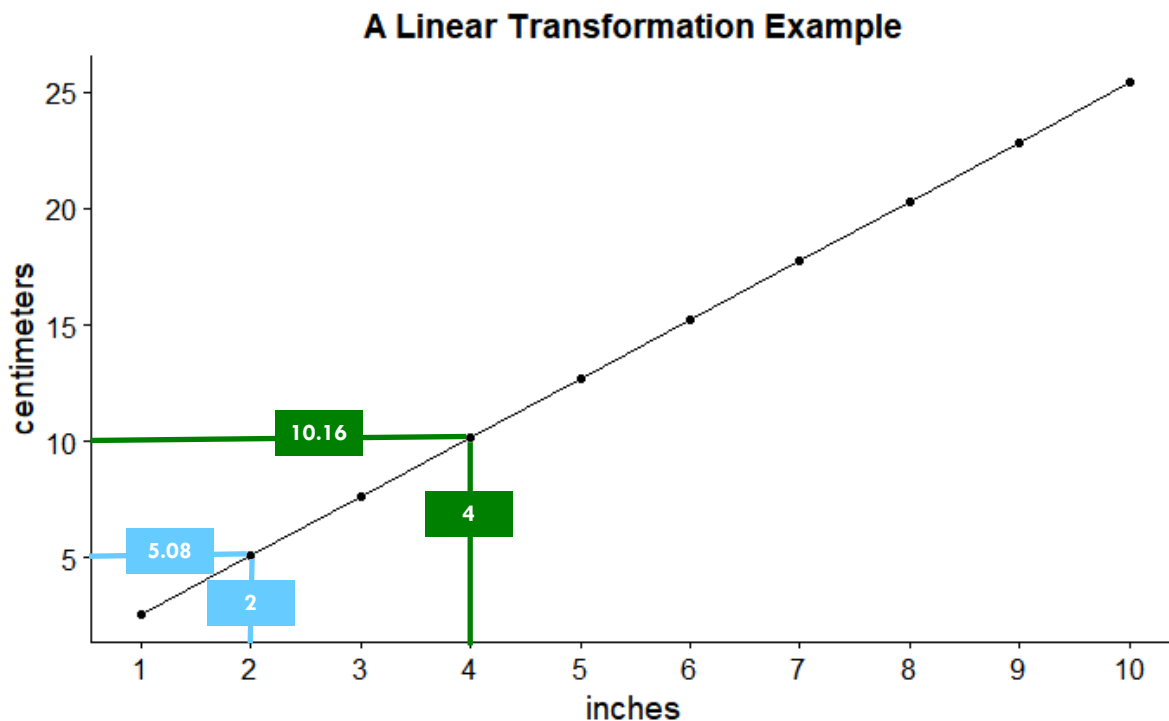
Transformations are a common part of everyday life—for example, we often transform U.S. Dollars into another currency (i.e., Pounds), a temperature in Celsius to a temperature in Fahrenheit, or inches into centimeters. Let's look at a chart that transforms inches (x-axis) to centimeters (y-axis).

Linear transformation

```
trans1 <- tibble(inches = 1:10, centimeters = inches*2.54)

ggplot(trans1, aes(x = inches, y = centimeters)) +
  geom_line() +
  geom_point() +
  scale_x_continuous(breaks = c(1:10)) +
  labs (title = "A Linear Transformation Example")
```

| inches | centimeters |
|--------|-------------|
| 1      | 2.54        |
| 2      | 5.08        |
| 3      | 7.62        |
| 4      | 10.16       |
| 5      | 12.70       |
| 6      | 15.24       |
| 7      | 17.78       |
| 8      | 20.32       |
| 9      | 22.86       |
| 10     | 25.40       |



Notice that this is a linear transformation. We know this because the original and transformed values can be joined by a straight line. Notice that linear transformations preserve relative spacing. That is, values that are evenly spaced before transformation remain evenly spaced after transformation. For example, values that are spaced twice as far apart as other values before transformation remain twice as far apart after transformation.

## Logarithmic Transformations—Base 2 Logarithm

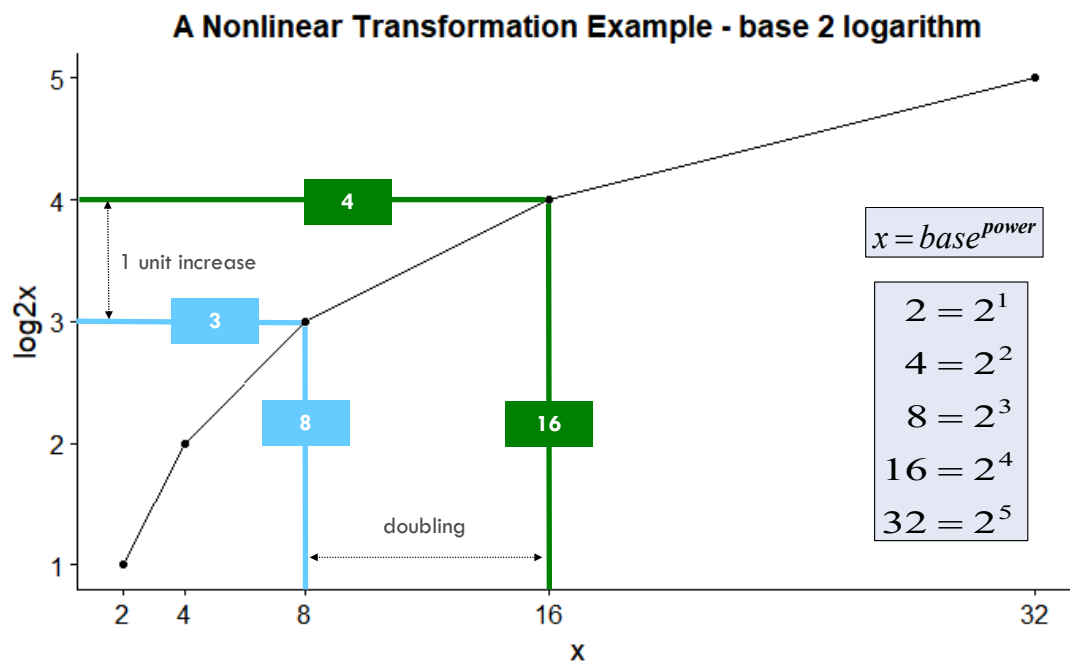
Non-linear transformations are also possible. With a nonlinear transformation, the initial and transformed values DO NOT fall on a straight line and nonlinear transformations DO NOT preserve spacing. A logarithm is an example of a non-linear transformation, and it is one of the most common transformations that we use in statistics.

Nonlinear transformation—base 2 logarithm

```
trans2 <- tibble(x = c(2,4,8,16,32), log2x = log2(x))
```

```
ggplot(trans2, aes(x = x, y = log2x)) +  
  geom_line() +  
  geom_point() +  
  scale_x_continuous(breaks = c(2,4,8,16,32)) +  
  labs (title = "A Nonlinear Transformation Example - base 2 logarithm")
```

| x  | log2x |
|----|-------|
| 2  | 1     |
| 4  | 2     |
| 8  | 3     |
| 16 | 4     |
| 32 | 5     |



This chart depicts a base 2 logarithm. The logarithm is the power (i.e., exponent) to which a base must be raised in order to produce a number. Therefore,  $\log_2(8) = 3$ , because 2 must be raised to a power of 3 to get 8.

Notice that each 1 unit increase in a base 2 logarithm (e.g., going from 3 to 4) represents a doubling of  $x$  (e.g., going from 8 to 16).

## Logarithmic Transformations—Base 10 Logarithm (common log)

Another frequently used base for a logarithm is 10 — a base 10 logarithm is also called the common log. Each 1 unit increase in a base 10 logarithm equals a 10-fold increase in  $x$ .

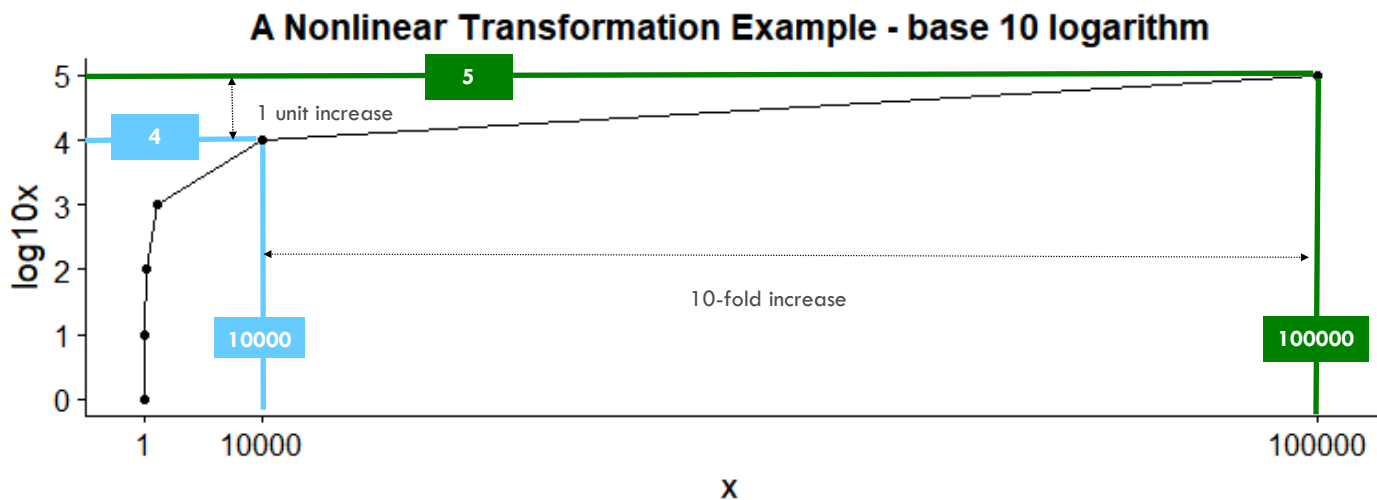
Nonlinear transformation—base 10 logarithm

```
options(scipen = 999) # disable scientific notation

trans3 <- tibble(x = c(1,10,100,1000,10000,100000), log10x = log10(x))

ggplot(trans3, aes(x = x, y = log10x)) +
  geom_line() +
  geom_point() +
  scale_x_continuous(breaks = c(1,10000,100000)) +
  labs(title = "A Nonlinear Transformation Example - base 10 logarithm")
```

| x      | log10x |
|--------|--------|
| 1      | 0      |
| 10     | 1      |
| 100    | 2      |
| 1000   | 3      |
| 10000  | 4      |
| 100000 | 5      |



This chart depicts a base 10 logarithm. The logarithm is the power (i.e., exponent) to which a base must be raised in order to produce a number. Therefore,  $\log_{10}(10,000) = 4$ , because 10 must be raised to a power of 4 to get 10,000.

Notice that each 1 unit increase in a base 10 logarithm (e.g., going from 4 to 5) represents a 10-fold increase of  $x$  (e.g., going from 10,000 to 100,000).

## Logarithmic Transformations—Natural Logarithm

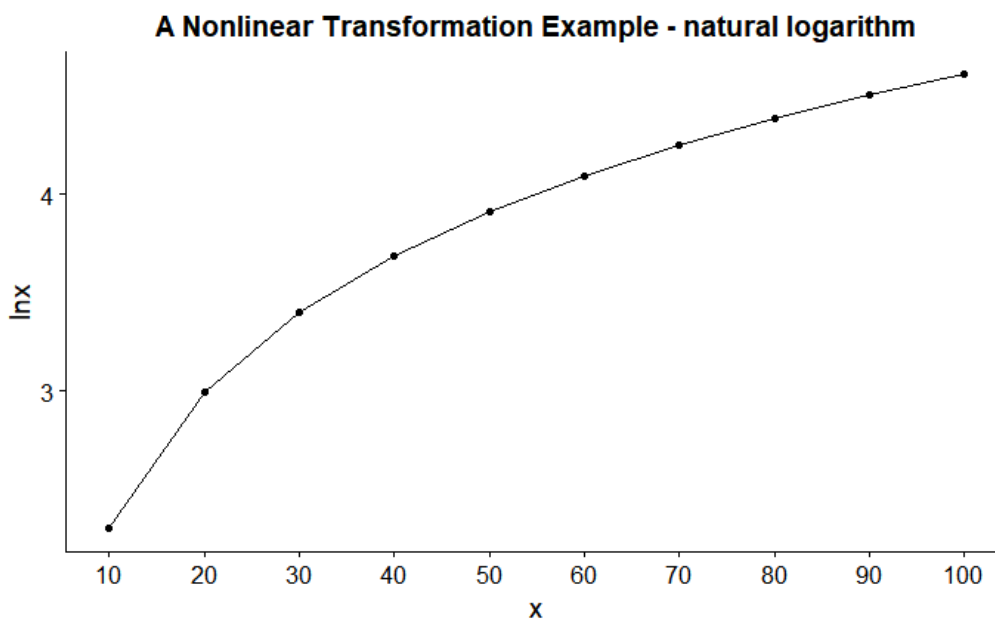
The most often used logarithm in statistics is the natural logarithm. The base of the natural logarithm is the mathematical constant  $e$ , which is called Euler's number. It is an irrational number approximately equal to 2.718281828459. The natural logarithm is often abbreviated  $\ln$ .

Nonlinear transformation—natural logarithm

```
trans4 <- tibble(x = seq(10,100, by = 10), lnx = log(x))
```

```
ggplot(trans4, aes(x = x, y = lnx)) +  
  geom_line() +  
  geom_point() +  
  scale_x_continuous(breaks = seq(10,100, by = 10)) +  
  labs(title = "A Nonlinear Transformation Example - natural logarithm")
```

| x   | lnx      |
|-----|----------|
| 10  | 2.302585 |
| 20  | 2.995732 |
| 30  | 3.401197 |
| 40  | 3.688879 |
| 50  | 3.912023 |
| 60  | 4.094345 |
| 70  | 4.248495 |
| 80  | 4.382027 |
| 90  | 4.499810 |
| 100 | 4.605170 |



This chart depicts a natural logarithm. The logarithm is the power (i.e., exponent) to which a base must be raised in order to produce a number. Therefore,  $\ln(10) \approx 2.3026$ , because Euler's number must be raised to a power of approximately 2.3026 to get 10. All logarithms are linear transformations of one another — that is, log base 2 of a set of numbers will be perfectly correlated with the natural log of the same set of numbers. The key difference is how you desire to explain the results — for example, if a predictor variable in a regression model is log transformed, you should decide if you want to interpret the results in terms of a 10-fold increase in  $x$  (base 10 logarithm) or a 2-fold in  $x$  (base 2 logarithm). As you will see, the natural logarithm has some nice properties that make it particularly useful for interpretation, so we will stick with it in our examples.

## Antilog (Inverse log) of the Natural Log Transformation


Once a set of numbers is log transformed (i.e.,  $\ln(10) = 2.3026$ ), we can back transform it (called taking the antilog or inverse log), by exponentiating the log transformed value to the base that was used. For the natural log transformation, that means raising  $e$  to the power of the transformed value, so  $e^{2.3026} = 10$ .

Demonstrate the back transformation of values

```
trans4 <- mutate(trans4, invx = exp(lnx))
```

In R, type  $\ln x = \log(x)$

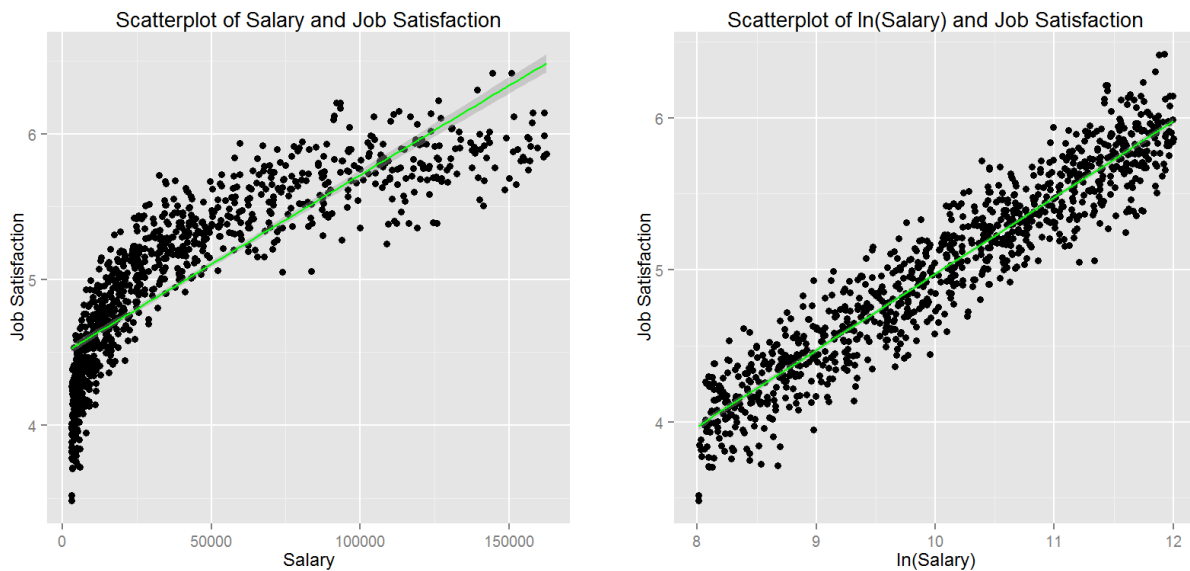
In R, type  $\exp(\ln x)$



| x   | lnx      | invx |
|-----|----------|------|
| 10  | 2.302585 | 10   |
| 20  | 2.995732 | 20   |
| 30  | 3.401197 | 30   |
| 40  | 3.688879 | 40   |
| 50  | 3.912023 | 50   |
| 60  | 4.094345 | 60   |
| 70  | 4.248495 | 70   |
| 80  | 4.382027 | 80   |
| 90  | 4.499810 | 90   |
| 100 | 4.605170 | 100  |

## Non-Linear Transformation to Achieve Linearity between Two Variables

So, what is the relevance of logarithmic transformations to statistical modeling? By transforming one or more variables in which the relationship is nonlinear, we have the possibility of turning a nonlinear relationship at the level of the raw variables into a linear relationship at the level of the transformed variable(s). If we're successful, we can then use OLS regression to model the transformed variables(s).



Imagine that a researcher was interested in the relationship between salary and job satisfaction. The chart on the left plots her data. Here, we see a general positive trend (i.e., as salary increases, job satisfaction increases), but the relationship between salary and job satisfaction doesn't seem to be constant across the whole range of salary. That is, at low salaries each additional dollar seems to be related to a larger increase in job satisfaction as compared to high salaries.

The chart on the right plots job satisfaction against the natural log of salary. Now the relationship looks linear. Each one unit increase in the natural log of salary is associated with a similar increase in job satisfaction across the entire range of salaries. You will see in a bit that we can interpret the slope as the expected change in job satisfaction for a certain percent increase in salary (rather than a unit change). Since we discuss the effect in terms of a percentage change in salary, rather than a unit change, then the non-linear nature of the raw relationship is captured because, for example, a 5% increase in salary is a non-constant increase in actual dollars (a 5% increase for a person earning 10,000 equates to a \$500 raise; a 5% increase for a person earning 100,000 equates to a \$5000 raise).



## Data Example



### HAPPY PLANET

#### Are the Residents of Rich Countries Happier & Healthier?

In this example, we will use data compiled by The New Economics Foundation to compute the Happy Planet Index. The dataset represents 151 countries. We will explore a subset of data, all of the available data are here:

<http://www.happyplanetindex.org/>

We will model 3 variables:

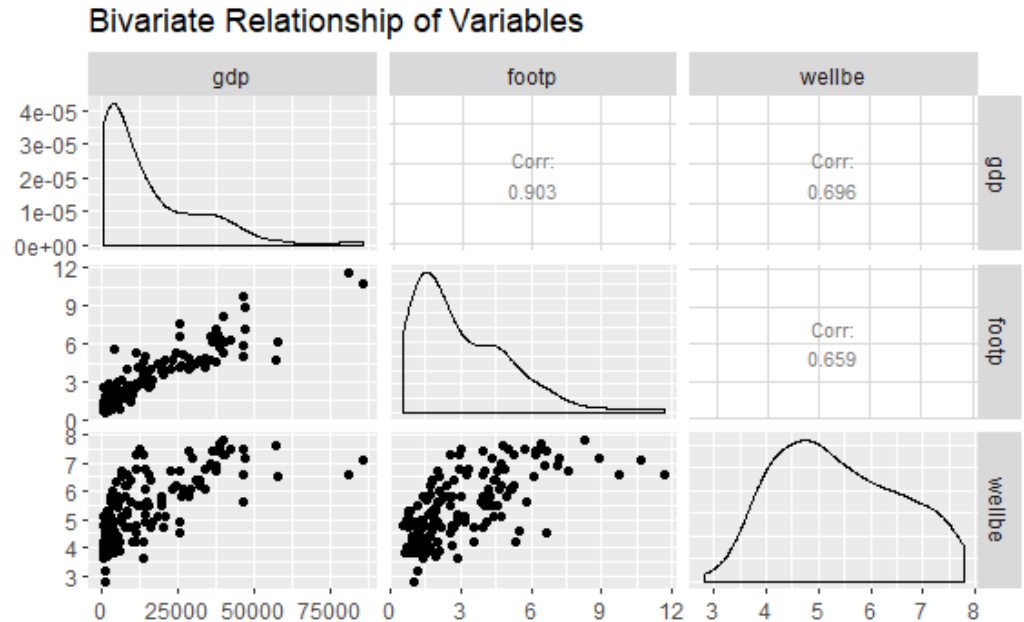
1. **gdp**—gross domestic product per capita—this is a measure of a country's wealth per person.
2. **wellbe**—using the 'Ladder of Life' from the Gallup World Poll, a random sample of residents in each country was asked to imagine a ladder, where 0 represents the worst possible life and 10 the best possible life, and report the step of the ladder they are currently standing on. The wellbe score for each country is the average of the responses for surveyed residents.
3. **footp**—the ecological footprint of the country, expressed as global hectares (gha) per capita.

**DATASET:** happyplanet.csv

Prepare and describe the data

```
# read in data
hp <- read_csv("happyplanet.csv")

# create a scatterplot matrix
scatterplot <- ggpairs(hp, columns = c("wellbe", "gdp", "footp"),
  upper = list(continuous = wrap("cor", size=3)),
  title = "Bivariate Relationship of Variables",
  print(scatterplot, progress=FALSE )
```

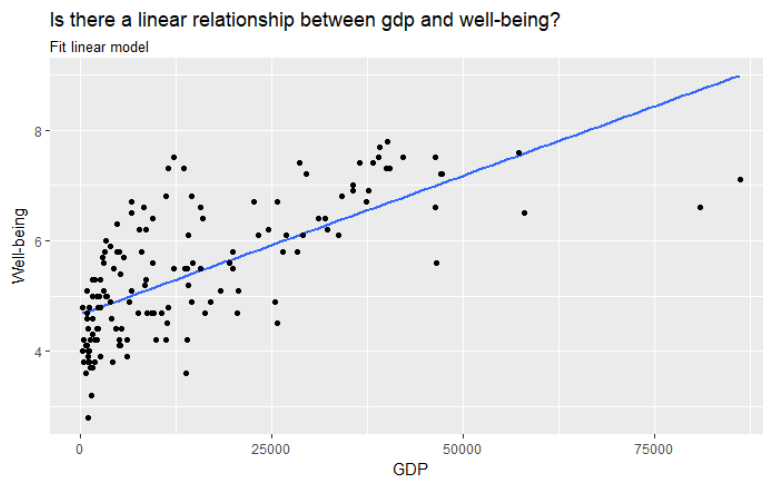


The histograms for gdp and footp are not at all normal, they both have a clear positively (i.e, right), skewed distribution.

## Let's Start by Considering Wellbeing and GDP

Create a scatterplot of GDP and well-being - overlay linear model

```
ggplot(hp, aes(x = gdp, y = wellbe)) +  
  geom_smooth(method = lm, se = FALSE) +  
  geom_point() +  
  labs(title = "Is there a linear relationship between gdp and well-being?",  
        subtitle = "Fit linear model",  
        x = "GDP", y = "Well-being")
```

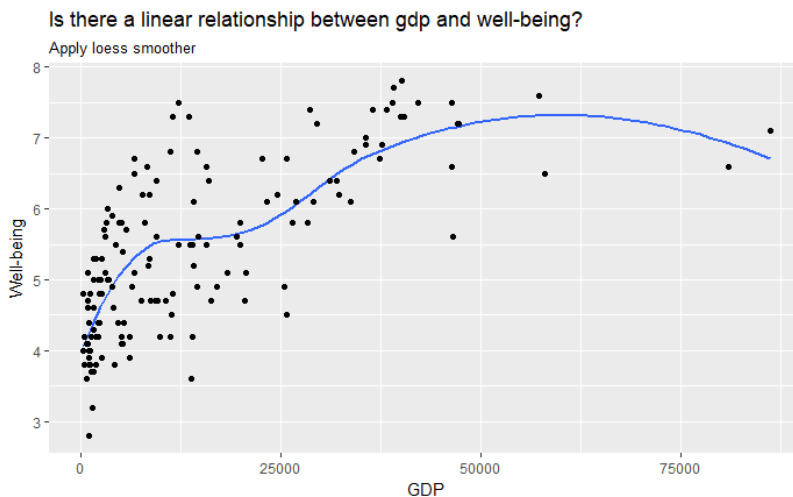


There is a clear increasing trend to these data (i.e., a positive relationship), but does a straight line adequately describe this relationship?

## Wellbeing and GDP—with loess smoother

Create a scatterplot of GDP and well-being - apply loess smoother

```
ggplot(hp, aes(x = gdp, y = wellbe)) +  
  geom_smooth(method = "loess", se = FALSE) +  
  geom_point() +  
  labs(title = "Is there a linear relationship between gdp and well-being?",  
        subtitle = "Apply loess smoother",  
        x = "GDP", y = "Well-being")
```



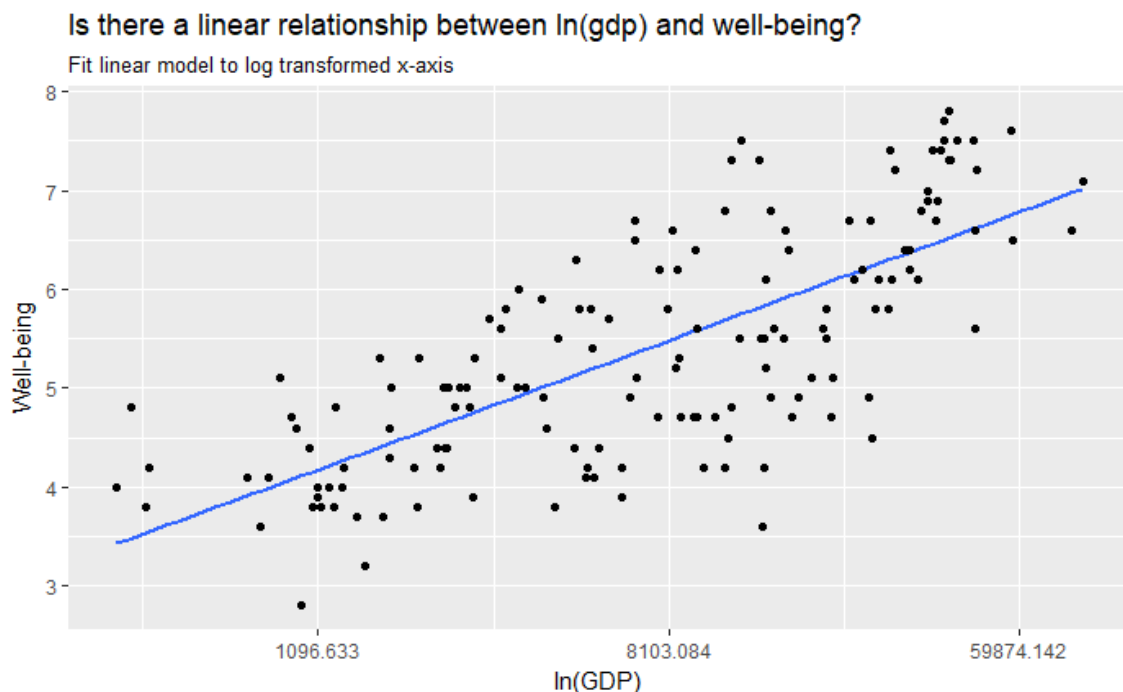
A loess smoother is a weighted localized regression that fits a series of models to clusters of points. Data points in a given cluster are weighted by a smooth decreasing function of their distance from the center of the cluster. The loess smoother helps us to see the pattern of observed data as we move from low to high values on the x-axis, rather than imposing a certain functional form (e.g., a straight line). When we apply a loess smoother to these data, it becomes obvious that the relationship between GDP and well-being is not linear.

## Apply a Natural Log Transformation to GDP

Our initial histogram depicted a positively skewed distribution for GDP. Often when we see this type of distribution, a non-linear transformation that pulls in the tail of the variable can improve the distribution of the positively skewed variable, but it can also linearize the relationship that it has with other variables. Let's use the `scale_x_continuous` function in `ggplot` to transform the x-axis to the natural log of GDP.

Create a scatterplot of  $\ln(\text{GDP})$  and wellbeing

```
ggplot(hp, aes(x = gdp, y = wellbe)) +  
  geom_smooth(method = lm, se = FALSE) +  
  geom_point() +  
  scale_x_continuous(trans = "log") +  
  labs(title = "Is there a linear relationship between  $\ln(\text{gdp})$  and well-being?",  
       subtitle = "Fit linear model to log transformed x-axis",  
       x = " $\ln(\text{GDP})$ ", y = "Well-being")
```



By taking the natural log of GDP, we see a spreading out of the data points (notice there is no more clumping at the low end). In addition, the relationship between  $\ln(\text{GDP})$  and well-being seems to be well-captured by a straight line.

## Fit a Linear Regression Model to the Data

Log transform gdp and fit linear regression model

```
# take the natural log transformation of gdp
hp <- mutate(hp, lngdp = log(gdp))

# fit the simple linear regression
logx <- lm(data = hp, wellbe ~ lngdp)
ols_regress(logx)
```

Model Summary

|                |       |           |        |
|----------------|-------|-----------|--------|
| R              | 0.737 | RMSE      | 0.794  |
| R-Squared      | 0.542 | Coef. Var | 14.712 |
| Adj. R-Squared | 0.539 | MSE       | 0.631  |
| Pred R-Squared | 0.532 | MAE       | 0.655  |

RMSE: Root Mean Square Error

MSE: Mean Square Error

MAE: Mean Absolute Error

ANOVA

|            | Sum of Squares | DF  | Mean Square | F       | Sig.   |
|------------|----------------|-----|-------------|---------|--------|
| Regression | 111.415        | 1   | 111.415     | 176.654 | 0.0000 |
| Residual   | 93.974         | 149 | 0.631       |         |        |
| Total      | 205.389        | 150 |             |         |        |

Parameter Estimates

| model       | Beta   | Std. Error | Std. Beta | t      | sig   | lower  | upper |
|-------------|--------|------------|-----------|--------|-------|--------|-------|
| (Intercept) | -0.385 | 0.440      |           | -0.874 | 0.383 | -1.254 | 0.485 |
| lngdp       | 0.651  | 0.049      | 0.737     | 13.291 | 0.000 | 0.554  | 0.748 |

The intercept represents the predicted wellbe score when  $\ln \text{gdp} = 0$ . By back-transforming a  $\ln(\text{GDP})$  score of 0,  $\exp(0)$ , we obtain 1. So, if a country has a GDP of 1, we predict it will have a wellbe score of  $-0.39$ . This is not particularly useful since the lowest GDP in the dataset is 347 (for Congo). We could center  $\ln(\text{GDP})$  to make a more useful intercept.

The slope of this model presents the expected change in wellbe for a one unit increase in  $\ln(\text{GDP})$ . The p-value for the slope is small, indicating that there is a significant, positive relationship between  $\ln(\text{GDP})$  and well-being.

## A More Useful Interpretation

Typically, when we discuss the effects of our predictors, we want to discuss them in terms of the original metric of the variables. Telling an audience about a one unit increase in  $\ln gdp$  isn't nearly as intuitive or useful as telling them about a change in GDP. Fortunately, we can, with a little care, discuss the effects in terms of the raw variables. The regression coefficient associated with a  $\ln(x)$  variable and a  $y$  variable in its original metric can be interpreted as the expected change in  $y$  for a 1 percent change in  $x$ . Since we express the effect in terms of a 1 percent increase in GDP, rather than a 1 unit increase, we account for the nonlinear relationship between GDP and well-being. When we are at low values of GDP, a one percent increase is a much smaller unit increase than at high values of GDP. For example, at 5,000 GDP, a one percent increase in GDP equates to a 50 unit increase (i.e.,  $5,000 \cdot .01 = 50$ ). But at 50,000 GDP, a one percent increase in GDP equates to a 500 unit increase (i.e.,  $50,000 \cdot .01 = 500$ ). Therefore, by discussing the effect in terms of percent change, we capture the nonlinear nature of the relationship.

The function `ln.x` helps you to interpret a regression coefficient in which a natural log transformation has been applied to the  $x$  variable, but the  $y$  variable is in its natural metric.

| Parameter Estimates |        |            |           |        |       |        |       |
|---------------------|--------|------------|-----------|--------|-------|--------|-------|
| model               | Beta   | Std. Error | Std. Beta | t      | Sig   | lower  | upper |
| (Intercept)         | -0.385 | 0.440      |           | -0.874 | 0.383 | -1.254 | 0.485 |
| $\ln gdp$           | 0.651  | 0.049      | 0.737     | 13.291 | 0.000 | 0.554  | 0.748 |

This code sets up the function — you don't need to change anything here. Just execute it to create the function.

Function to interpret regression slope for  $\ln$  transformed  $x$ , original  $y$

```
ln.x <- function(slope, x_chg) {
  new_slope <- slope * (log(1 + (x_chg/100)))
  return(new_slope)
}
```

This code executes the function. You add in the slope you want to interpret (.651 in this case), and the percent change in  $x$  that you want to consider.

Use function `ln.x` to interpret parameters

```
ln.x(slope = .651, x_chg = 1)

ln.x(slope = logx$coefficients["lngdp"], x_chg = 1)
```

If you want to be precise, rather than typing in the slope (which has been rounded to 3 decimal places in the output), you can refer to the slope in the model output object.

```
ln.gdp
0.006480771
```

A 1% increase in GDP is associated with a .006 unit increase in the average well-being score of residents in the country.

Choose a different percent change (you can pick whatever makes most sense).

```
ln.x(slope = logx$coefficients["lngdp"], x_chg = 100)
```

```
ln.gdp
0.4514551
```

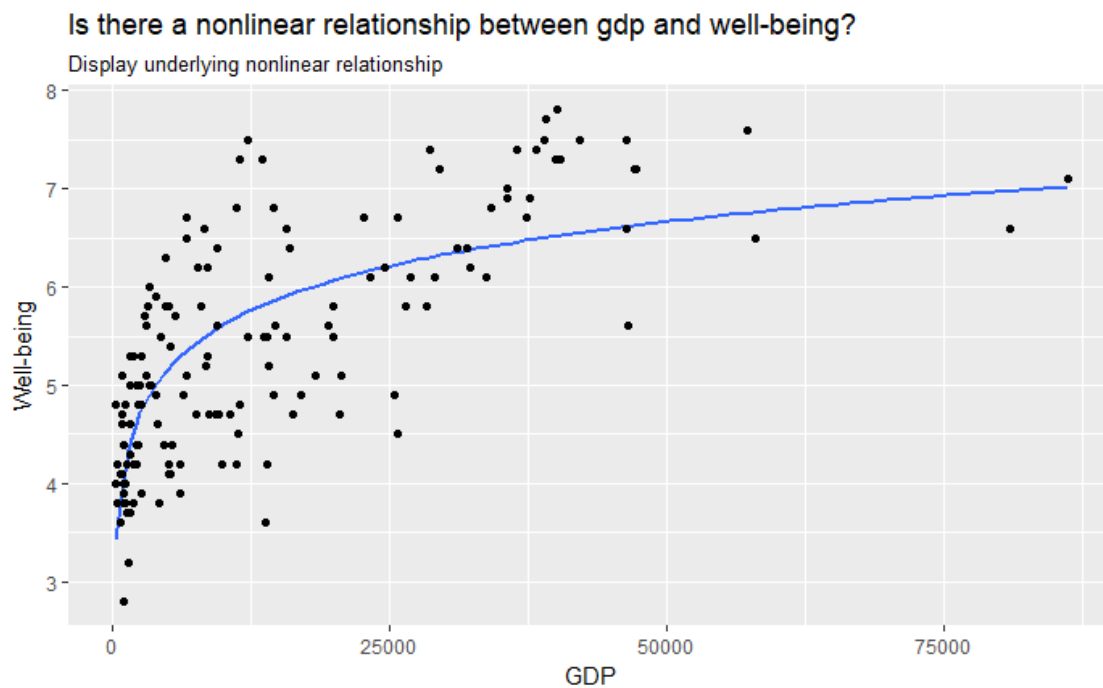
A 100% increase in GDP is associated with a .451 unit increase in wellbe.

## The Underlying Curved Relationship

When we fit a straight line to the relationship between  $\ln \text{gdp}$  and  $\text{wellbe}$ , we are actually fitting a curved line to the relationship between GDP and well-being. Note that a linear model on the logarithmic scale corresponds to a nonlinear model on the original scale.

Visualize the nonlinear relationship

```
# create a scatterplot of GDP and wellbeing - represent the underlying nonlinear curve
ggplot(hp, aes(x = gdp, y = wellbe)) +
  geom_smooth(method = lm, se = FALSE, formula = y ~ 1 + log(x)) +
  geom_point() +
  labs(title = "Is there a nonlinear relationship between gdp and well-being?",
       subtitle = "Display underlying nonlinear relationship",
       x = "GDP", y = "Well-being")
```



## What if the Outcome, Rather than the Predictor, Needs to be Transformed?

In some cases, it will be your y variable that needs to be transformed. This can also be accommodated. For the sake of demonstration, let's simply consider lngdp as the y variable, and wellbe as the x variable.

Fit regression model

```
logy <- lm(data = hp, lngdp ~ wellbe)
ols_regress(logy)
```

Model summary

|                |       |           |        |
|----------------|-------|-----------|--------|
| R              | 0.737 | RMSE      | 0.898  |
| R-Squared      | 0.542 | Coef. Var | 10.115 |
| Adj. R-Squared | 0.539 | MSE       | 0.807  |
| Pred R-Squared | 0.532 | MAE       | 0.745  |

RMSE: Root Mean Square Error  
MSE: Mean Square Error  
MAE: Mean Absolute Error

ANOVA

|            | Sum of Squares | DF  | Mean Square | F       | Sig.   |
|------------|----------------|-----|-------------|---------|--------|
| Regression | 142.473        | 1   | 142.473     | 176.654 | 0.0000 |
| Residual   | 120.170        | 149 | 0.807       |         |        |
| Total      | 262.644        | 150 |             |         |        |

Parameter Estimates

| model       | Beta  | Std. Error | Std. Beta | t      | Sig.  | lower | upper |
|-------------|-------|------------|-----------|--------|-------|-------|-------|
| (Intercept) | 4.383 | 0.346      |           | 12.664 | 0.000 | 3.699 | 5.066 |
| wellbe      | 0.833 | 0.063      | 0.737     | 13.291 | 0.000 | 0.709 | 0.957 |

The intercept (4.383) is the predicted lngdp when wellbe = 0. By taking the antilog of the intercept we obtain 80.04, so this is the predicted GDP when wellbe = 0. Now, there are no countries with a wellbe score of 0, so it would make sense to center wellbe and refit the model for a more meaningful intercept.

```
exp(logy$coefficients["(Intercept)"])
```

The slope (.833) is the expected change in lngdp for a one unit increase in wellbe.

Function to interpret regression slope for ln transformed y, original x

```
ln.y <- function(slope, x_chg) {
  new_slope <- 100 * (exp(slope * x_chg) - 1)
  return(new_slope)
}
```

The function ln.y helps you to interpret a regression coefficient in which a natural log transformation has been applied to the y variable, but the x variable is in its natural metric.

Use function ln.y to interpret parameters

```
ln.y(slope = .833, x_chg = 1)
ln.y(slope = logy$coefficients["wellbe"], x_chg = 1)
```

```
wellbe
129.9913
```

A 1 unit increase in wellbe is associated with a 130% increase in GDP.



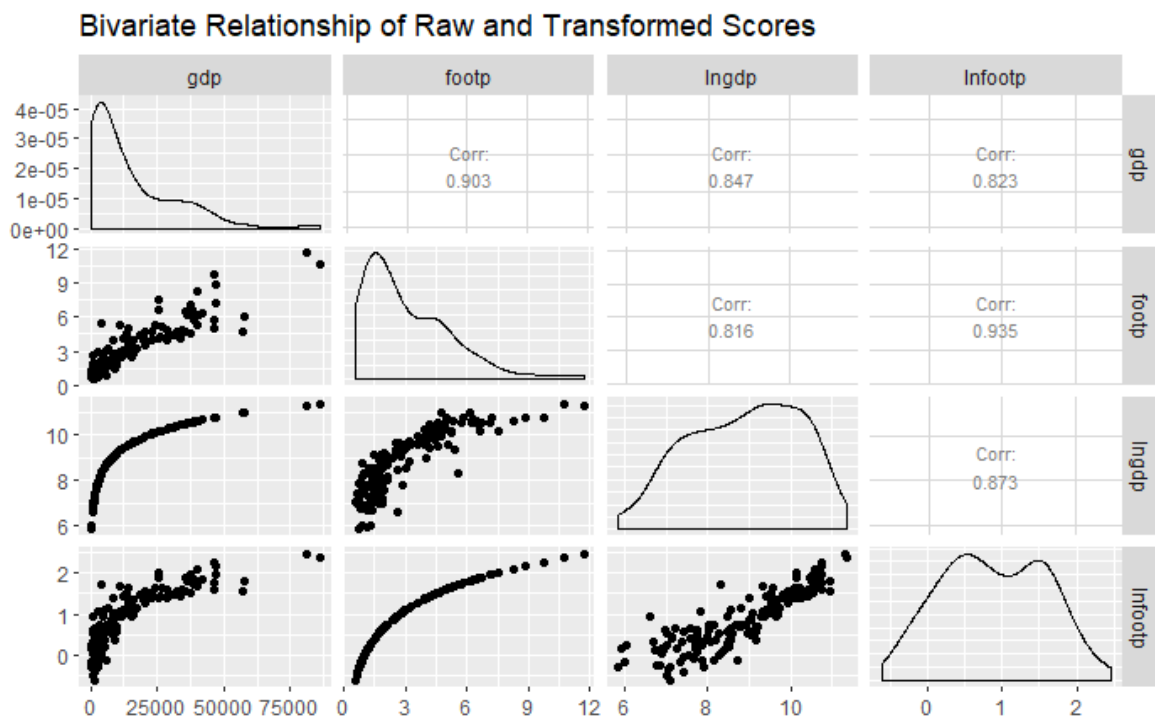
## What if Both the Outcome and Predictor Need to be Transformed?

There are times when both an x variable and the y outcome need to be transformed. In this example we will consider another variable in the dataset—the ecological footprint of each country (footp). We will examine the relationship between GDP and ecological footprint.

Transform footp and create a scatterplot matrix of the raw and transformed variables

```
hp <- mutate(hp, lnfootp = log(footp))

scatterplot <- ggpairs(hp, columns = c("gdp", "footp", "lngdp", "lnfootp"),
  upper = list(continuous = wrap("cor", size=3)),
  title = "Bivariate Relationship of Raw and Transformed Scores")
print(scatterplot, progress=FALSE)
```



The scatterplot of gdp and footp also demonstrates a curvilinear relationship. Transforming just gdp doesn't help. Transforming both makes the scatterplot look more linear.

## Fit Regression Model for ln Transformed Outcome and Predictor

Fit regression model

```
logxy <- lm(data = hp, lnfootp ~ lngdp)
ols_regress(logxy)
```

| Model Summary  |                |            |             |         |        |        |        |
|--|----------------|------------|-------------|---------|--------|--------|--------|
| R  | 0.873          | RMSE       |             | 0.343   |        |        |        |
| R-Squared  | 0.762          | Coef. var  |             | 38.701  |        |        |        |
| Adj. R-Squared   | 0.761          | MSE        |             | 0.117   |        |        |        |
| Pred R-Squared   | 0.755          | MAE        |             | 0.264   |        |        |        |
| RMSE: Root Mean Square Error<br>MSE: Mean Square Error<br>MAE: Mean Absolute Error |                |            |             |         |        |        |        |
| ANOVA  |                |            |             |         |        |        |        |
|  | Sum of Squares | DF         | Mean Square | F       | Sig.   |        |        |
| Regression   | 56.045         | 1          | 56.045      | 477.702 | 0.0000 |        |        |
| Residual   | 17.481         | 149        | 0.117       |         |        |        |        |
| Total  | 73.526         | 150        |             |         |        |        |        |
| Parameter Estimates  |                |            |             |         |        |        |        |
| model  | Beta           | Std. Error | Std. Beta   | t       | Sig.   | lower  | upper  |
| (Intercept)  | -3.216         | 0.190      |             | -16.954 | 0.000  | -3.591 | -2.841 |
| lngdp  | 0.462          | 0.021      | 0.873       | 21.856  | 0.000  | 0.420  | 0.504  |

The intercept (-3.216) is the predicted lnfootp when lngdp = 0. By taking the antilog of this value we obtain the intercept in the raw metric of footp. This value is .04.

The slope (.462) is the expected change in lnfootp for a one unit increase in lngdp. It is statistically significant (p-value < .05), indicating that countries with a higher lngdp tend to have a higher lnfootp.

```
exp(logxy$coefficients["(Intercept)"])
```

Function to interpret regression slope for ln transformed y and x

```
ln.xy <- function(slope, x_chg) {
  new_slope <- 100 * (exp(slope * (log(1 + (x_chg/100))))-1)
  return(new_slope)
}
```

The function ln.xy helps you to interpret a regression coefficient in which a natural log transformation has been applied to both x and y.

Use function ln.xy to interpret parameters

```
ln.xy(slope = .462, x_chg = 100)
```

```
ln.xy(slope = logxy$coefficients["lngdp"], x_chg = 100)
```

1 lngdp  
37.73926

A 100% increase in GDP is associated with a 38% increase in the ecological footprint.

## **Write Up:**

The relationship between GDP and ecological footprint was examined among 151 countries worldwide. We hypothesized that GDP would be a significant predictor of ecological footprint, specifically, that as GDP increased, so too would the ecological footprint of the country. Before fitting a model to the data, histograms of each variable and a scatterplot were examined. Both variables demonstrated strong positive skew and the scatterplot appeared nonlinear. A natural log transformation was applied to both GDP and ecological footprint. The histograms of the transformed variables were much more normal, and the scatterplot of the transformed variables was indeed linear. A linear regression model was fit to the transformed variables. Consistent with our hypothesis, the slope relating  $\ln(\text{GDP})$  to  $\ln(\text{ecological footprint})$  was statistically significant ( $b=.46$ , 95% CI .42, .50). In terms of the original variables, this indicates that a 100% increase in GDP is associated with a 37.7% increase in ecological footprint.

## Back Transforming

When we transform variables, we need to be cautious to use the correct version of the variable when solving for  $\hat{y}$ . For example, let's use the model below to obtain the predicted ecological footprint for a country with a GDP of 50,000. First, we need to determine the natural logarithm of 50,000 (it's 10.81978). Then, we can plug that value into our resulting regression equation—remember that the predict function in R will do this for us. This gives us the predicted value (1.781842). However, this is in the log scale (b/c we log transformed both  $x$  and  $y$ ). The last step is to put this value back onto the original scale. The antilog (i.e., inverse log) for the natural log is the exponential function (called `exp` in R). Taking the exponential function gives us 5.94—this is the predicted ecological footprint for a country with a GDP of 50,000.

Back transform predicted values

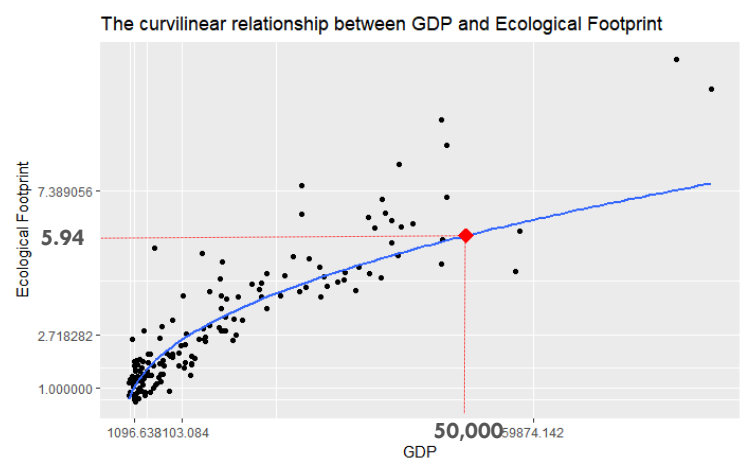
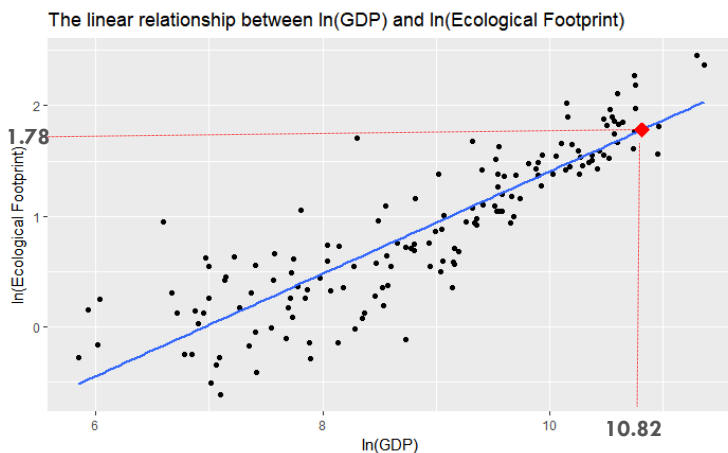
|   |          |
|---|----------|
| <b># get log of 50,000</b><br><b>log(50000)</b>   | 10.81978 |
| <b># get y-hat when gdp = 50,000</b><br><b>predict(logxy, data.frame(lngdp=10.81978))</b> | 1.781842 |
| <b># back transform y-hat to original metric of footp</b><br><b>exp(1.781842)</b>         | 5.940789 |

## Plot the Results

Make plots of transformed and original data

```
# create a scatterplot of ln(GDP) and ln(footp)
ggplot(hp, aes(x = lngdp, y = lnfootp)) +
  geom_point() +
  geom_smooth(method = lm, se = FALSE) +
  annotate("point", x = 10.82, y = 1.78, colour = "red", size = 5, shape = 18) +
  labs(title = "The linear relationship between ln(GDP) and ln(Ecological Footprint)", x = "ln(GDP)", y = "ln(Ecological Footprint)")

# create a scatterplot of GDP and footp
ggplot(hp, aes(x = gdp, y = footp)) +
  geom_point() +
  scale_x_continuous(trans = "log") +
  scale_y_continuous(trans = "log") +
  geom_smooth(method = lm, se = FALSE) +
  coord_trans(x = "exp", y = "exp") +
  annotate("point", x = 50000, y = 5.94, colour = "red", size = 5, shape = 18) +
  labs(title = "The curvilinear relationship between GDP and Ecological Footprint", x = "GDP", y = "Ecological Footprint")
```



## Other Types of Nonlinear Transformations

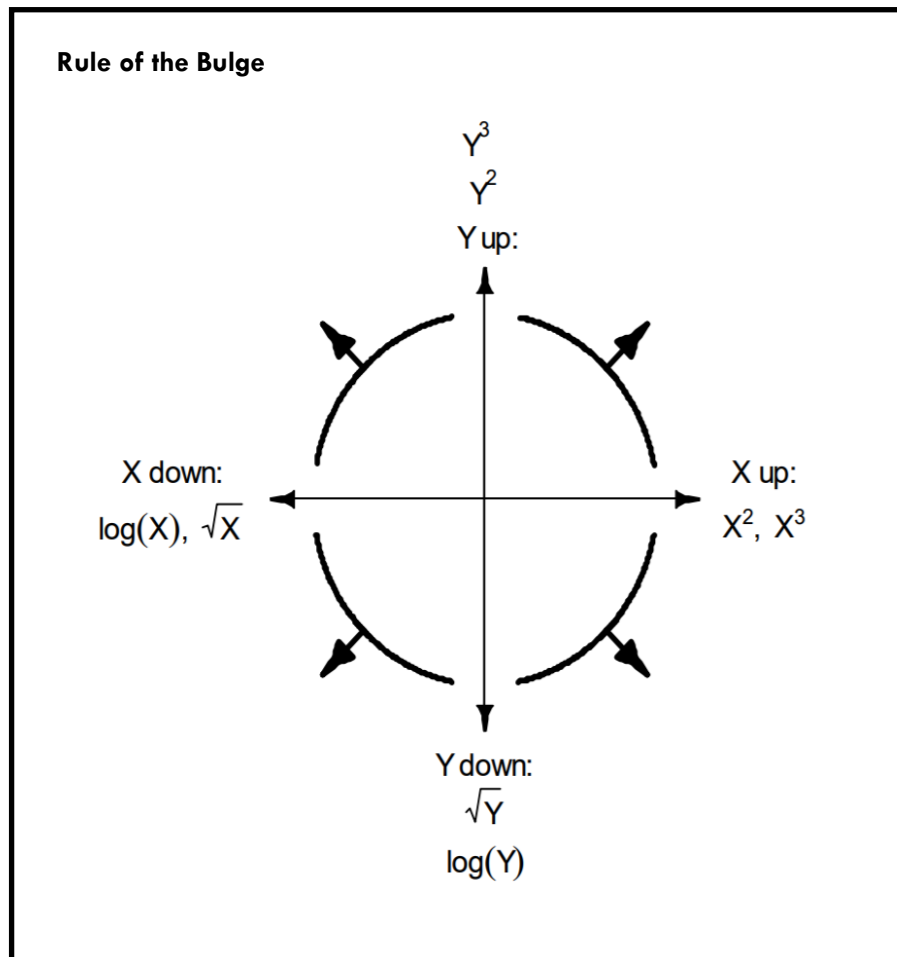
There are other types of nonlinear transformations that you can apply as well. The chart below takes a variable,  $x$ , and applies various transformations. The excel spreadsheet called “transform\_ladder” shows the formulas to transform (top) and back transform (bottom).

| TRANSFORM                |            |          |         |            |                    |                  |           |
|--------------------------|------------|----------|---------|------------|--------------------|------------------|-----------|
| $x$                      | $-(1/x^2)$ | $-(1/x)$ | $\ln x$ | $\sqrt{x}$ | $x\text{-squared}$ | $x\text{-cubed}$ | $\exp(x)$ |
| 5                        | -0.04000   | -0.20000 | 1.60944 | 2.23607    | 25                 | 125              | 148.4132  |
| 10                       | -0.01000   | -0.10000 | 2.30259 | 3.16228    | 100                | 1000             | 22026.47  |
| 15                       | -0.00444   | -0.06667 | 2.70805 | 3.87298    | 225                | 3375             | 3269017   |
| 20                       | -0.00250   | -0.05000 | 2.99573 | 4.47214    | 400                | 8000             | 4.85E+08  |
| 25                       | -0.00160   | -0.04000 | 3.21888 | 5.00000    | 625                | 15625            | 7.2E+10   |
| 30                       | -0.00111   | -0.03333 | 3.40120 | 5.47723    | 900                | 27000            | 1.07E+13  |
| 35                       | -0.00082   | -0.02857 | 3.55535 | 5.91608    | 1225               | 42875            | 1.59E+15  |
| 40                       | -0.00063   | -0.02500 | 3.68888 | 6.32456    | 1600               | 64000            | 2.35E+17  |
| 45                       | -0.00049   | -0.02222 | 3.80666 | 6.70820    | 2025               | 91125            | 3.49E+19  |
| 50                       | -0.00040   | -0.02000 | 3.91202 | 7.07107    | 2500               | 125000           | 5.18E+21  |
| BACK TRANSFORM (INVERSE) |            |          |         |            |                    |                  |           |
| $x$                      | $-(1/x^2)$ | $-(1/x)$ | $\ln x$ | $\sqrt{x}$ | $x\text{-squared}$ | $x\text{-cubed}$ | $\exp(x)$ |
| 5                        | 5          | 5        | 5       | 5          | 5                  | 5                | 5         |
| 10                       | 10         | 10       | 10      | 10         | 10                 | 10               | 10        |
| 15                       | 15         | 15       | 15      | 15         | 15                 | 15               | 15        |
| 20                       | 20         | 20       | 20      | 20         | 20                 | 20               | 20        |
| 25                       | 25         | 25       | 25      | 25         | 25                 | 25               | 25        |
| 30                       | 30         | 30       | 30      | 30         | 30                 | 30               | 30        |
| 35                       | 35         | 35       | 35      | 35         | 35                 | 35               | 35        |
| 40                       | 40         | 40       | 40      | 40         | 40                 | 40               | 40        |
| 45                       | 45         | 45       | 45      | 45         | 45                 | 45               | 45        |
| 50                       | 50         | 50       | 50      | 50         | 50                 | 50               | 50        |

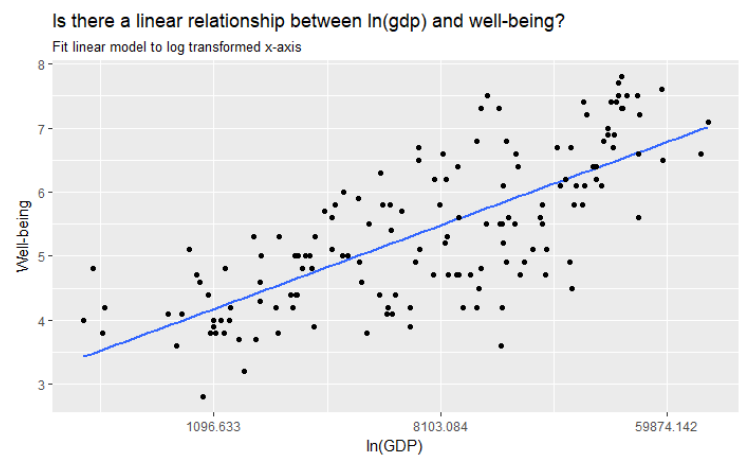
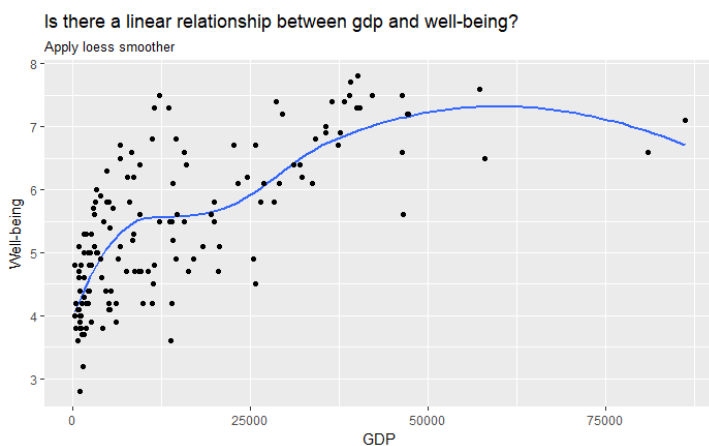
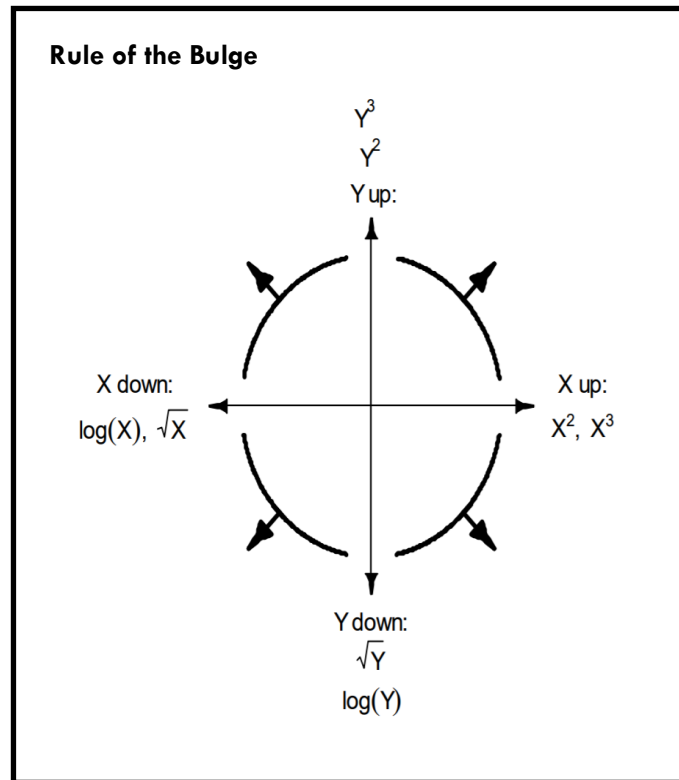
Transformations in the gray columns are “down” transformations, they make  $x$  smaller, and the remainder are “up” transformations, they make  $x$  bigger. In choosing a transformation, consider interpretability and favor the least extreme transformation that will linearize the relationship.

### Rule of the Bulge for Identifying Transformations

How do you know which type of transformation to apply? Start by plotting the data, then use the Rule of the Bulge to suggest an initial transformation to try. Identify your shape and then choose the transformation located on either side of the bulge. If possible, transform  $x$  rather than  $y$  when you have multiple  $x$  variables because a transformation of  $y$  will have to be the right transformation for all other  $x$  variables.



## Rule of the Bulge Example

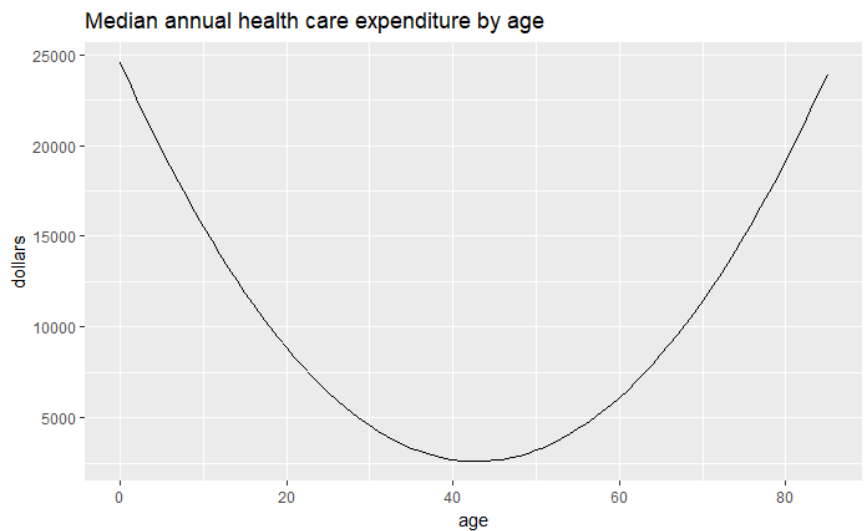


Recall our first example, the relationship between GDP and well-being. The chart on the left (of the raw variables) looks like the upper left quadrant of the Rule of the Bulge. We could then try going up in Y or down in X. We chose to go down in X. The chart on the right plots  $\ln(\text{GDP})$  against well-being and we see that the transformation worked, we now have a linear relationship.



## Introduction to Polynomial Regression

The basic regression model assumes that the relationship between  $x$  and  $y$  is linear. However, in some cases the effect of a given predictor may differ by levels of that very predictor—the “effect of  $x$ ” differs as  $x$  increases. Consider the relationship between age and annual health care expenditures. As humans, we tend to cost a lot when we’re first born, and in old age. The relationship between age and median medical expenditures can be represented by a quadratic function that takes on the shape of a parabola. As demonstrated by the graph, there is a relationship between  $x$  (age) and  $y$  (medical costs in dollars), but it is not linear. A close examination of the graph indicates that the effect of each additional year changes depending on where on the  $x$  axis we focus. During childhood, the effect of age on medical expenses is negative. During old age, the effect of age on medical expenses is positive.



$$\hat{y}_i = 24602 + (-1028x_i) + (12x_i^2)$$

We can accommodate this type of relationship between  $x$  and  $y$  with polynomial regression. In polynomial regression we include squared ( $x^2$ ), cubed ( $x^3$ ) or even higher order terms in order to account for the changing effect of  $x$  on  $y$  at different levels of  $x$ . This addition of a squared term (i.e.,  $y$  regressed on  $x$  and  $x^2$ , referred to as a quadratic function) allows for one bend of the curve. The addition of each additional higher order polynomial term allows for an additional bend. For example,  $y$  regressed on  $x$ ,  $x^2$ , and  $x^3$  would allow for 2 bends (cubic model). However, it is rare to see polynomials beyond the second order ( $x^2$ ) in Psychology.

## Find the Best Polynomial Function for a Curvilinear Relationship



### AN EXAMPLE

#### Practice and Performance

Suppose that we are interested in the effect of time spent in practice on the performance of a visual discrimination task. Subjects are randomly assigned to different levels of practice, following which a test of visual discrimination is administered, and the number of correct responses is recorded for each subject. 40 subjects were randomly assigned to practice 0 minutes, 2 minutes, 4 minutes, 6 minutes, 8 minutes, 10 minutes, 12 minutes, or 14 minutes.

There are two variables:

1. practice — minutes spent practicing, this was assigned by the experimenter
2. score — the number of correct answers on the test

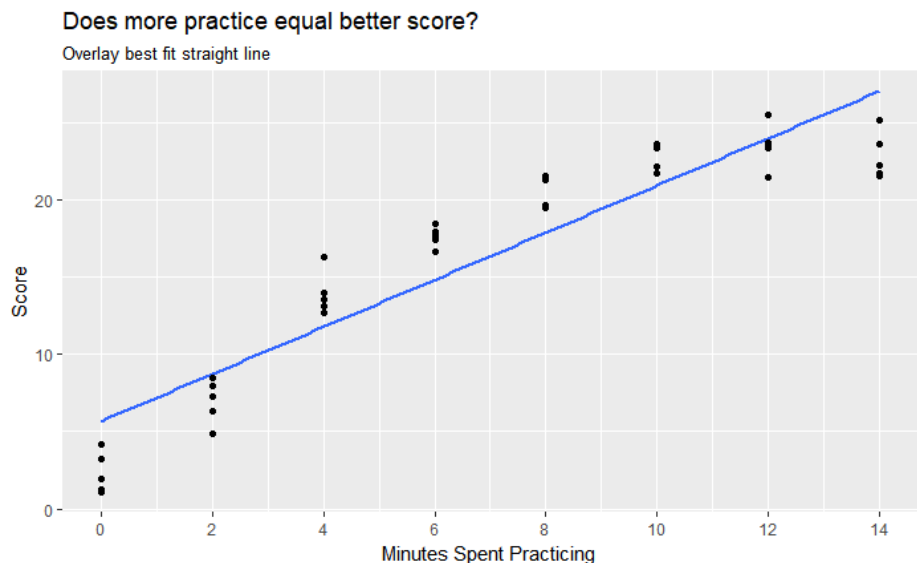
**DATASET:** cogtest.csv

Import data

```
cogtest <- read_csv("cogtest.csv")
```

Overlay a linear function

```
# plot data with best fit straight line
ggplot(cogtest, aes(x = practice, y = score)) +
  geom_smooth(method = "lm", se = FALSE) +
  geom_point() +
  scale_x_continuous(limits=c(0,14), breaks = seq(0, 14, by = 2)) +
  labs(title = "Does more practice equal better score?",
       subtitle = "Overlay best fit straight line",
       x = "Minutes Spent Practicing", y = "Score")
```

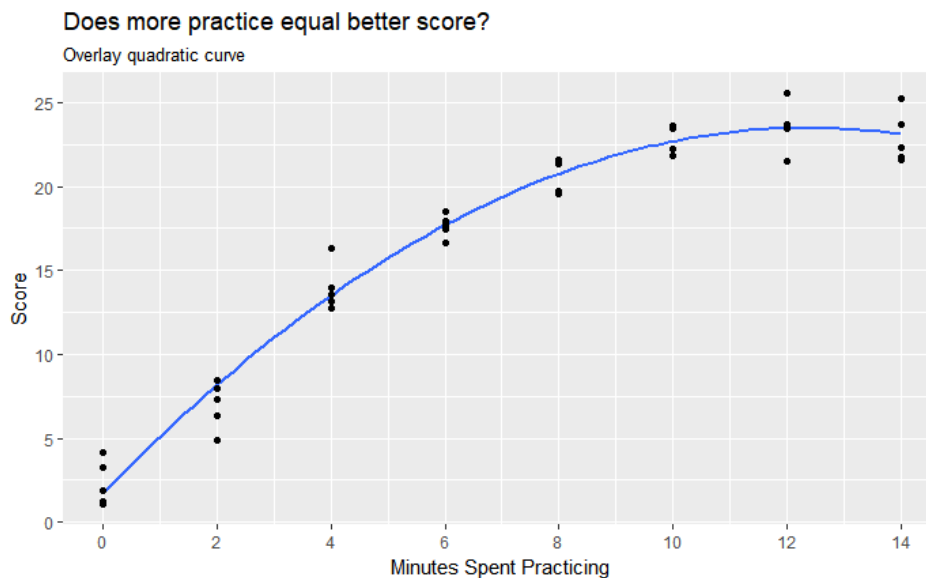


We see a clear increasing trend. However, the increasing trend appears to slow down at higher levels of x.

## Does a Quadratic Function Fit the Points Better?

Overlay a quadratic function

```
# plot data with quadratic function
ggplot(cogtest, aes(x = practice, y = score)) +
  geom_smooth(method = "lm", formula = y ~ poly(x, 2), se = FALSE) +
  geom_point() +
  scale_x_continuous(limits=c(0,14), breaks = seq(0, 14, by = 2)) +
  labs(title = "Does more practice equal better score?",
       subtitle = "Overlay quadratic curve",
       x = "Minutes Spent Practicing", y = "Score")
```



The quadratic function seems to provide a much better fit to the data. The curved line goes through the data points at each level of x.

## Let's Fit a Polynomial Regression Model to these Data

When you assess a polynomial regression model, the procedure is to test a series of models, including one additional polynomial term in each successive model (e.g.,  $x$ ,  $x^2$ ,  $x^3$ , and so forth). We want to adopt the simplest model, so we will assess each model starting with the simplest, and stop when the highest order term is no longer significant, then we'll adopt the previous model (i.e., where the highest order term was significantly different than 0). Note that we will keep all lower order terms if the highest order term is significant, regardless of whether they are significant or not.

Prepare polynomial terms

```
cogtest <- mutate(cogtest,  
  practice2 = practice^2,  
  practice3 = practice^3)
```

## Explore the Fit of a Straight Line

Fit a linear model

```
lm_lin <- lm(data = cogtest, score ~ practice)
ols_regress(lm_lin)
```

Model Summary

|                |       |           |        |
|----------------|-------|-----------|--------|
| R              | 0.925 | RMSE      | 2.952  |
| R-Squared      | 0.856 | Coef. Var | 18.017 |
| Adj. R-Squared | 0.852 | MSE       | 8.713  |
| Pred R-Squared | 0.837 | MAE       | 2.512  |

RMSE: Root Mean Square Error

MSE: Mean Square Error

MAE: Mean Absolute Error

ANOVA

|            | Sum of Squares | DF | Mean Square | F       | Sig.   |
|------------|----------------|----|-------------|---------|--------|
| Regression | 1964.353       | 1  | 1964.353    | 225.462 | 0.0000 |
| Residual   | 331.078        | 38 | 8.713       |         |        |
| Total      | 2295.431       | 39 |             |         |        |

Parameter Estimates

| model       | Beta  | Std. Error | Std. Beta | t      | Sig.  | lower | upper |
|-------------|-------|------------|-----------|--------|-------|-------|-------|
| (Intercept) | 5.678 | 0.852      |           | 6.664  | 0.000 | 3.953 | 7.403 |
| practice    | 1.529 | 0.102      | 0.925     | 15.015 | 0.000 | 1.323 | 1.735 |

In this simple linear regression, the intercept is the predicted score for someone who receives no practice. The slope is the predicted change in the score for a one unit increase in minutes spent practicing. It is statistically significant, indicating there is a significant positive linear trend — more practice equals a better score in general. However, we observed that there is not a constant rate of increase. At lower levels of practice, each additional minute matters quite a lot, but at high levels of practice, the increment to score for each additional minute is smaller.

## Explore the Need for a Curvilinear Relationship

Fit a quadratic model

```
lm_quad <- lm(data = cogtest, score ~ practice + practice2)
ols_regress(lm_quad)
```

Model Summary

|                |       |           |       |
|----------------|-------|-----------|-------|
| R              | 0.987 | RMSE      | 1.276 |
| R-Squared      | 0.974 | Coef. Var | 7.787 |
| Adj. R-Squared | 0.972 | MSE       | 1.627 |
| Pred R-Squared | 0.969 | MAE       | 0.945 |

RMSE: Root Mean Square Error  
MSE: Mean Square Error  
MAE: Mean Absolute Error

ANOVA

|            | Sum of Squares | DF | Mean Square | F       | Sig.   |
|------------|----------------|----|-------------|---------|--------|
| Regression | 2235.214       | 2  | 1117.607    | 686.706 | 0.0000 |
| Residual   | 60.217         | 37 | 1.627       |         |        |
| Total      | 2295.431       | 39 |             |         |        |

Parameter Estimates

| model       | Beta   | Std. Error | Std. Beta | t       | Sig.  | lower  | upper  |
|-------------|--------|------------|-----------|---------|-------|--------|--------|
| (Intercept) | 1.703  | 0.480      |           | 3.547   | 0.001 | 0.730  | 2.676  |
| practice    | 3.517  | 0.160      | 2.127     | 21.949  | 0.000 | 3.192  | 3.841  |
| practice2   | -0.142 | 0.011      | -1.250    | -12.901 | 0.000 | -0.164 | -0.120 |

These are the results of the quadratic model. The quadratic term (practice2) is statistically significant, indicating that there is a substantial curve to the relationship. We need to maintain this term in the model.

Fit a cubic model

```
lm_cubic <- lm(data = cogtest, score ~ practice + practice2 + practice3)
ols_regress(lm_cubic)
```

Model Summary

|                |       |           |       |
|----------------|-------|-----------|-------|
| R              | 0.987 | RMSE      | 1.270 |
| R-Squared      | 0.975 | Coef. Var | 7.750 |
| Adj. R-Squared | 0.973 | MSE       | 1.612 |
| Pred R-Squared | 0.968 | MAE       | 0.922 |

RMSE: Root Mean Square Error  
MSE: Mean Square Error  
MAE: Mean Absolute Error

ANOVA

|            | Sum of Squares | DF | Mean Square | F       | Sig.   |
|------------|----------------|----|-------------|---------|--------|
| Regression | 2237.395       | 3  | 745.798     | 462.622 | 0.0000 |
| Residual   | 58.036         | 36 | 1.612       |         |        |
| Total      | 2295.431       | 39 |             |         |        |

Parameter Estimates

| model       | Beta   | Std. Error | Std. Beta | t      | Sig.  | lower  | upper |
|-------------|--------|------------|-----------|--------|-------|--------|-------|
| (Intercept) | 1.988  | 0.537      |           | 3.703  | 0.001 | 0.899  | 3.077 |
| practice    | 3.144  | 0.358      | 1.902     | 8.786  | 0.000 | 2.418  | 3.870 |
| practice2   | -0.071 | 0.062      | -0.624    | -1.140 | 0.262 | -0.197 | 0.055 |
| practice3   | -0.003 | 0.003      | -0.416    | -1.163 | 0.252 | -0.009 | 0.003 |

These are the results of the cubic model. The cubic term (practice3) is not significant—there is not a second bend to the relationship. Therefore, the quadratic model is the best one for these data. Therefore, we will move forward with the quadratic model.

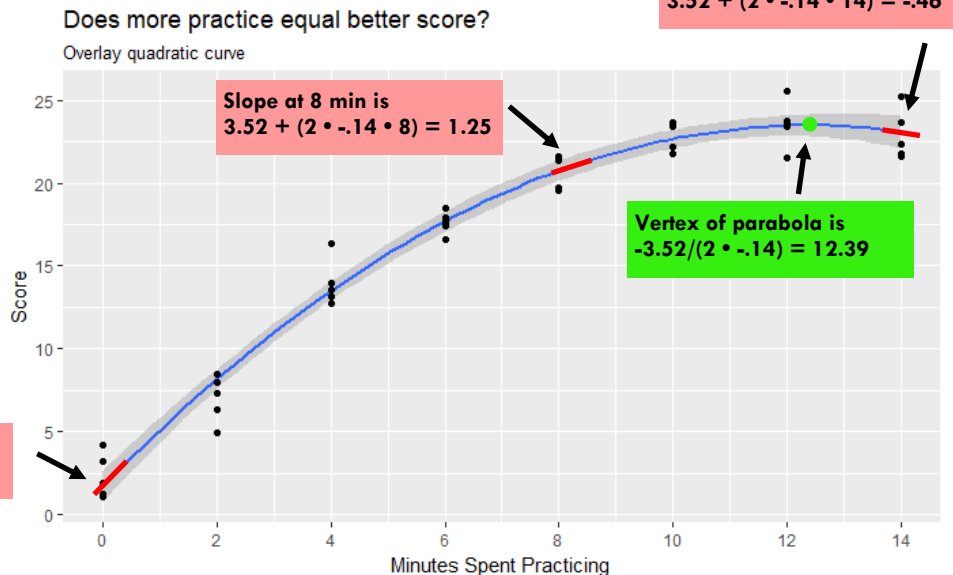
## Model Interpretation for Quadratic Function

| model       | Beta                    |
|-------------|-------------------------|
| (Intercept) | 1.703                   |
| practice    | 3.517 $\leftarrow b_1$  |
| practice2   | -0.142 $\leftarrow b_2$ |

$$\hat{y}_i = b_0 + b_1x_i + b_2x_i^2$$

$$\hat{y}_i = 1.70 + 3.52x_i - .14x_i^2$$

Slope at 0 min  
= 3.52



- The intercept is the predicted test score for people who practice 0 minutes (i.e., practice = 0).
- The graph of a quadratic regression model is the shape of a parabola. This shape can be mound shaped (i.e., an inverted U) or bowl shaped (i.e., a U). Our example is mound shaped. The sign of  $b_2$  indicates whether the shape is a mound or bowl. If  $b_2$  is positive then the parabola is bowl shaped and if  $b_2$  is negative then the parabola is mound shaped.
- The slope of a line drawn tangent to the parabola at a certain  $x$  is estimated by:  $b_1 + 2b_2x_i$ . So, when practice = 0 the slope is:  $3.52 + (2 \cdot -.14 \cdot 0) = 3.52$ . When practice = 8, the slope is  $3.52 + (2 \cdot -.14 \cdot 8) = 1.25$ . In other words, at 0 minutes of practice time, one additional minute of practice is predicted to increase the test score by about 3.52 points (i.e., we predict they'll get more than 3 additional problems correct). But at 8 minutes, one additional minute of practice is predicted to increase the test score by about 1.25 points. Notice that by 12 minutes of practice time, the beneficial effect of additional time spent practicing is nearly gone, and by 14 minutes, each additional minute appears to be detrimental (see the negative slope).
- There is a value of  $x$  along the curve when the slope drawn tangent to the line is 0. In other words, this is the point at which  $\hat{y}$  takes a maximum value if the parabola is a mound or a minimum value if the parabola is a bowl. This point can be estimated with the following formula:  $-b_1/2b_2$ . For our example, this is:  $-3.52/(2 \cdot -.14) = 12.39$ . This is the point where the effect of practicing goes from positive to negative, and this would indicate that there is no need to study for this test for about 12.4 minutes or longer. At this point, there is no additional benefit (i.e., no increment to the test score), and it would seem that any longer will begin to deteriorate one's score. In any regression model, we should not extrapolate beyond the range of our observed data, so we wouldn't want to speculate what will happen beyond 14 minutes. If we want to know, we'd need to conduct a new study with a wider range of practice time.

## A Johnson Neyman Graph to Display the Effect of Practice

```

tcrit = qt(c(.025, .975), df = 38)

varmat <- as.matrix(vcov(lm_quad))
vbx <- varmat["practice", "practice"]
vbx2 <- varmat["practice2", "practice2"]
vbxx2 <- varmat["practice", "practice2"]

slopes <- cogtest %>%
  data_grid(practice)

slopes <- mutate(slopes, tangent_slope = lm_quad$coefficients["practice"] + 2*lm_quad$coefficients["practice2"]*practice,
  se = sqrt(vbx+(4*practice*vbxx2)+((4*practice^2)*vbx2)),
  lwr = tangent_slope - tcrit*se,
  upr = tangent_slope + tcrit*se,
  tstar = tangent_slope/se)

slopes_plot <- cogtest %>%
  data_grid(practice = seq_range(practice, 1000))

slopes_plot <- mutate(slopes_plot, tangent_slope = lm_quad$coefficients["practice"] + 2*lm_quad$coefficients["practice2"]*practice,
  se = sqrt(vbx+(4*practice*vbxx2)+((4*practice^2)*vbx2)),
  lwr = tangent_slope - tcrit*se,
  upr = tangent_slope + tcrit*se,
  tstar = tangent_slope/se)

ggplot(slopes_plot, aes(x = practice, y = tangent_slope)) +
  geom_line() +
  geom_ribbon(aes(ymin = lwr, ymax = upr, color = NULL), alpha = .8, fill = "grey60") +
  geom_line(size = 1) +
  geom_hline(yintercept=0, linetype=2) +
  scale_x_continuous(limits=c(0,14), breaks = seq(0, 14, by = 2)) +
  labs(title = "The changing effect of minutes spent practicing on score",
  x = "Minutes Spent Practicing", y = "Score")

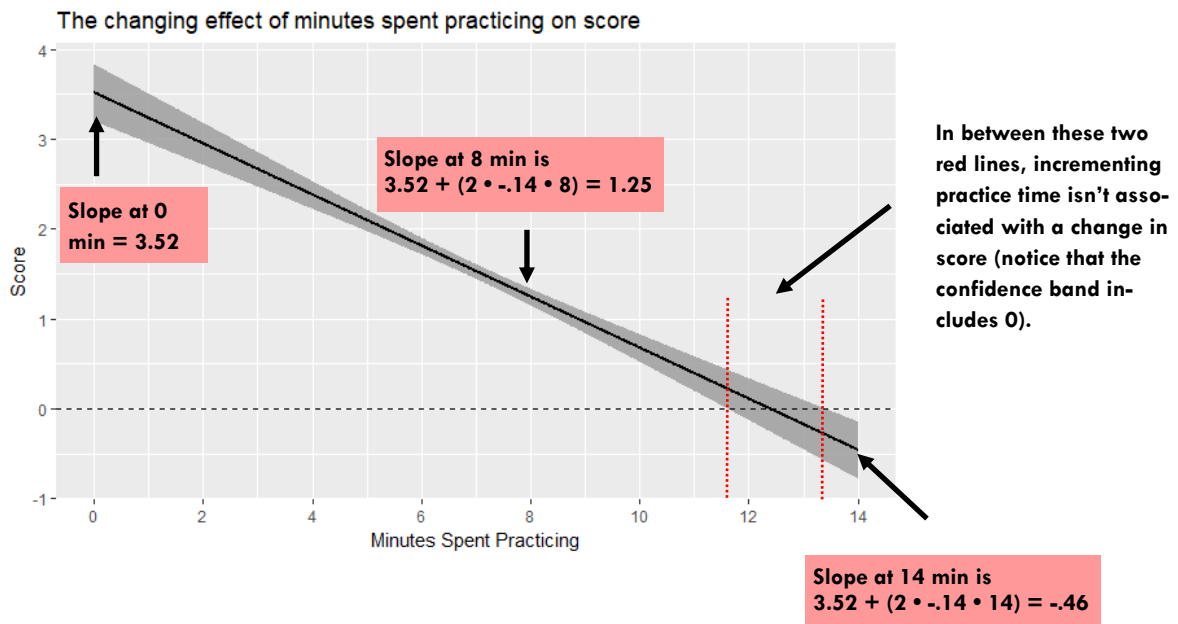
# Calculate vertex
vertex = -lm_quad$coefficients["practice"]/(2*lm_quad$coefficients["practice2"])
print(vertex)

```



## Descriptive Output and Graph

| practice | tangent_slope | se         | lwr        | upr        | tstar      |
|----------|---------------|------------|------------|------------|------------|
| 0        | 3.5166944     | 0.16022398 | 3.8410509  | 3.1923379  | 21.9486141 |
| 2        | 2.9488448     | 0.11851917 | 2.7089152  | 3.1887743  | 24.8807416 |
| 4        | 2.3809951     | 0.07935262 | 2.5416361  | 2.2203542  | 30.0052495 |
| 6        | 1.8131455     | 0.04921241 | 1.7135202  | 1.9127708  | 36.8432613 |
| 8        | 1.2452959     | 0.04921241 | 1.3449212  | 1.1456706  | 25.3045116 |
| 10       | 0.6774463     | 0.07935262 | 0.5168053  | 0.8380872  | 8.5371632  |
| 12       | 0.1095966     | 0.11851917 | 0.3495261  | -0.1303329 | 0.9247165  |
| 14       | -0.4582530    | 0.16022398 | -0.7826095 | -0.1338965 | -2.8600774 |



**Write Up:**

We assessed the effect of practice time on test performance. Forty subjects were randomly assigned to practice between 0 and 14 minutes (5 subjects each in increments of 2 minutes). After practicing, each was administered a visual discrimination test. The number of correct answers (out of 25) was recorded. We hypothesized a curvilinear relationship, in which the beneficial effect of additional practice would dissipate at higher levels of practice. That is, that each additional minute of practice would result in a larger increase when practice time was short than when practice time was long. To test our hypothesis, we regressed test scores on polynomial specifications of minutes spent practicing. We tested a linear, quadratic and cubic model. The results are presented in Table 1. The quadratic model provided the best fit as the quadratic term was significantly different from zero, but the cubic term in the cubic model was not significantly different from zero. Figure 1 presents the data with the quadratic best fit line. The results of the study support our hypothesis, practice time is significantly related to test performance in a curvilinear fashion. There is indeed a positive effect of more practice time on test performance, but the beneficial effect is attenuated as practice time increases, and appears to dissipate, and perhaps become iatrogenic at about 12 minutes.

**Table 1: Results of polynomial regression models**

|                  | Linear Model         | Quadratic Model       | Cubic Model          |
|------------------|----------------------|-----------------------|----------------------|
| Intercept        | 5.678 ***<br>(0.852) | 1.703 ***<br>(0.480)  | 1.988 ***<br>(0.537) |
| Practice         | 1.529 ***<br>(0.102) | 3.517 ***<br>(0.160)  | 3.144 ***<br>(0.358) |
| Practice-squared |                      | -0.142 ***<br>(0.011) | -0.071<br>(0.062)    |
| Practice-cubed   |                      |                       | -0.003<br>(0.003)    |
| N                | 40                   | 40                    | 40                   |
| R-Squared        | 0.856                | 0.974                 | 0.975                |
| F statistic      | 225.462              | 686.706               | 462.622              |
| P value          | 0.000                | 0.000                 | 0.000                |

Tabled values are unstandardized regression coefficients and (standard errors). \*\*\*  $p < 0.001$ ; \*\*  $p < 0.01$ ; \*  $p < 0.05$ .

**Figure 1**

Does more practice equal better score?

Overlay quadratic curve

