Regression in Practice

Regression in Practice

- Regression Statistics
- Regression Diagnostics
- Data Transformations

Regression Statistics: How Good is a Fit?

Ice Cream Sales vs. Temperature

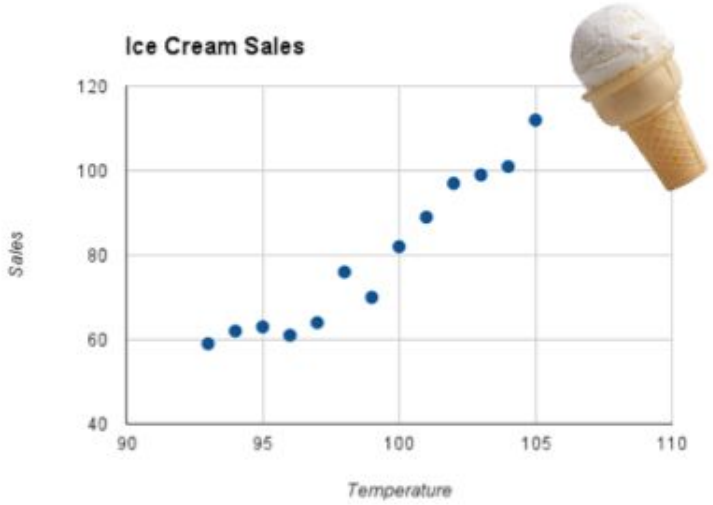
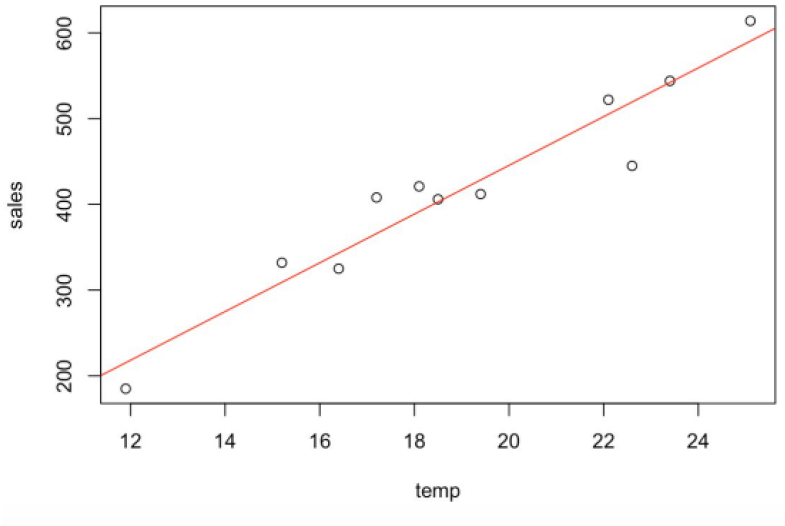


Image source



Linear Regression in R

```
> summary(lm(sales ~ temp))

Call:
lm(formula = sales ~ temp)

Residuals:
    Min      1Q  Median      3Q     Max
-74.467 -17.359   3.085  23.180  42.040

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -122.988     54.761  -2.246   0.0513 .
temp          28.427      2.816  10.096 3.31e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 35.07 on 9 degrees of freedom
Multiple R-squared:  0.9189,  Adjusted R-squared:  0.9098
F-statistic: 101.9 on 1 and 9 DF,  p-value: 3.306e-06
```

Interpreting the standard error: 95% of the observations should fall within +/- 1.96 standard deviations of the regression line, which for regression, defines a prediction interval

t-test

- Null hypothesis: The value of the coefficient is zero
- *t*-value: coefficient estimate divided by the SE
- `Pr(>|t|)` is a *p*-value

Linear Regression in R

```
> summary(lm(sales ~ temp))

Call:
lm(formula = sales ~ temp)

Residuals:
    Min      1Q  Median      3Q     Max
-74.467 -17.359   3.085  23.180  42.040

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -122.988     54.761  -2.246   0.0513 .
temp          28.427      2.816  10.096 3.31e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 35.07 on 9 degrees of freedom
Multiple R-squared:  0.9189,  Adjusted R-squared:  0.9098
F-statistic: 101.9 on 1 and 9 DF,  p-value: 3.306e-06
```

Goodness of fit measures

- Residual standard error
- R^2 and adjusted R^2
- F statistic

Anatomy of Regression Errors

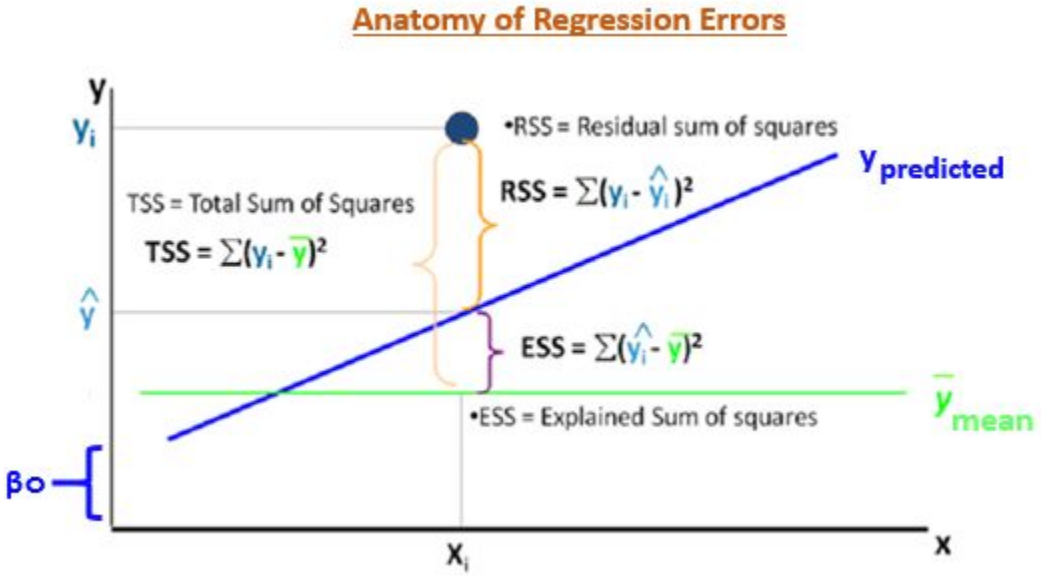


Image Source

R^2: Coefficient of Determination

- R^2 = ESS / TSS
- Interpretations:
  - The proportion of the variance in the dependent variable that the model explains.
  - The proportion of the variance in the dependent variable that independent variable predicts.
- Higher values of R^2 are preferred to lower values.
  - Caveat: Adding additional independent variables to the model almost always increases R^2. Adjusted R^2 adjusts for this increase by penalizing model complexity.
- R^2 is so called because it is mathematically equivalent to the square of the correlation coefficient, *r*.
  - Like *r*, R^2 is valid only for linear models.

Proof that R^2 = correlation coefficient squared

Straightforward algebra!

Standard Error of the Residuals

- A residual is a difference between a fitted value and an observed value.
- The residual error (RSS) is the sum of the squared residuals.
  - Intuitively, RSS is the error that the model does not explain.
- It is a measure of how far the data are from the regression line (i.e., the model), on average, expressed in the units of the dependent variable.
- The residual standard error is roughly the square root of the average residual error (RSS / n).
  - Technically, it's not $\sqrt{(RSS / n)}$, it's $\sqrt{(RSS / (n - 2))}$; it's adjusted by degrees of freedom.

F Statistic

- The *F* statistic is the ratio of the error explained by the model to the residual error: ESS / RSS (adjusted by degrees of freedom).
- The *F* statistic is used to test the predictive power of the independent variable. In particular, does a non-zero coefficient improve the model?
- The *F* statistic extends naturally to multiple regression, where it is used to test the predictive power of *all* independent variables, to see how unlikely it is that *all* regression coefficients equal zero.
- Prob(*F*) is the probability that the null hypothesis is true: i.e., that *all* the regression coefficients are zero.

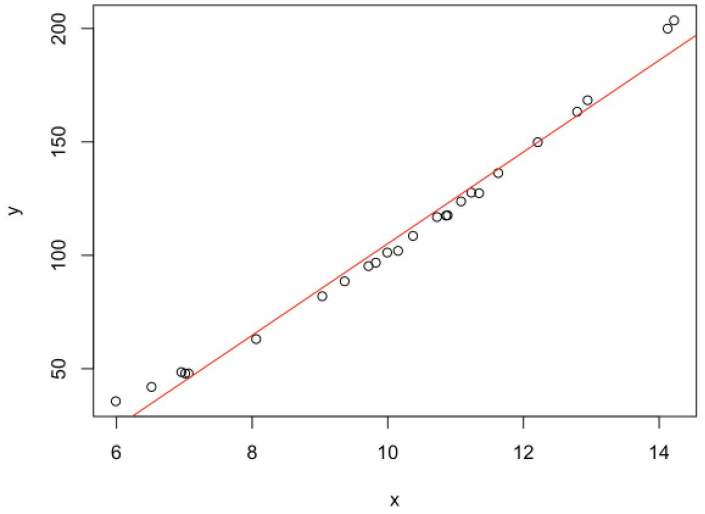
Violations are not always easy to detect!

- Assume a nonlinear relationship:
  - `x <- rnorm(25, 10, 2)`
  - `e <- rnorm(25, 0, 1)`
  - `y <- x ** 2 + e`
- Build a linear model:

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -99.4095     3.9529  -25.15   <2e-16 ***
x            20.3762     0.3833   53.16   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.792 on 23 degrees of freedom
Multiple R-squared:  0.9919,    Adjusted R-squared:  0.9916
F-statistic:  2826 on 1 and 23 DF,  p-value: < 2.2e-16
```

- R tells us the results are statistically significant, but a linear model is incorrect!

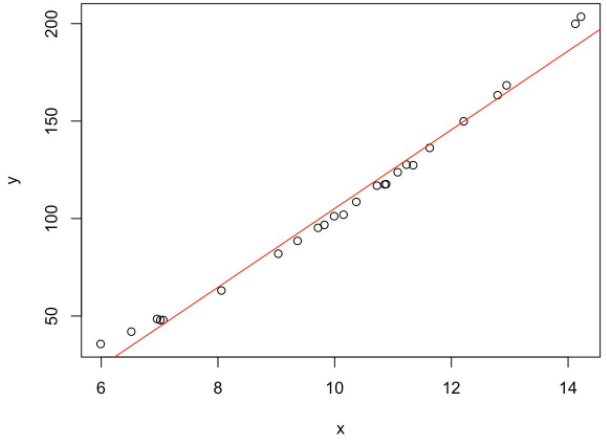


Standard Deviation = 2

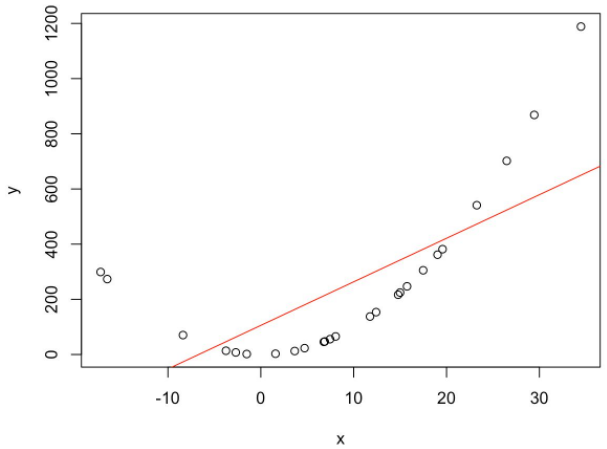
Regression Diagnostics

Checking Linearity

- Always begin an analysis with EDA
- Plot dependent vs. independent variables
  - Is a linear relationship plausible?
  - Are there outliers?

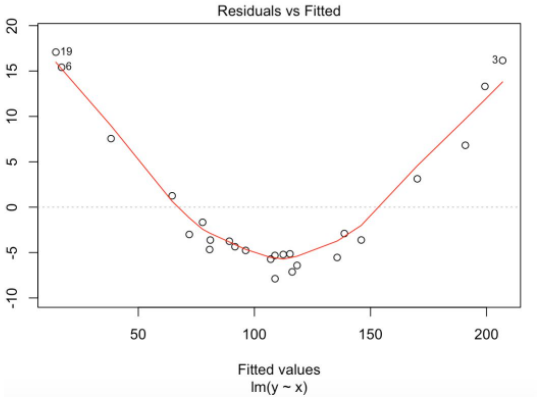


- Example: a nonlinear relationship
  - `x <- rnorm(25, 10, 10)`
  - `e <- rnorm(25, 0, 1)`
  - `y <- x ** 2 + e`
- And a linear model

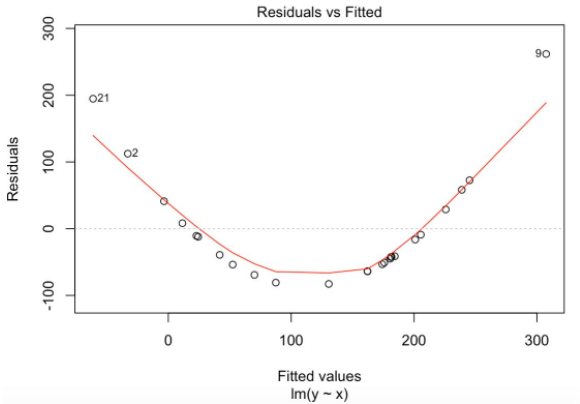


Residual Plots

- A residual is the difference between a fitted and an observed value.
- A plot of residuals vs. fitted values should look like a formless cloud.
- If a model accurately captures the structure in the data, then all that should remain after the model is through making its predictions is random noise!



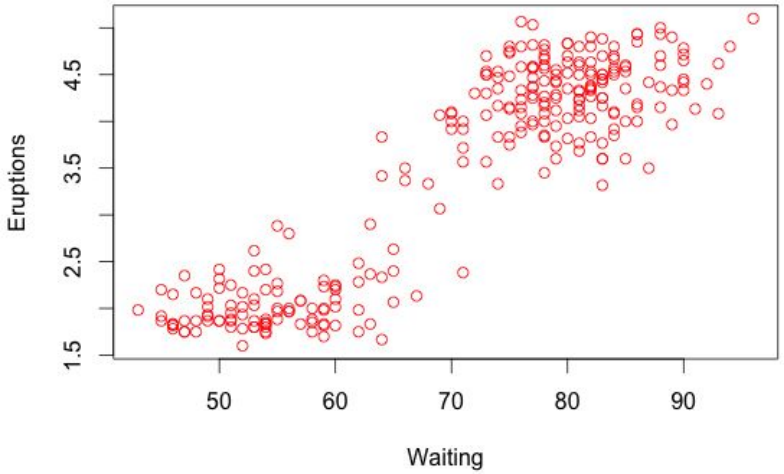
Standard Deviation = 2



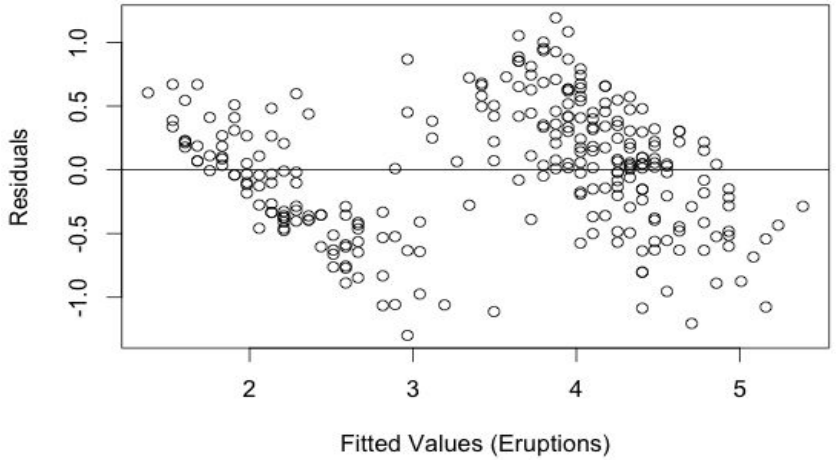
Standard Deviation = 10

Scatter and Residual Plots of Old Faithful

- The scatter plot shows two clusters, one with low eruptions and low waiting time, and another with high eruptions and high waiting time



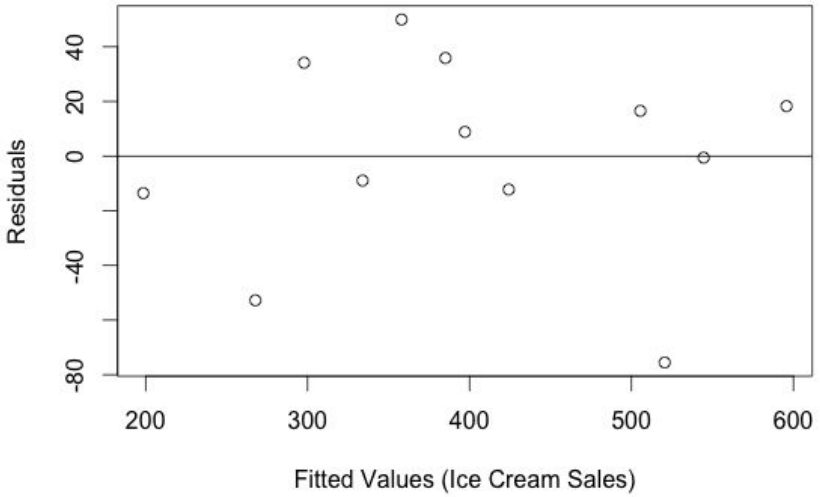
- The scatter plot is not a formless cloud
- There are diagonal stripes, suggesting a nonlinear pattern in the data



A Residual Plot for Ice Cream Sales

- This residual plot is a formless cloud.
- Thus, a linear model seems suitable.

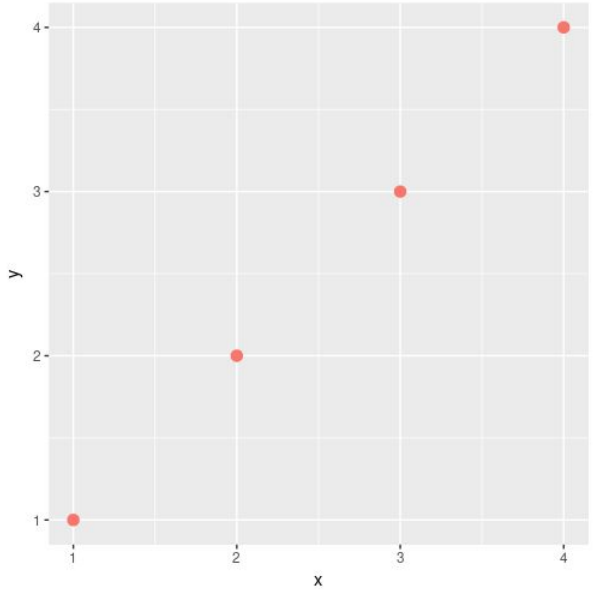
```
> fit <- lm(sales ~ temperature)
> plot(residuals(fit) ~ fitted(fit),
ylab = "Residuals", xlab = "Fitted
Values") + abline(a = 0, b = 0)
```



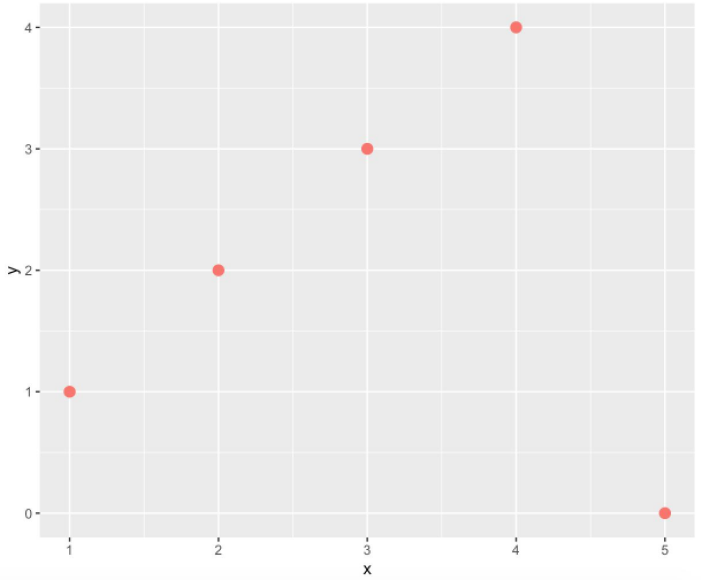
If there is non-randomness in your residuals, then your model is not explaining all that it can.

Outliers can gravely impact correlation

```
> cor(line$x, line$y)
[1] 1
```

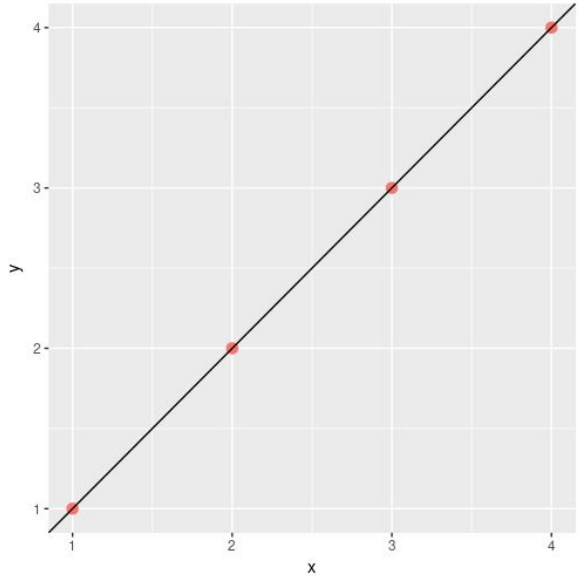



```
> cor(outlier$x, outlier$y)
[1] 0
```

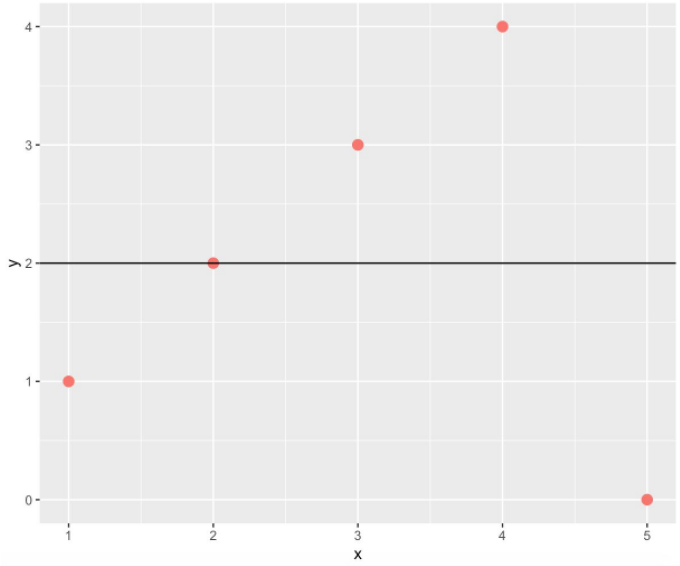


Outliers can gravely impact regression

```
> line_model <- lm(line$y ~ line$x))
> plot(line_model)
> abline(line_model, col = “red”)
```

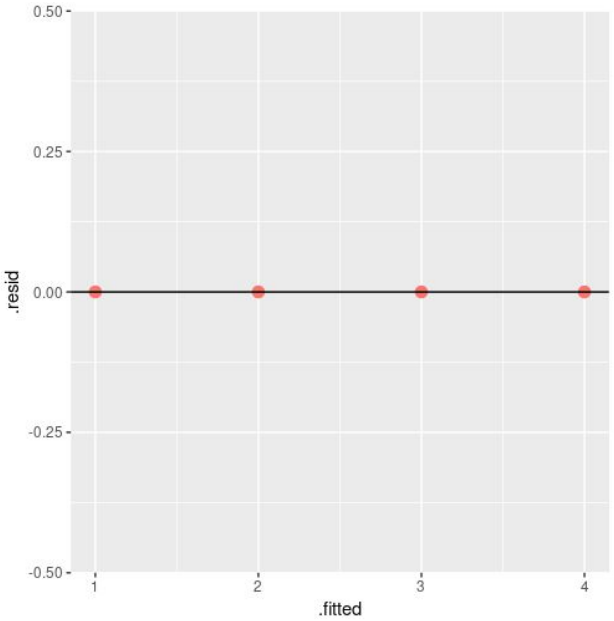



```
> outlier_model <- lm(outlier$y ~ outlier$x))
> plot(outlier_model)
> abline(outlier_model, col = “red”)
```

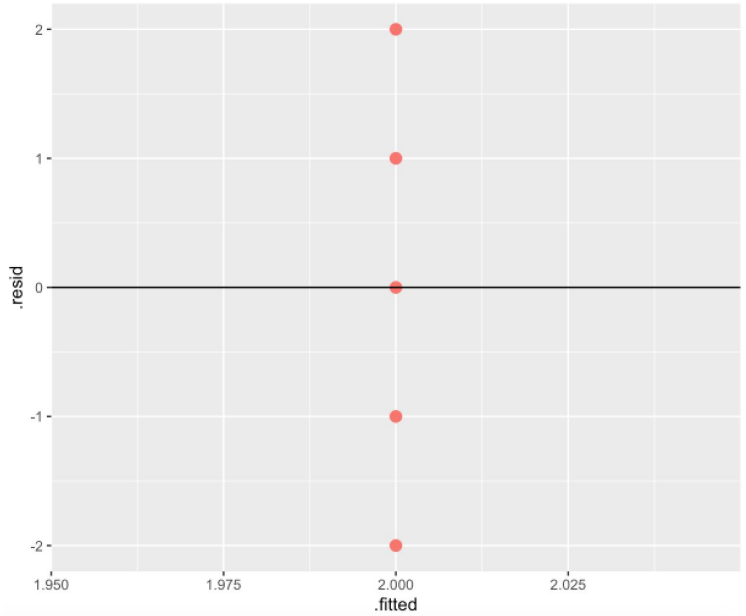


Their impact can be seen in residual plots

```
> plot(residuals(line_model) ~ fitted(line_model))
```




```
> plot(residuals(outlier_model) ~ fitted(outlier_model))
```



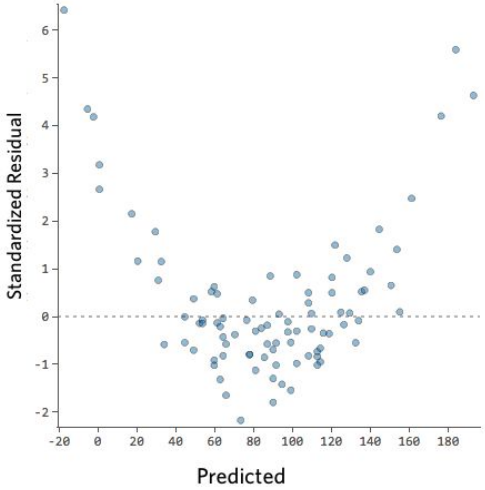
How to handle outliers

- First, try to determine whether their removal can be justified.
  - Was a clerical error made in entering the data? Correct any obvious mistakes.
  - Check whether they might result from a failure to abide by the correct experimental procedure.
  - Look for any other possible explanations.
- If there are no convincing justifications for their removal, conduct analyses with and without the outliers, and report all results. Hopefully, their impact is minor.

Standardized Residuals

- A standardized residual is a residual divided by the standard deviation of the residuals.
- A plot of standardized residuals vs. fitted values should (also) look like a formless cloud.
- Standardizing residuals can be useful for identifying outliers.

In case it isn't obvious, these plots are NOT formless clouds!



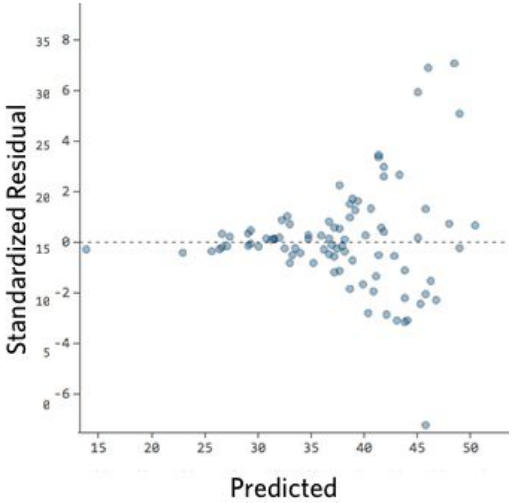


Image Source

Summary

In a well-behaved plot of residuals vs. fitted values:

- The residuals bounce around the *x*-axis randomly; they don't smile or frown.
- No residuals stand out from the others, so there are no obvious outliers.
- They form a band around the 0 line; they don't funnel in or out.

Aside: Why plot residuals vs. fitted values, and not observations?

- Because residuals and fitted values are uncorrelated by construction.
- Residuals and observations may be correlated—they both depend on observations—which would make such plots harder to interpret.

Additional Residual Plots

- Plot residuals vs. independent variables
  - Check for conditional expectation of 0
  - Check for constant conditional variance (homoskedacity)

Assumption: Errors are Homoskedastic

- Homoskedasticity means that the variance of the error term, conditional on *X*, is constant.
  - $\text{Var}[\varepsilon_i \mid X = x] = \sigma^2$
- In words, the errors should not funnel or fan, as we move along the x-axis.

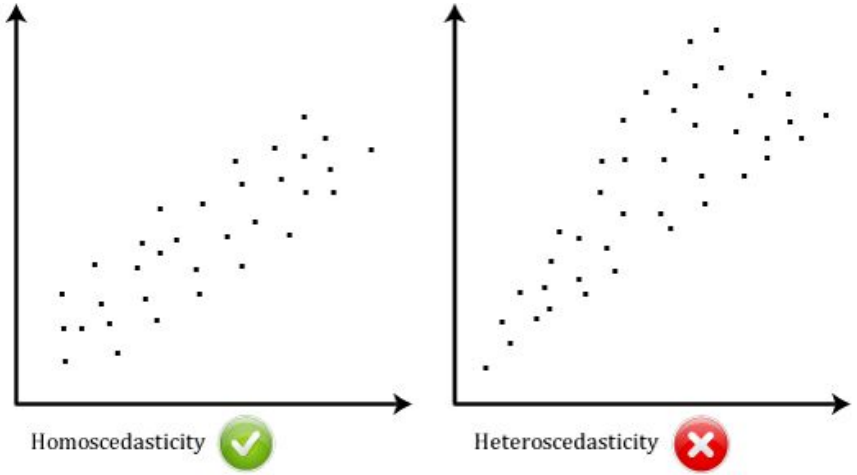


Image Source

What if Errors are Heteroskedastic?

- We can to adjust the way we calculate the standard error of our regression estimates if errors are heteroskedastic.
  - Doing this in base R is quite hard, but someone made an open-source function to take care of this for us; the function can be found here.
- When we build a linear model, we will store it first.
  - `mod <- lm(y ~ x)`
- Then, we can use the `summary.lm` function in the library above.
  - `summary.lm(mod, robust = TRUE)` will make a correction in the calculation for standard error of the regression estimates when errors are heteroskedastic.

Data Transformations

Why are linear models so popular?

- Because linear models are simple!
  - Perhaps the simplest non-trivial relationship imaginable.
- Because true relationships are often close to linear in the domain of interest.
  - They are always linear on a small enough domain, so multiple simple linear models can be concatenated into one more complicated model.
- Because variables can be transformed to make relationships linear.
  - If your data suffer from non-linearity, transform the independent variables.
  - If your data suffer from heteroskedasticity, transform the dependent variable.

Data Transformations

Linear regression requires that the relationship between *x* and *y* be linear. What if the relationship is not linear? We can transform the data!

How?

- Log transformation
- Exponential transformation
- Polynomial (power) transformations (square root, square, cube, etc.)

What?

- We can transform the independent variable only.
- We can transform the dependent variable only.
- We can transform both the independent and dependent variables.

Logarithms

- To exponentiate is to raise a number to a power. For example, $10^2 = 100$.
- Calculating a logarithm is the inverse (opposite) of exponentiating:
  - A logarithm is an exponent that yields a number.
  - E.g., the log (short for logarithm) of 100 is 2.
  - To take logs requires a base, which we assumed to be 10 in this example.
- The function e^x is special (because its derivative is itself).
- Taking logs base e is also special, and is called the natural log (written $\ln(x)$).
  - N.B. R (and most programming languages) default to computing $\ln(x)$ if no base is specified.

Data Transformations

- Transforming a non-linear (exponential) relationship into a linear one

$$y = x^n$$

$$\log(y) = \log(x^n)$$
$$= n\log(x)$$

slope

$$y = cx^n$$

$$\log(y) = \log(cx^n)$$
$$= \log(c) + \log(x^n)$$
$$= \log(c) + n\log(x)$$

intercept

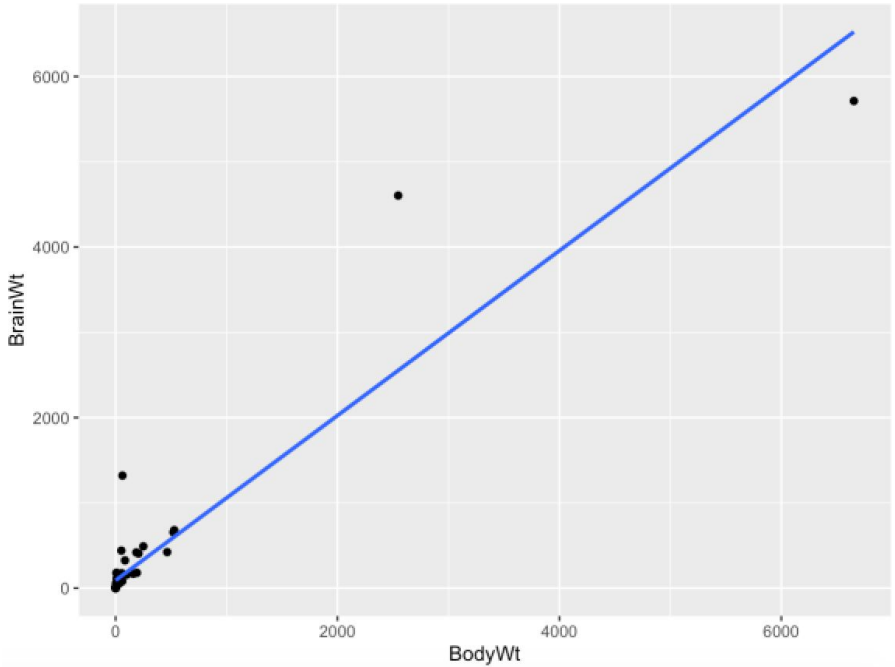
slope

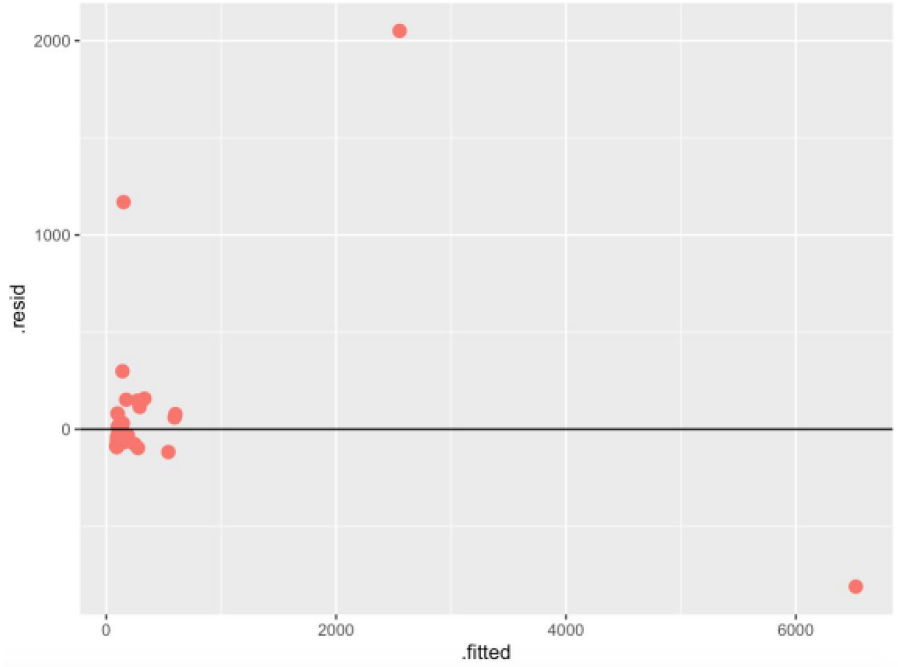
An Example: Mammalian Brain Weights

Do mammals with heavier bodies have heavier (and hence, bigger) brains?

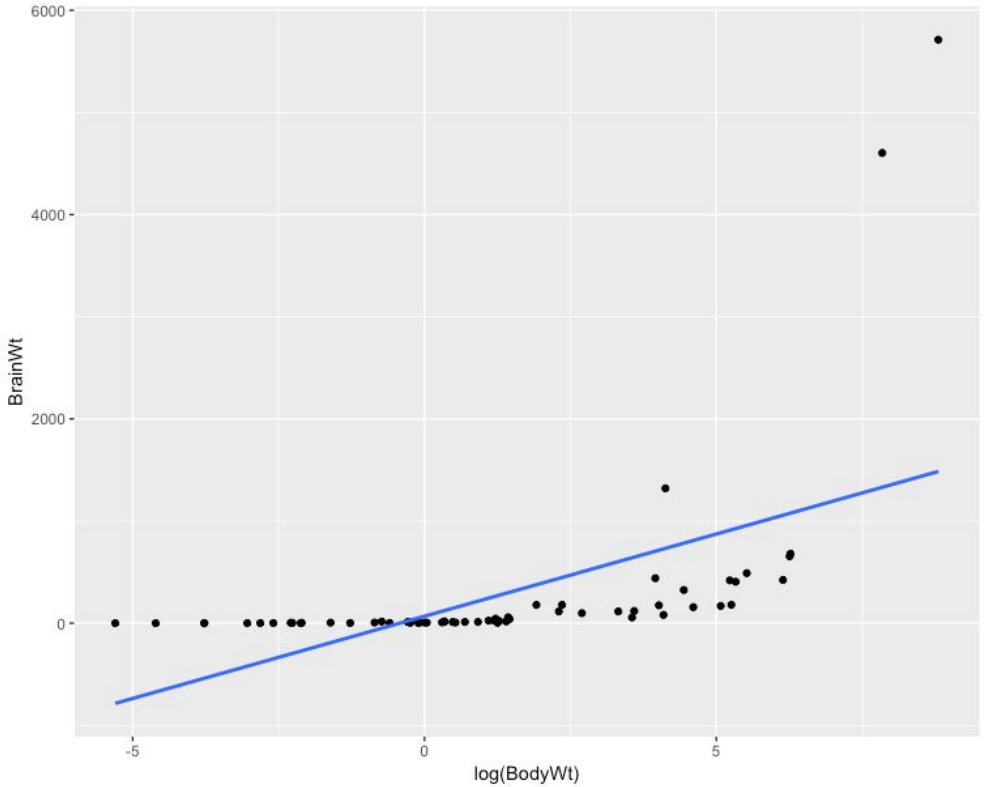
```
> brains
                             X  BrainWt   BodyWt
1                   Arctic fox   44.500    3.385
2                   Owl monkey   15.499    0.480
3                       Beaver    8.100    1.350
4                          Cow  423.012  464.983
5                    Gray wolf  119.498   36.328
6                         Goat  114.996   27.660
7                     Roe deer   98.199   14.831
8                   Guinea pig    5.500    1.040
9                       Vervet   57.998    4.190
10                   Chinchilla    6.400    0.425
11              Ground squirrel    4.000    0.101
12    Arctic ground_squirrel    5.700    0.920
13 African giant_pouched_rat    6.600    1.000
```

BodyWt vs. BrainWt

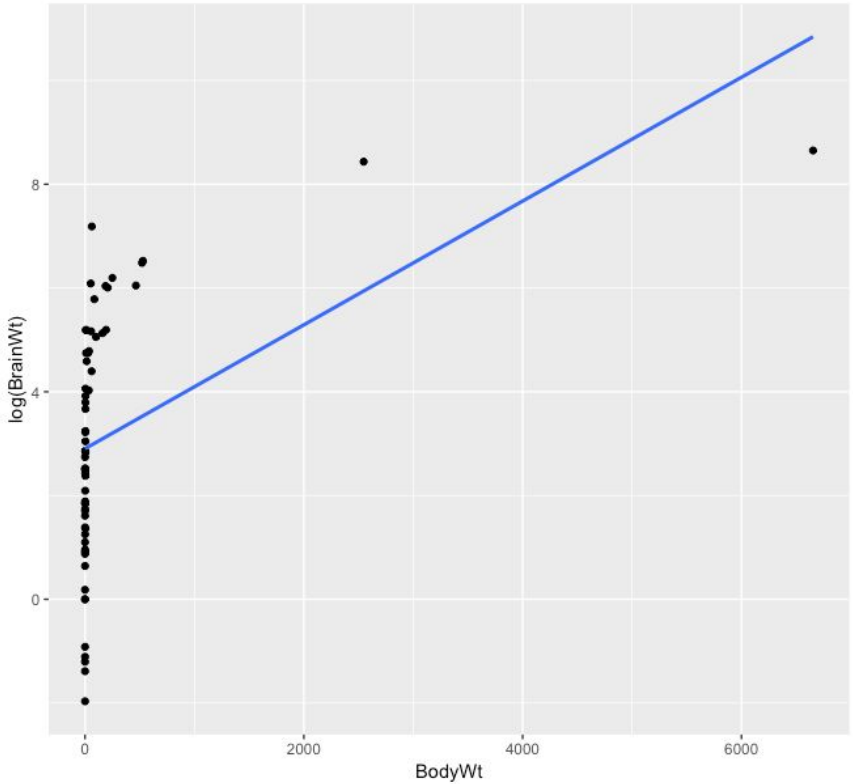




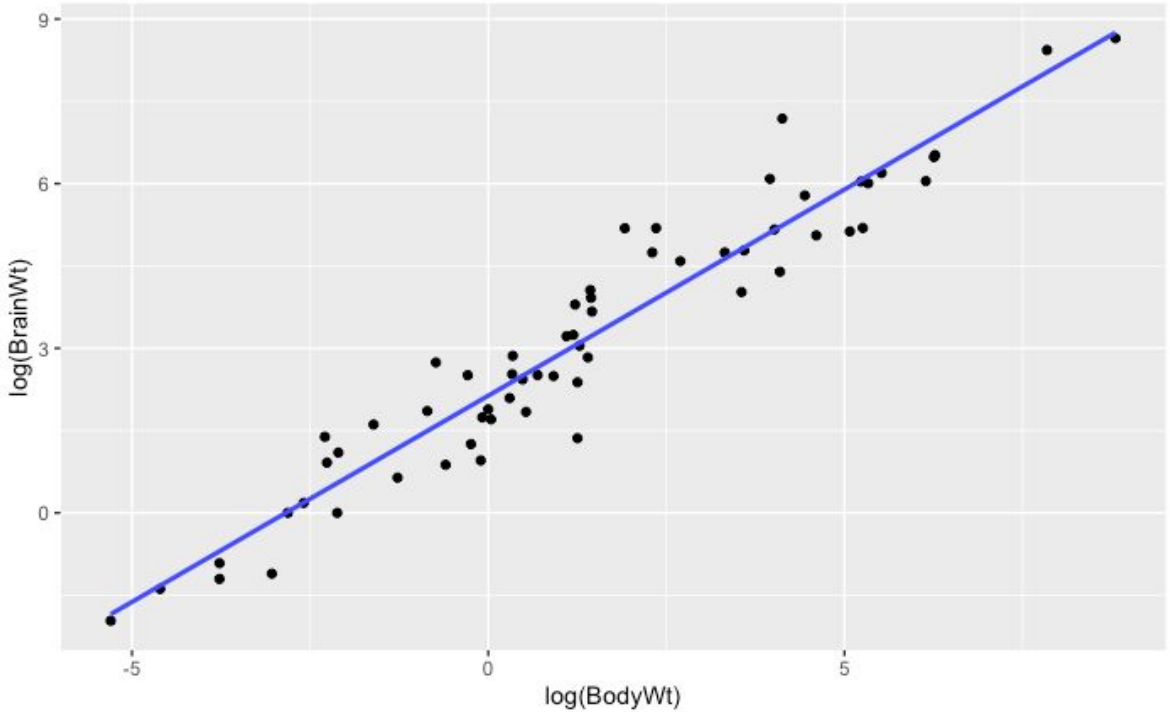
`BrainWt` vs. `log(BodyWt)`



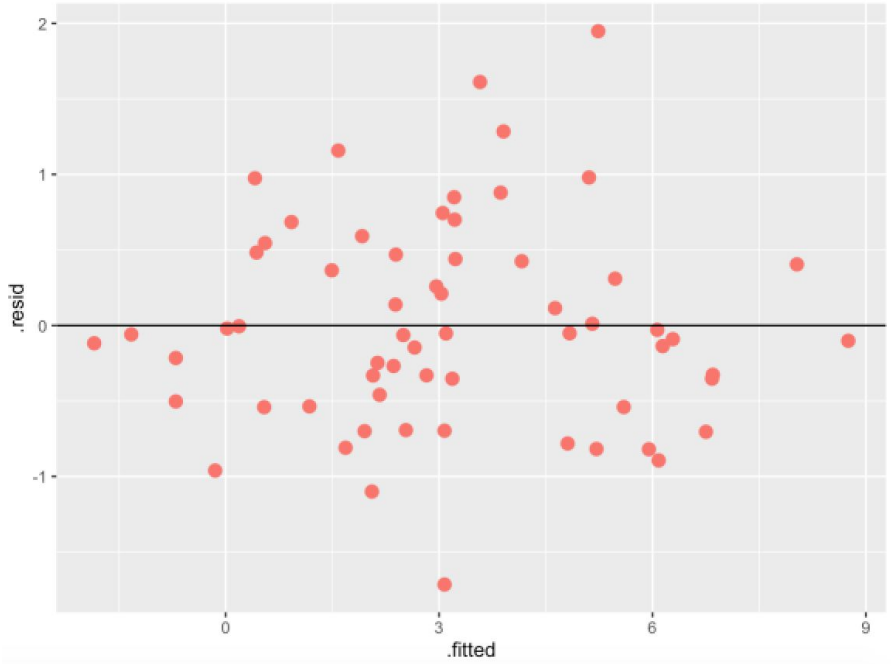
log(BrainWt) vs. BodyWt



log(BrainWt) vs. log(BodyWt)



Residual Plot

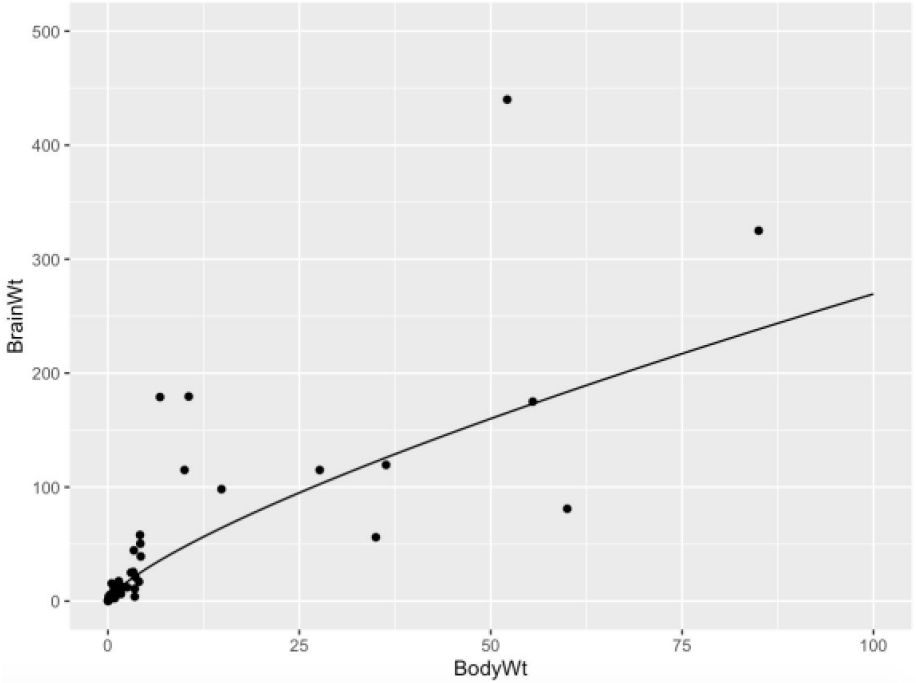


This residual plot looks much better!

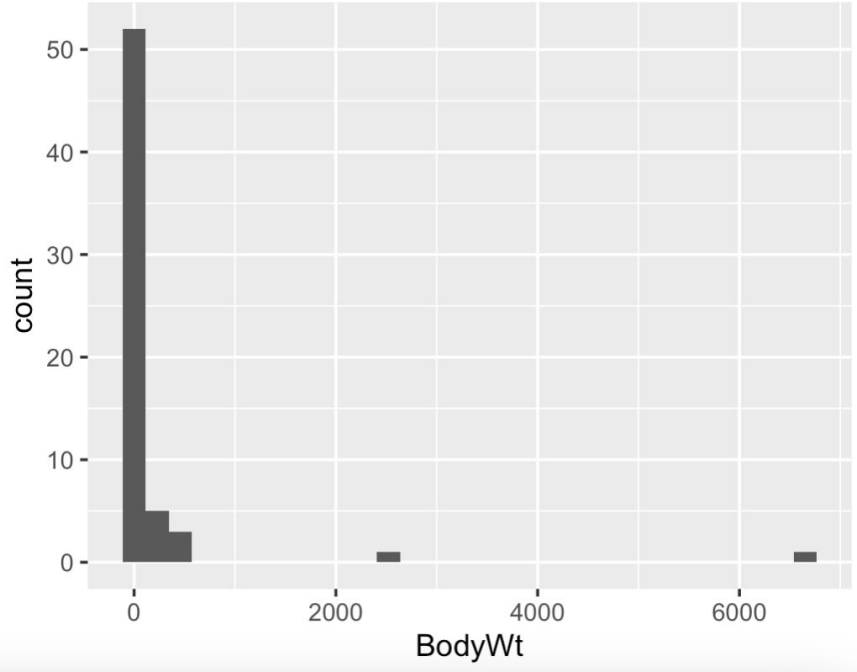
“Linear” Model

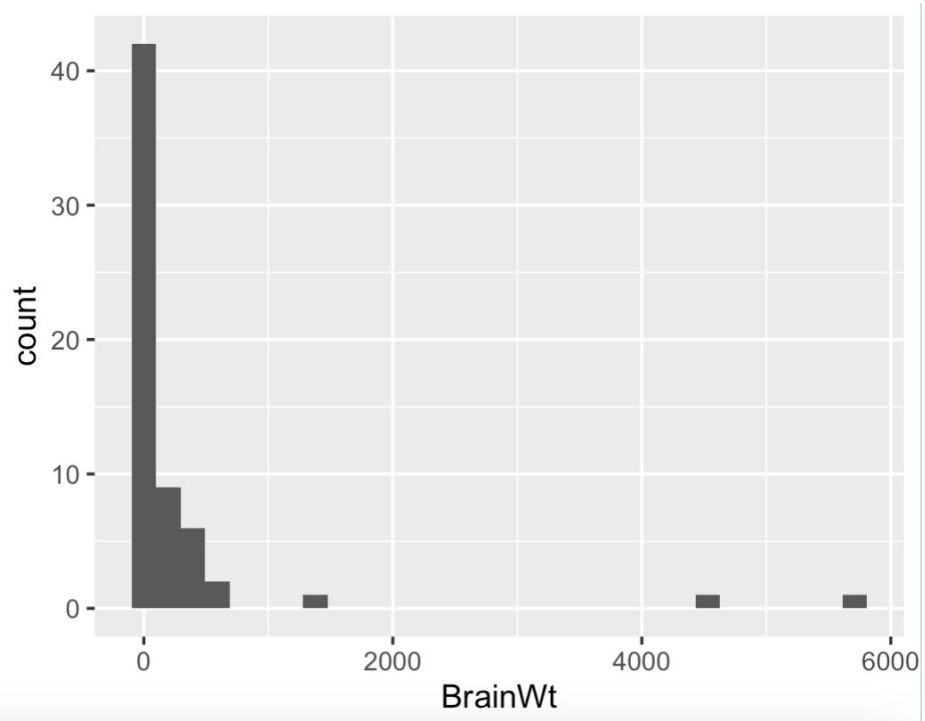
```
my_model <- lm(log(BrainWt) ~ log(BodyWt))

ggplot(data = brains,
  mapping = aes(BodyWt, BrainWt)) +
  geom_point() +
  xlim(c(0, 100)) + ylim(c(0, 500)) +
  stat_function(fun = function(x) (exp(1)
** my_model$coefficients[1] * x **
my_model$coefficients[2]))
```

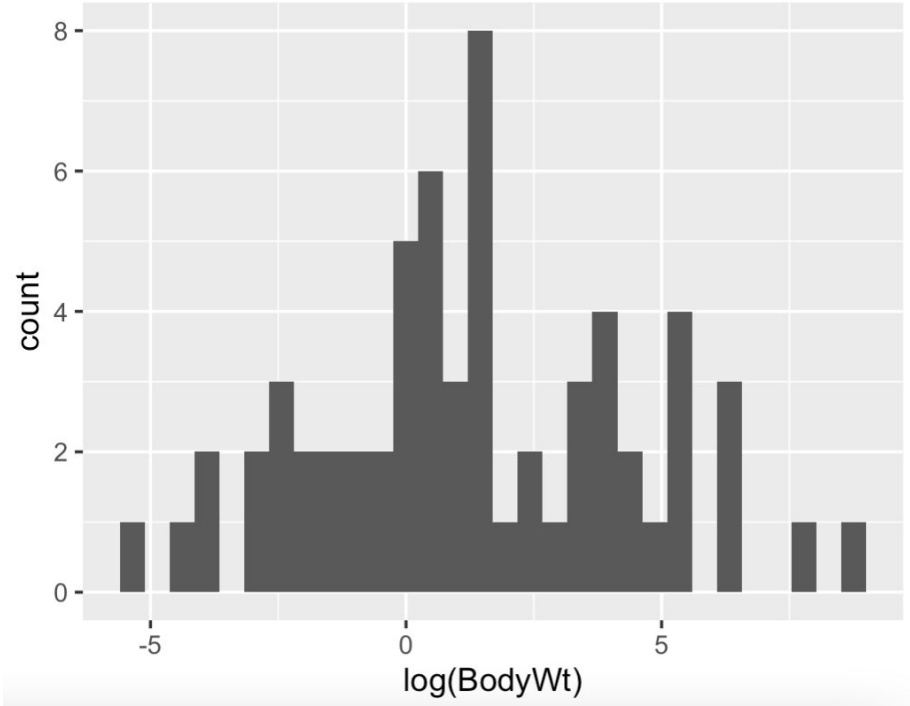


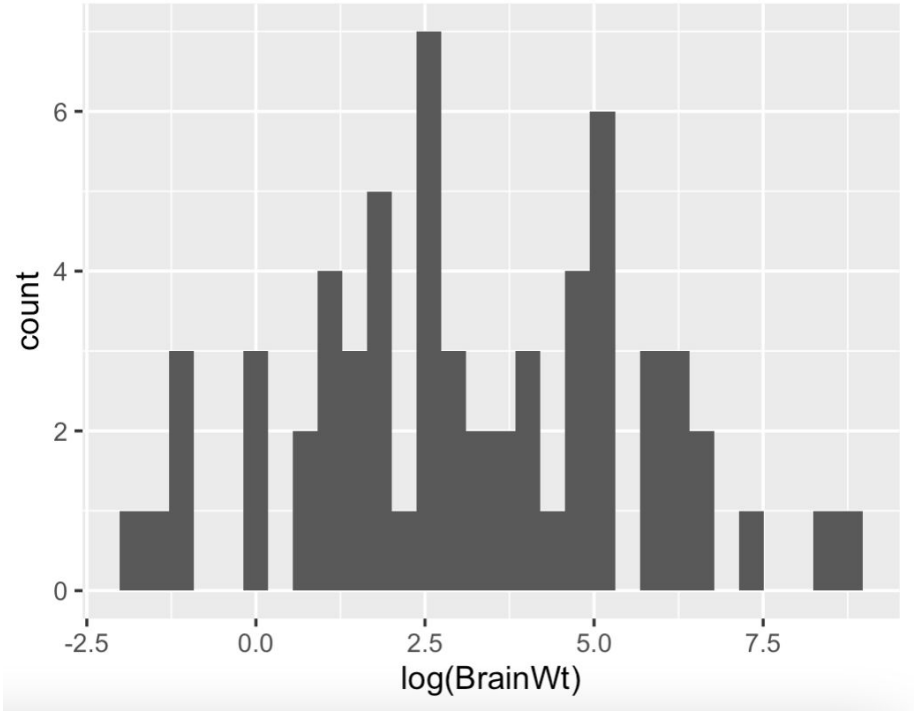
BodyWt and BrainWt





log(BodyWt) and log(BrainWt)





Extras

Relationships

- $Y = a + bX$
  - Linear relationship: Y increases proportionally with X.
  - A unit increase in X is associated with an additive increase in Y by b units.
- $Y = e^X$
  - Exponential relationship: Y increases “exponentially” with X.
  - A unit increase in X is associated with a factor of e increase in Y:
    a multiplicative increase in Y by e units.
- $Y = \ln(X)$
  - Logarithmic relationship: Y increases “logarithmically” with X.
  - A factor of e increase in X yields a unit increase in Y,
    because $\ln(eX) = \ln(e) + \ln(X) = 1 + \ln(X)$.

Transformations

- $Y = a + bX$
  - As X increases linearly, Y increases linearly.
  - A unit increase in X is associated with an additive increase in Y by b units.
- $\ln(Y) = a + bX$, so that $Y = e^{(a + bX)}$
  - As X increases linearly, Y increases exponentially.
  - A unit increase in X is associated with an additive increase in $\ln(Y)$ by b units.
  - In other words, a unit increase in X is associated with a factor of e^b increase in Y.
- $Y = a + b \ln(X)$
  - As X increases multiplicatively, Y increases linearly.
  - A factor of e increase in X is associated with an additive increase in Y by b units, because $\ln(eX) = \ln(e) + \ln(X) = 1 + \ln(X)$, so $a + b \ln(eX) - (a + b \ln(X)) = b$.

Transformations

- $\ln(Y) = a + b \ln(X)$, so that $Y = e^a X^b$
  - As X increases multiplicatively, Y increases polynomially.
  - A factor of k increase in X is associated with a factor of k^b increase in Y:
    a multiplicative increase in Y by k^b units, because $e^a(kX)^b - e^a X^b = k^b(e^a X^b) - e^a X^b$.
  - If $b = 3$, then doubling X (i.e., a factor of 2 increase) leads to a factor of 8 increase in Y.
  - If $b = 2$, then tripling X (i.e., a factor of 3 increase) leads to a factor of 9 increase in Y.

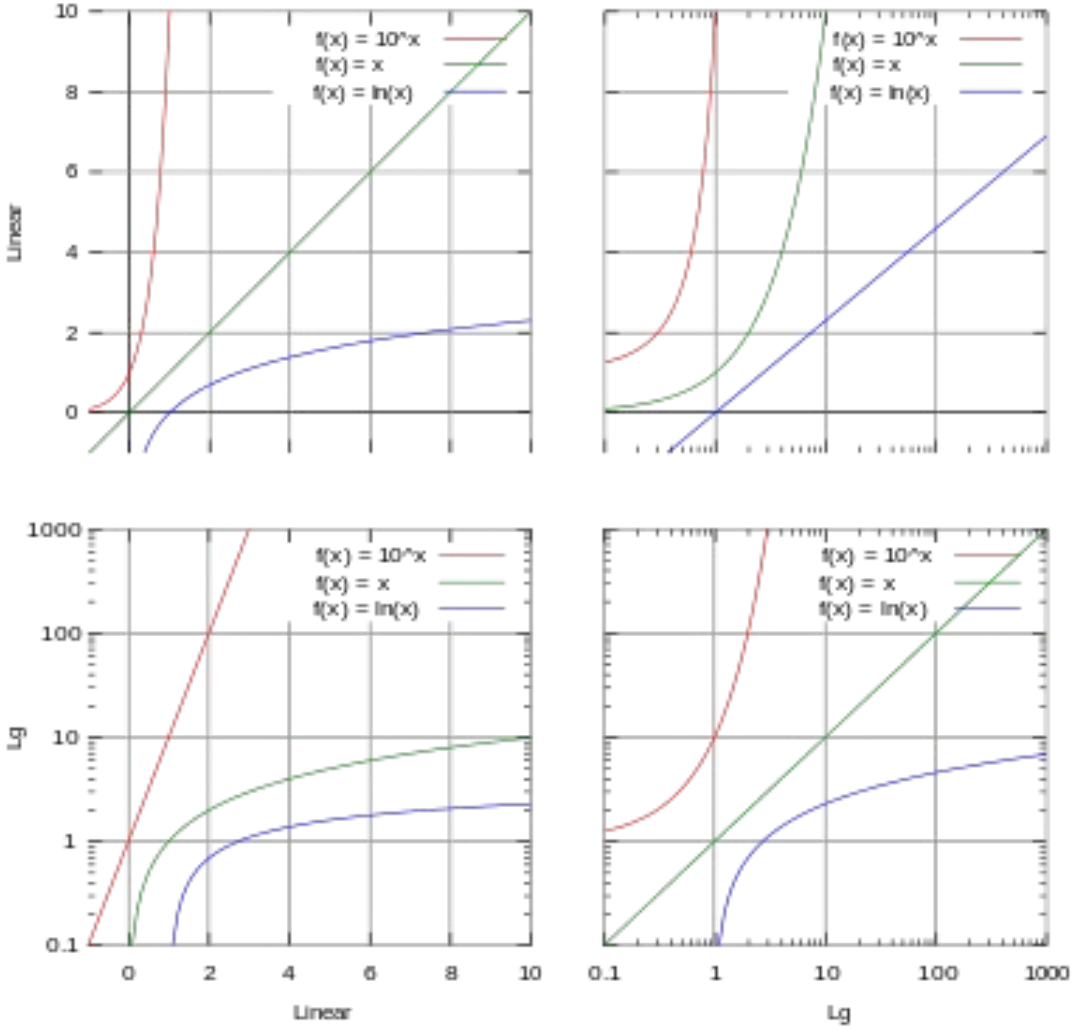
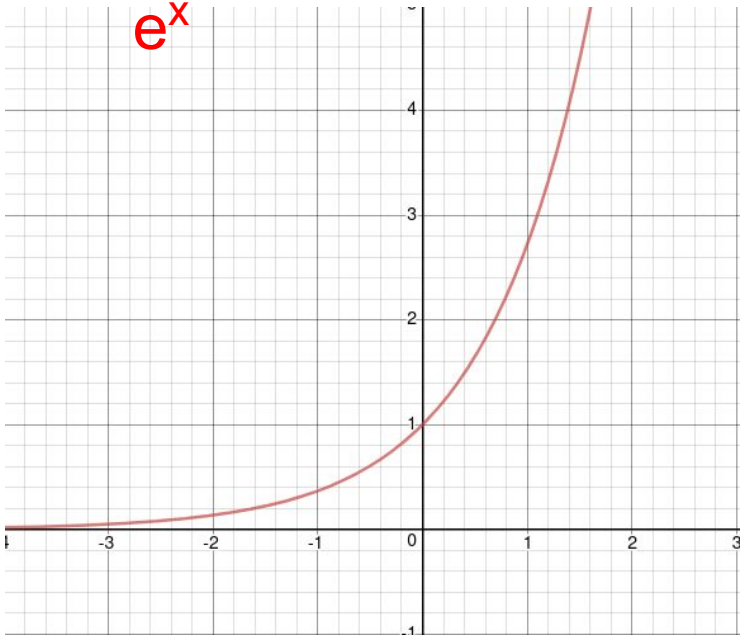
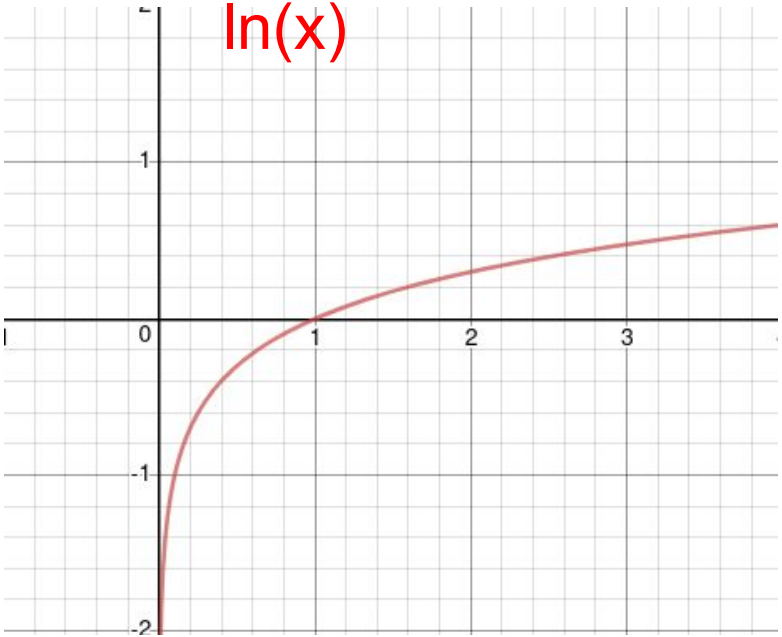


Image Source

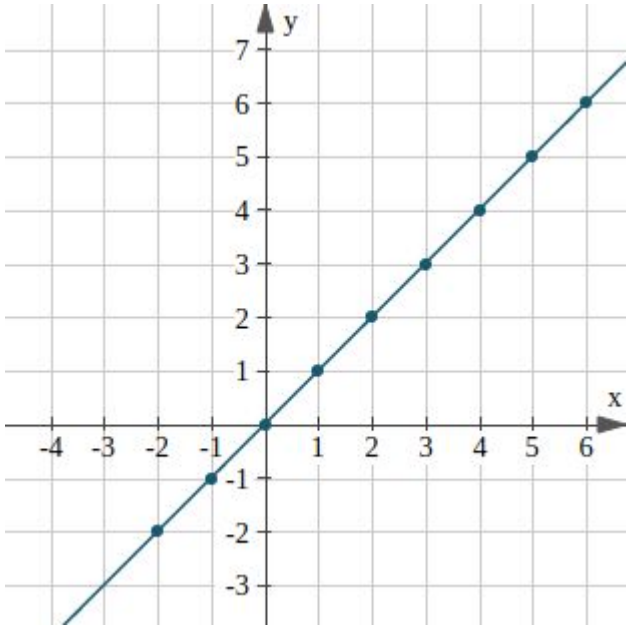
Exponentials & Logarithms, Graphically





e^x is a growth rate: after x time, growth increases by e^x ln(x) is the inverse of e^x: to grow by x takes ln(x) time

Linear Function on a Semi-log Plot



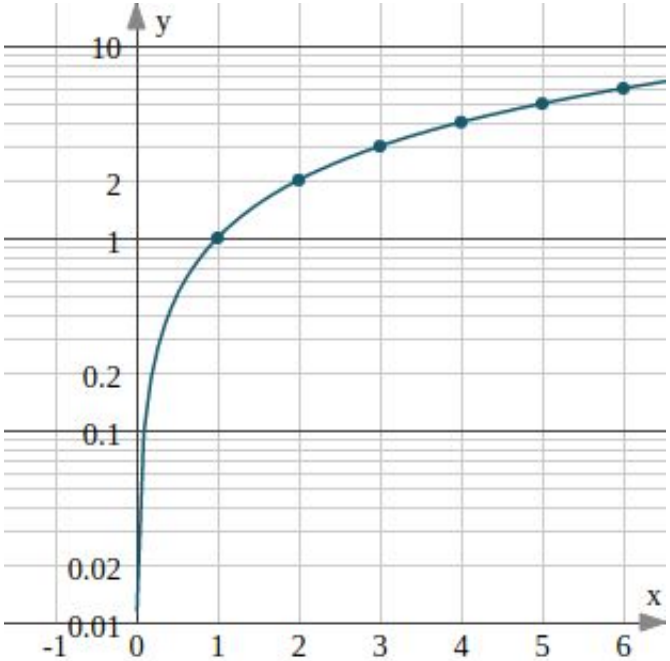
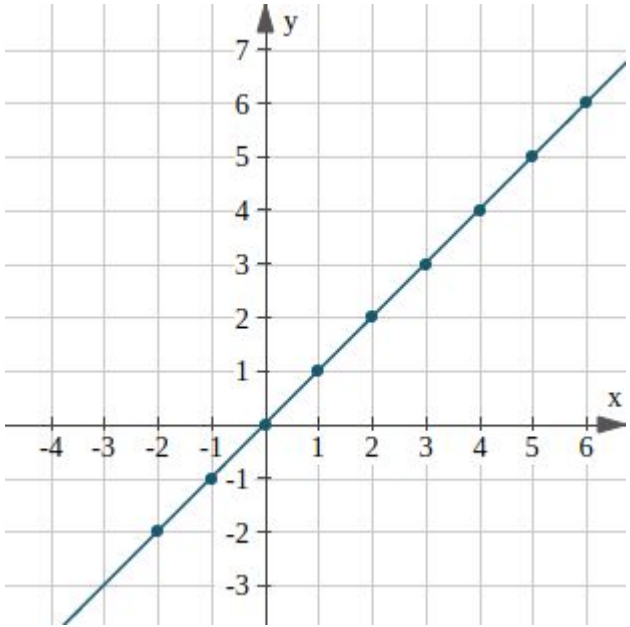


Image Source

Linear Function on a Semi-log Plot



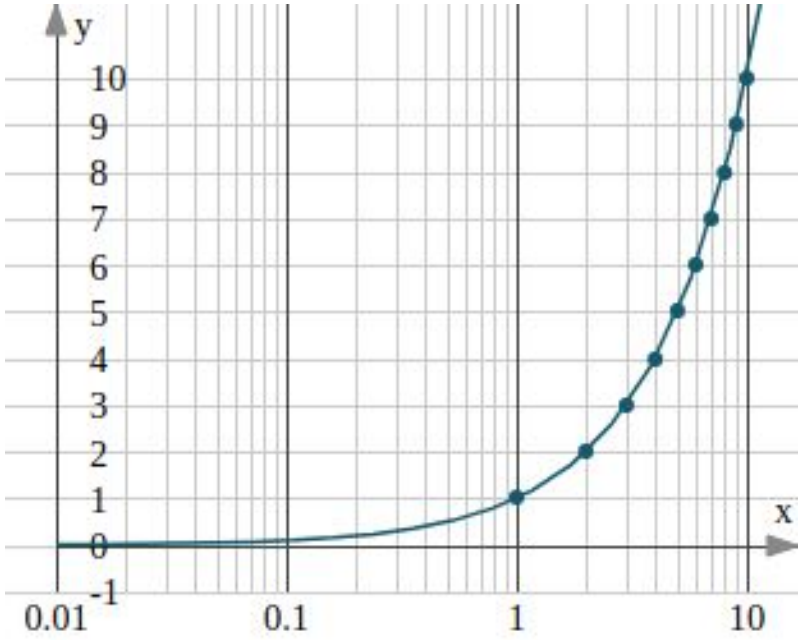


Image Source