Non-linear transformations of variables (transformations for model fit)

1. Load the `tidyverse` library.

```
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse
1.3.1 ──

## ✓ ggplot2 3.3.5     ✓ purrr   0.3.4
## ✓ tibble  3.1.5     ✓ dplyr   1.0.7
## ✓ tidyr   1.1.3     ✓ stringr 1.4.0
## ✓ readr   2.0.1     ✓ forcats 0.5.1

## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

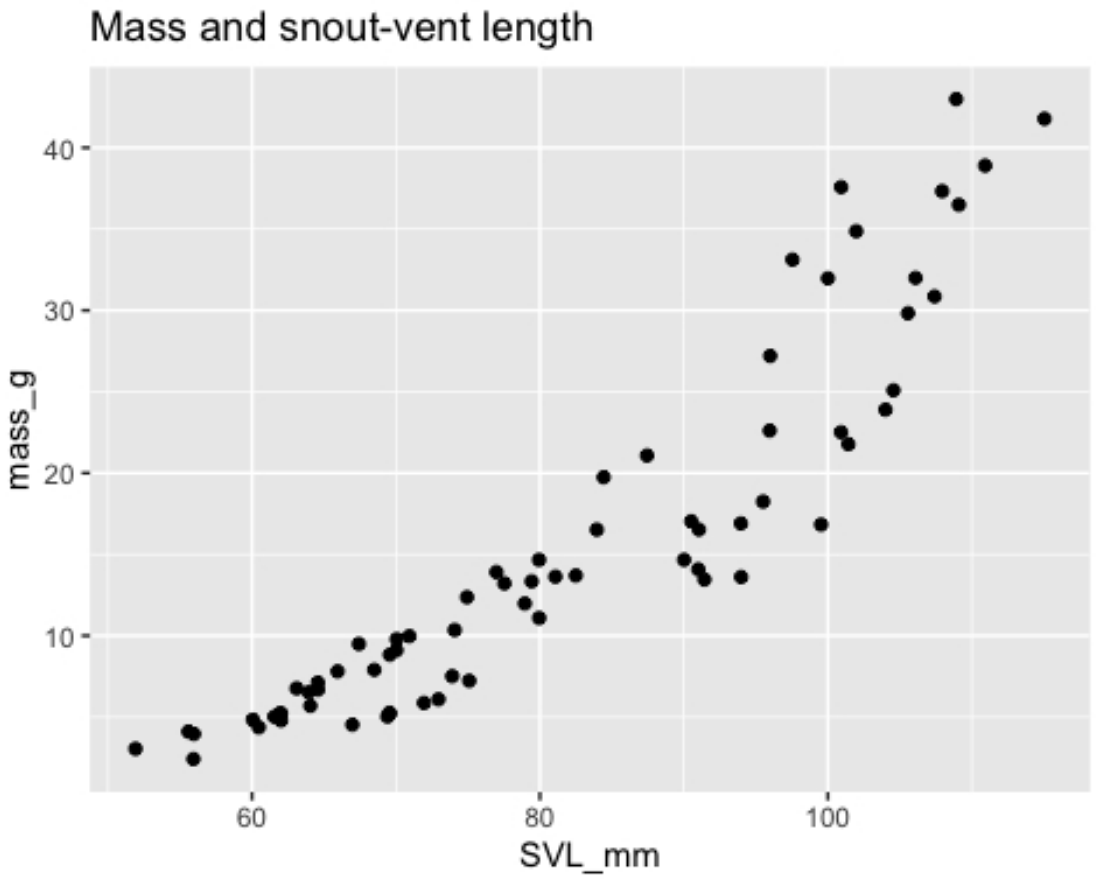
```
lizards <- read_csv('lizards.csv')
```

```
## Rows: 74 Columns: 3

## ── Column specification
──────────────────────────────────────────────────────
## Delimiter: ","
## dbl (3): mass_g, SVL_mm, type

##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.
```

```
set.seed(31878039)
my_lizards <- lizards %>%
  sample_n(70)
```

2. Use a `ggplot` to plot mass_g (y-axis) against SVL_mm (x-axis). `ggplot2` is part of `tidyverse` so you do not need to load it separately.

```
my_lizards %>% ggplot(aes(x = SVL_mm, y = mass_g)) +
 geom_jitter(width = 0.1) +
  labs(title = "Mass and snout-vent length")
```

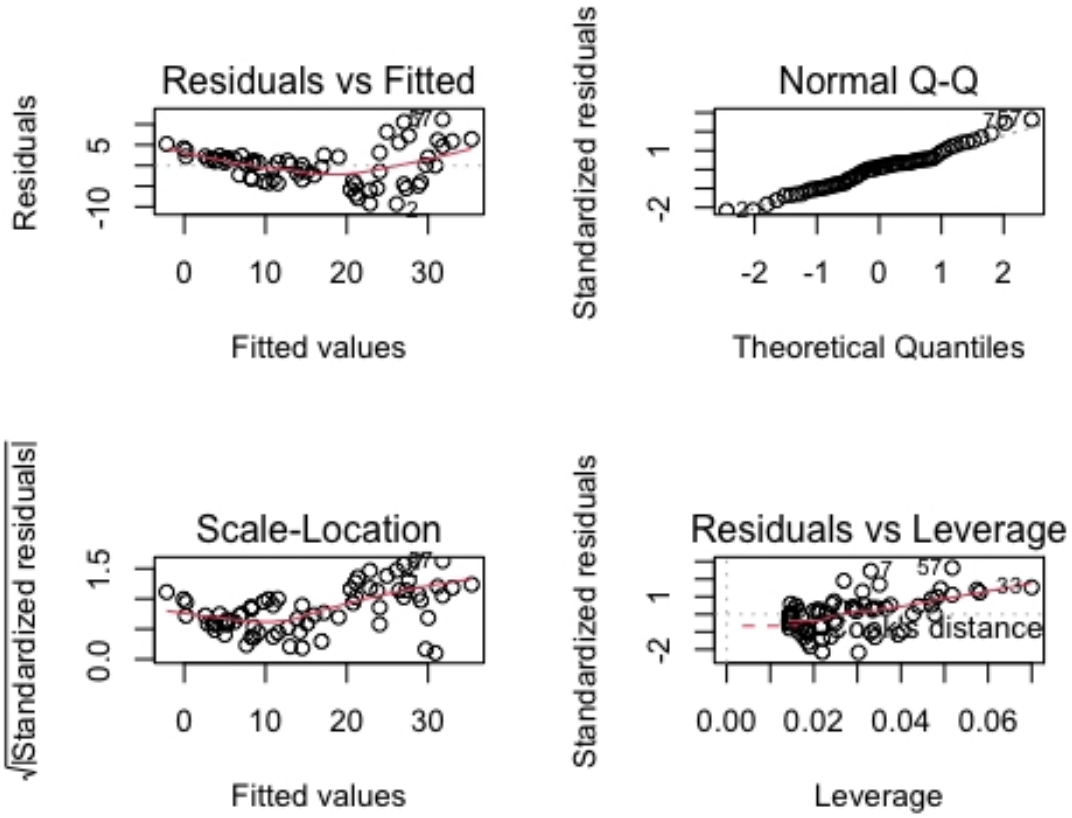


3. Create a linear model m1 for mass_g as a function of SVL_mm. Show the code used to create the model and the model summary (using summary) in your report.

```
m1 <- lm(mass_g ~ SVL_mm, data=my_lizards)
summary(m1)
```

```
##
## Call:
## lm(formula = mass_g ~ SVL_mm, data = my_lizards)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.3021 -3.3782  0.5577  2.1744 11.1949
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -33.14702    2.56179  -12.94   <2e-16 ***
## SVL_mm        0.59579    0.03074   19.38   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.348 on 68 degrees of freedom
```

```
## Multiple R-squared:  0.8467, Adjusted R-squared:  0.8445
## F-statistic: 375.7 on 1 and 68 DF,  p-value: < 2.2e-16

par(mfrow = c(2,2))
plot(m1)
```

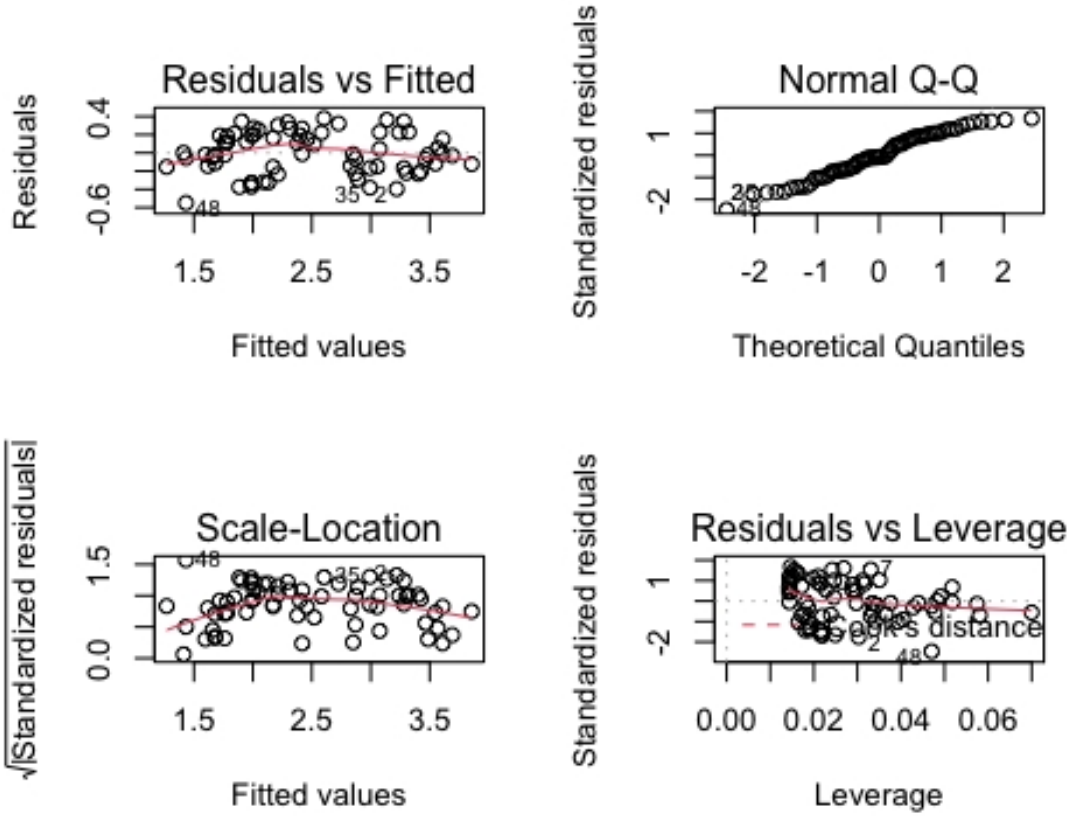


The standardised residuals do not meet the equal variance assumption as there is no constant scatter, there is a curve, and it approximates a skewed distribution. The residuals do not follow linearity due to the curve and the patten it formed. The Q-Q plot do not follow normality as well as there is deviation in the middle and at the tails. However, there is no influential points. Overall, the model needs some changes.

4. Create a log-linear model m2 with response log(mass_g) and regressor SVL_mm. Use natural log (log in R) Show the code used to create the model and the model summary in your report.

```
m2 <- lm(log(mass_g) ~ SVL_mm, data=my_lizards)
summary(m2)

##
## Call:
## lm(formula = log(mass_g) ~ SVL_mm, data = my_lizards)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.54951 -0.15510 -0.01751  0.19252  0.37973
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.866893   0.134590  -6.441 1.42e-08 ***
## SVL_mm       0.041067   0.001615  25.429  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2285 on 68 degrees of freedom
## Multiple R-squared:  0.9048, Adjusted R-squared:  0.9035
## F-statistic: 646.7 on 1 and 68 DF,  p-value: < 2.2e-16

par(mfrow = c(2,2))
plot(m2)
```



m2 has better residuals than m1. They are more linear without a pattern. The Q-Q plot shows less deviation at the middle, though there are still deviations at the tails. The standardised residuals is not as curvy as that of m1. This shows a better scatter and a better equal variance. It also resembles a normal distribution rather than skewed. There is no influential points.

5. Use a suitable approximation to interpret the coefficient of the SVL_mm regressor in model m2 in the context of the relationship between mass and SVL (20-30 words).

```
(1 + 0.041067)
```

```
## [1] 1.041067
```

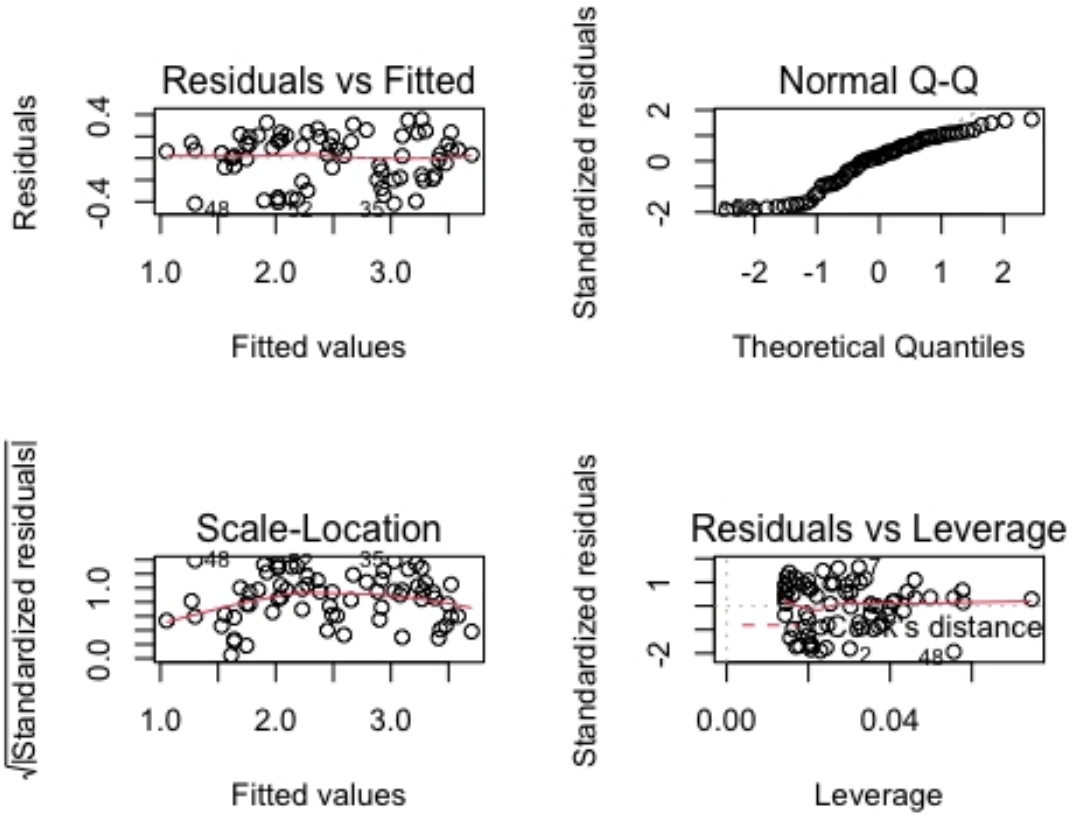
With a logged response and unlogged regressor, using natural logs with a smallish magnitude of 0.041067, an increase of 1 in SVL_mm multiplies log(mass_g) by approxiamately 1.041067 mm.

6. Create a log-log model m3 for response log(mass_g) and regressor log(SVL_mm), again using natural log. Show the code used to create the model and the model summary in your report.

```
m3 <- lm(log(mass_g) ~ log(SVL_mm), data=my_lizards)
summary(m3)
```

```
##
## Call:
## lm(formula = log(mass_g) ~ log(SVL_mm), data = my_lizards)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.41754 -0.17267  0.03564  0.18423  0.35848
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -12.1288     0.5575  -21.75   <2e-16 ***
## log(SVL_mm)   3.3362     0.1271   26.24   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.222 on 68 degrees of freedom
## Multiple R-squared:  0.9101, Adjusted R-squared:  0.9088
## F-statistic: 688.5 on 1 and 68 DF,  p-value: < 2.2e-16
```

```
par(mfrow = c(2,2))
plot(m3)
```



m3 has a lot better residuals than m2. m3's residuals have a lot better linearity with more scatter and no pattern. The standardised residuals are the same so still quite good. The Q-Q plot shows a reasonable normal distribution and this could be because the sample size is only 70. There is no influential points.

7. Use a suitable approximation to interpret the coefficient of the log(SVL_mm) regressor in model m3 in the context of the relationship between mass and SVL (20-30 words).

```
(1 + 3.3362 * 0.01)
```

```
## [1] 1.033362
```

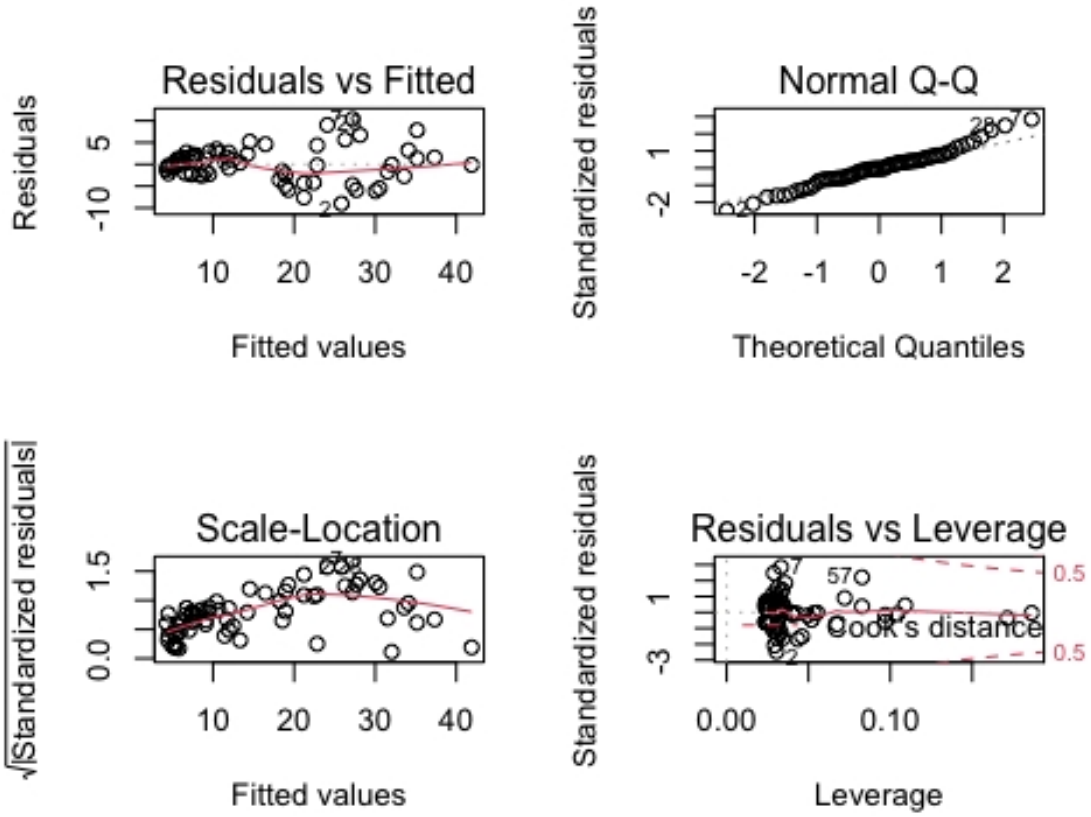
With a logged response and logged regressor, using natural logs with a smallish magnitude of 3.3362, a 1% increase in log(SVL_MM) multiplies log(mass_g) by approximately 1.033362.

8. Create a quadratic model m4 for response mass_g with predictor SVL_mm (ie, model mass_g as a quadratic function of SVL_mm). Show the code used to create the model and the model summary in your report.

```
m4 <- lm(mass_g ~ SVL_mm + I(SVL_mm^2), data=my_lizards)
summary(m4)
```

```
##
## Call:
## lm(formula = mass_g ~ SVL_mm + I(SVL_mm^2), data = my_lizards)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.0170 -2.1653 -0.1224  1.6301 10.3705
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 28.436157  12.105837   2.349  0.02178 *
## SVL_mm      -0.944539   0.298983  -3.159  0.00237 **
## I(SVL_mm^2)  0.009232   0.001785   5.172 2.28e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.704 on 67 degrees of freedom
## Multiple R-squared:  0.8905, Adjusted R-squared:  0.8872
## F-statistic: 272.3 on 2 and 67 DF,  p-value: < 2.2e-16
```

```
par(mfrow = c(2,2))
plot(m4)
```



m4's residuals are not any better than m3. The residuals do not follow linearity like m3 as it is now curvy. There is also a concentration of residuals at lower fitted values.The standardised residuals is now more curvy than m3 and approximates a skewed distribution implying not following variance is not constant. The Q-Q plot still has deviations at the tails not following normality. There is no influential points however.

9. Interpret the intercept and the coefficient of the SVL_mm regressor in model m4 in relation to the quadratic curve relating mass and SVL given by the model. How meaningful are these coefficients in context? (About 50-60 words in total.)

The intercept means that when snout-vent length is 0 then the lizard's mass is 28.44 g. Not meaningful since a lizard can't be that heavy without half of its body.

The b1 means that when snout-vent length is zero then the slope of the tangent to the line curve is -0.944539.

The b2 means that the curvature increases by 0.009232 g constantly.

10. **Add a variable SVL_mm_c for the centered SVL_mm to the data. Show the code to create this variable in your report. Create a quadratic model m5 for response mass_g using predictor SVL_mm_c (ie, model mass_g as a quadratic function of the centered SVL variable SVL_mm_c). Show the code used to create the model and the model summary in your report.**

```
my_lizards <- my_lizards %>%
  mutate(SVL_mm_c = SVL_mm - mean(SVL_mm))

m5 <- lm(mass_g ~ SVL_mm_c + I(SVL_mm_c^2), data=my_lizards)
summary(m5)

##
## Call:
## lm(formula = mass_g ~ SVL_mm_c + I(SVL_mm_c^2), data = my_lizards)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.0170 -2.1653 -0.1224  1.6301 10.3705
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   12.834480   0.675547  18.999  < 2e-16 ***
## SVL_mm_c       0.562178   0.026974  20.841  < 2e-16 ***
## I(SVL_mm_c^2)  0.009232   0.001785   5.172 2.28e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.704 on 67 degrees of freedom
## Multiple R-squared:  0.8905, Adjusted R-squared:  0.8872
## F-statistic: 272.3 on 2 and 67 DF,  p-value: < 2.2e-16
```

The intercept means that when snout-vent length = mean snout-vent length then the lizard's mass is 12.83 g. Meaningful since this would be the average weight of a lizard based on this sample anyway.

The b1 means that when snout-vent length = mean snout-vent length then the slope of the tangent to the line curve is 0.562178. Meaningful since one unit increase in SVL_mm_c increases mass by 0.562178 g.

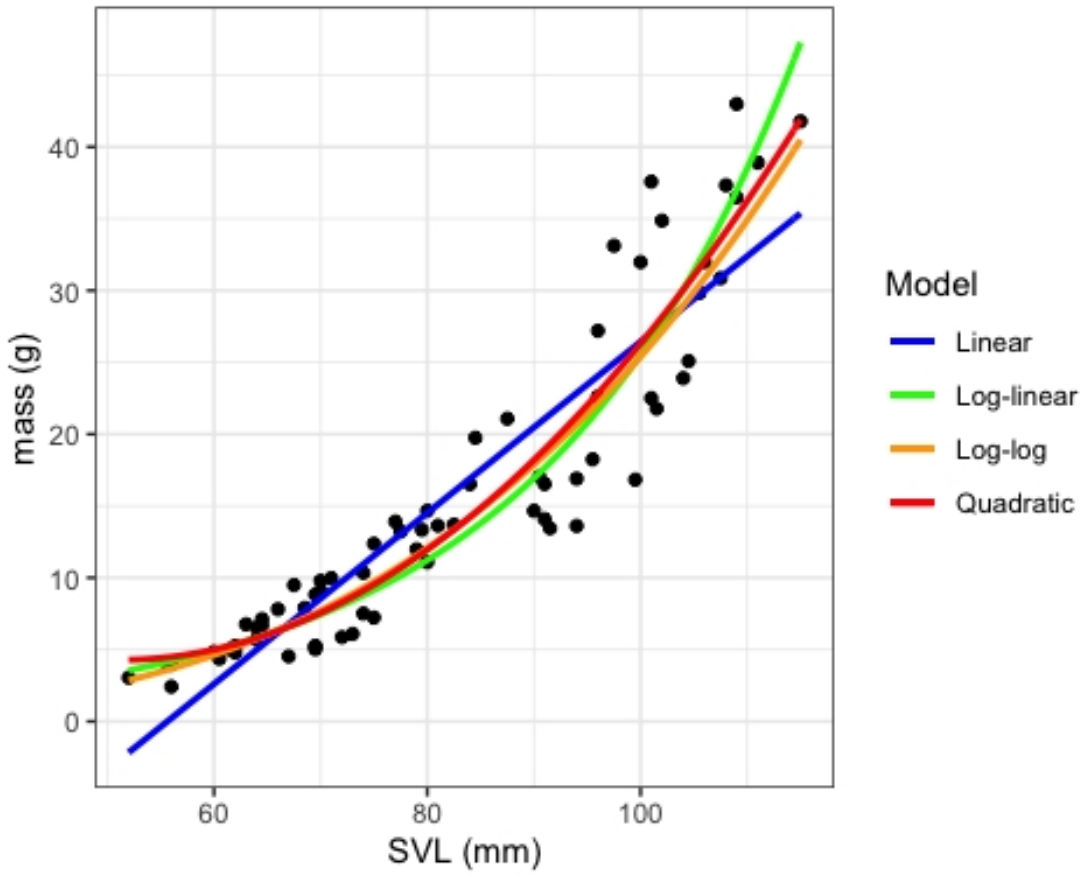
The b2 means that the curvature increases by 0.009232 g constantly. This doesn't change and unaffected by changes in SVL_mm_c.

11. Which are the only two models out of the models m1, m2, m3, m4, m5 created above that could be compared using a nested model F-test with anova? Briefly explain your answer in 50-70 words.

m5 and m4 are the only two models that can be compared using a nested model anova because their variables are not logged. m1, m2 and m3 all has log in response or regressors which cannot be compared as they are different. For instance, m3 and m2 cannot be compared as m2 has SVL_mm and m3 has log(SVl_mm) which are two different regressors.

12. Which model do you prefer? Very briefly explain your answer (about 30 words).

```
my_lizards %>% ggplot(aes(x = SVL_mm, y = mass_g)) +
 geom_point() +
 geom_function(
 fun = function(x) coef(m1)[[1]] + coef(m1)[[2]] * x,
 aes(color = "blue"), size = 1
 ) +
 # alternative for linear is geom_smooth(method = "lm", se = FALSE, aes(color
=
#"blue") ) +
 geom_function(
 fun = function(x) exp(coef(m2)[[1]]) * exp(x * coef(m2)[[2]]),
 aes(color = "green"), size = 1
 ) +
 geom_function(
 fun = function(x) exp(coef(m3)[[1]]) * x^coef(m3)[[2]],
 aes(color = "orange"), size = 1
 ) +
 geom_function(
 fun = function(x) coef(m4)[[1]] + coef(m4)[[2]] * x + coef(m4)[[3]] * x^2,
 aes(color = "red"), size = 1
 ) +
 labs(x = "SVL (mm)", y = "mass (g)") +
 # hack the legend!
 scale_color_identity(
 name = "Model",
 breaks = c("blue", "green", "orange", "red"),
 labels = c("Linear", "Log-linear", "Log-log", "Quadratic"),
 guide = "legend"
 ) +
 theme_bw()
```



I prefer the m3(log-log) model as it captures the values better than any models. For example, m3's residuals follow linearity whereas the linearity other models are curvy and produces a pattern.