

# Salamanders

Gao

2023-10-12

Import libraries

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.2      v readr      2.1.4
## v forcats    1.0.0      v stringr    1.5.0
## v ggplot2    3.4.3      v tibble     3.2.1
## v lubridate  1.9.2      v tidyr      1.3.0
## v purrr      1.0.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(tidymodels)
```

```
## -- Attaching packages ----- tidymodels 1.1.1 --
## v broom       1.0.5      v rsample     1.2.0
## v dials       1.2.0      v tune        1.1.2
## v infer       1.0.5      v workflows   1.1.3
## v modeldata   1.2.0      v workflowsets 1.0.1
## v parsnip     1.1.1      v yardstick   1.2.0
## v recipes     1.0.8
## -- Conflicts ----- tidymodels_conflicts() --
## x scales::discard() masks purrr::discard()
## x dplyr::filter()   masks stats::filter()
## x recipes::fixed()  masks stringr::fixed()
## x dplyr::lag()       masks stats::lag()
## x yardstick::spec() masks readr::spec()
## x recipes::step()   masks stats::step()
## * Use suppressPackageStartupMessages() to eliminate package startup messages
```

```
library(ggforce)
```

```
library(yardstick)
```

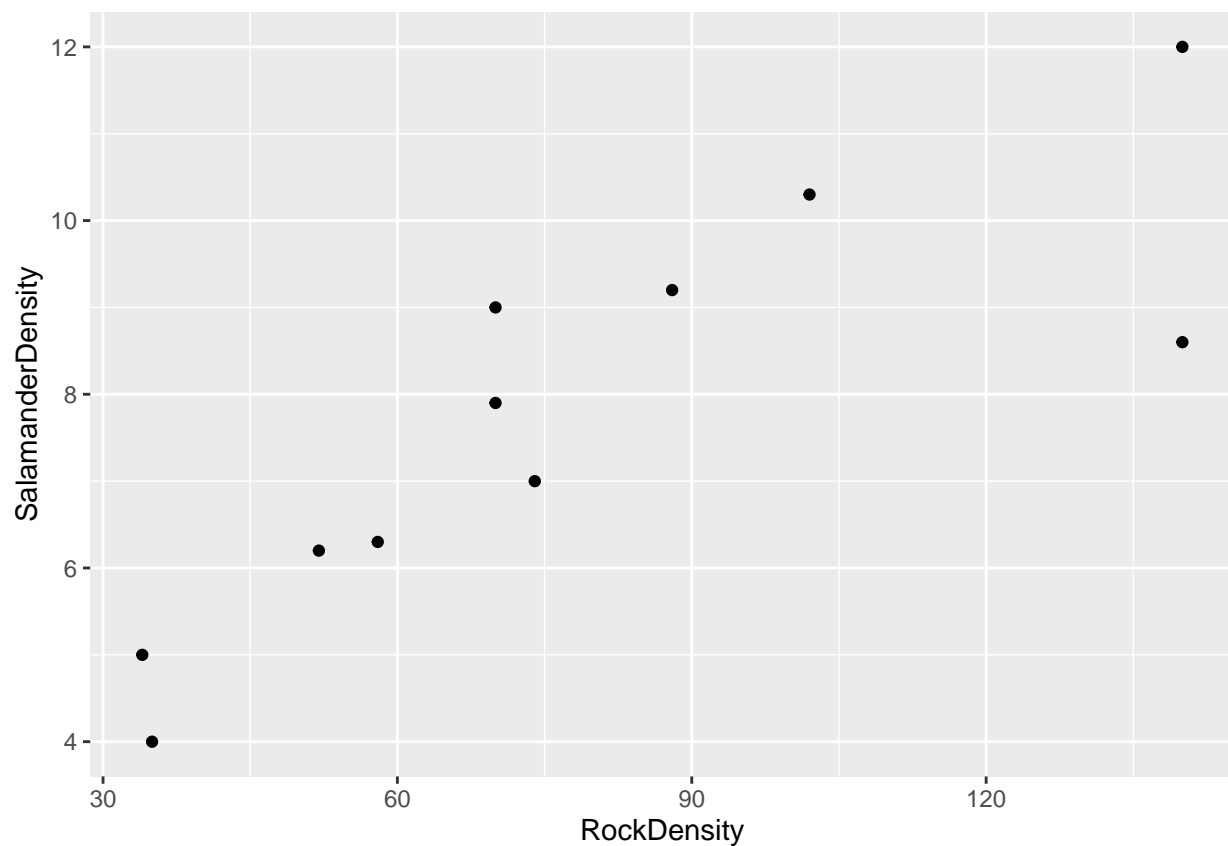
Import Salamanders data

```
salamanders <- read_csv("Salamanders.csv") %>% as_tibble()
```

```
## Rows: 11 Columns: 2
## -- Column specification -----
## Delimiter: ","
## dbl (2): RockDensity, SalamanderDensity
##
## 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.
```

Create a plot

```
ggplot(data = salamanders) + geom_point(aes(x = RockDensity, y = SalamanderDensity))
```



Create a linear regression model

```
model <- lm(SalamanderDensity ~ RockDensity, data = salamanders)
model
```

```
##
## Call:
## lm(formula = SalamanderDensity ~ RockDensity, data = salamanders)
##
## Coefficients:
## (Intercept)  RockDensity
##      3.458      0.055
```

```
summary(model)
```

```
##
## Call:
## lm(formula = SalamanderDensity ~ RockDensity, data = salamanders)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.5576 -0.4378 -0.1178  0.8724  1.6923
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.45792    0.96599   3.580 0.005934 **
## RockDensity  0.05500    0.01125   4.888 0.000862 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.301 on 9 degrees of freedom
## Multiple R-squared:  0.7264, Adjusted R-squared:  0.696
## F-statistic: 23.89 on 1 and 9 DF, p-value: 0.0008621
```

```
salamanders$sqrtSalamanderDensity <- salamanders$SalamanderDensity^0.5
model2 <- lm(sqrtSalamanderDensity ~ RockDensity, data = salamanders)
model2
```

```
##
## Call:
## lm(formula = sqrtSalamanderDensity ~ RockDensity, data = salamanders)
##
## Coefficients:
## (Intercept)  RockDensity
##      1.97228      0.01001
```

```
summary(model2)
```

```
##
## Call:
## lm(formula = sqrtSalamanderDensity ~ RockDensity, data = salamanders)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.44077 -0.07178 -0.00270  0.15904  0.32719
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.972285    0.179183  11.007 1.6e-06 ***
## RockDensity  0.010008    0.002087   4.795 0.000981 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2413 on 9 degrees of freedom
```

```
## Multiple R-squared:  0.7187, Adjusted R-squared:  0.6874
## F-statistic: 22.99 on 1 and 9 DF,  p-value: 0.0009808
```

```
salamanders$SalamanderDensity2 <- salamanders$SalamanderDensity^2
```

```
model3 <- lm(SalamanderDensity2 ~ RockDensity, data = salamanders)
model3
```

```
##
## Call:
## lm(formula = SalamanderDensity2 ~ RockDensity, data = salamanders)
##
## Coefficients:
## (Intercept)  RockDensity
##      -2.9275      0.8719
```

```
summary(model3)
```

```
##
## Call:
## lm(formula = SalamanderDensity2 ~ RockDensity, data = salamanders)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -45.176  -9.770  -1.716   15.464   24.864
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.9275     15.8443  -0.185  0.85751
## RockDensity   0.8719      0.1846   4.724  0.00108 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 21.34 on 9 degrees of freedom
## Multiple R-squared:  0.7126, Adjusted R-squared:  0.6807
## F-statistic: 22.32 on 1 and 9 DF,  p-value: 0.001083
```

```
salamanders$reSalamanderDensity <- 1/salamanders$SalamanderDensity
```

```
model4 <- lm(reSalamanderDensity ~ RockDensity, data = salamanders)
model4
```

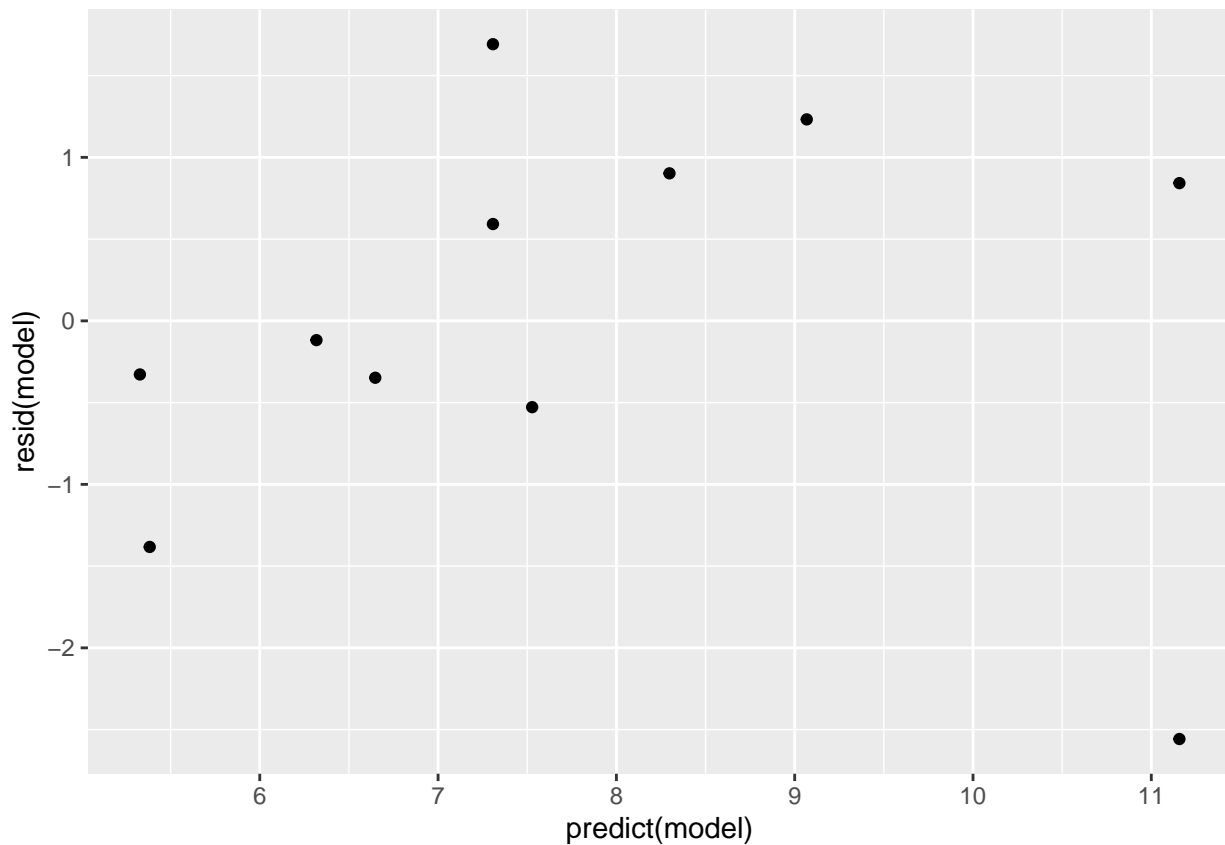
```
##
## Call:
## lm(formula = reSalamanderDensity ~ RockDensity, data = salamanders)
##
## Coefficients:
## (Intercept)  RockDensity
##      0.22542     -0.00107
```

```
summary(model4)
```

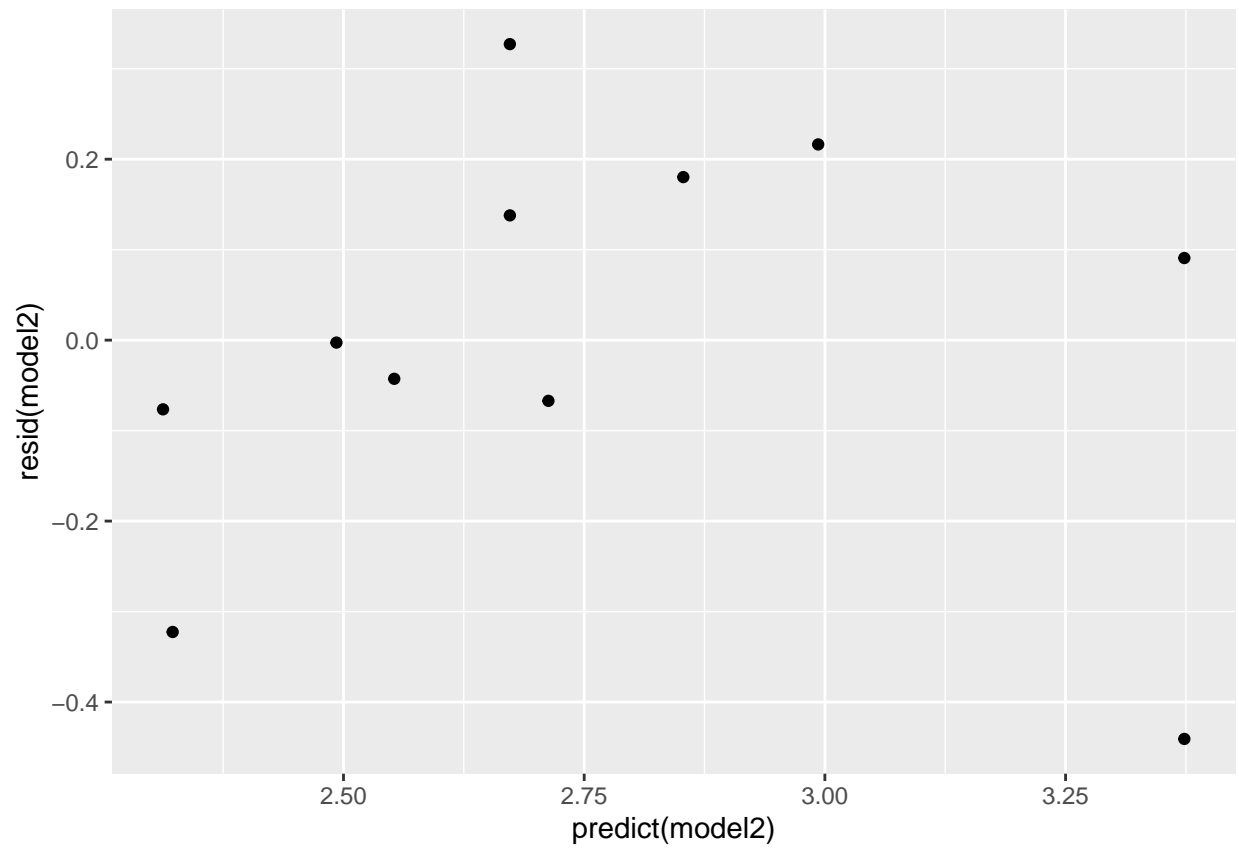
```
##
## Call:
## lm(formula = reSalamanderDensity ~ RockDensity, data = salamanders)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.039389 -0.020853 -0.004612  0.009361  0.062043
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.2254153  0.0232048   9.714 4.55e-06 ***
## RockDensity -0.0010702  0.0002703  -3.959  0.00331 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03125 on 9 degrees of freedom
## Multiple R-squared:  0.6353, Adjusted R-squared:  0.5948
## F-statistic: 15.68 on 1 and 9 DF,  p-value: 0.003307
```

Model 1 is the most accurate, as it has the lowest  $r^2$  value

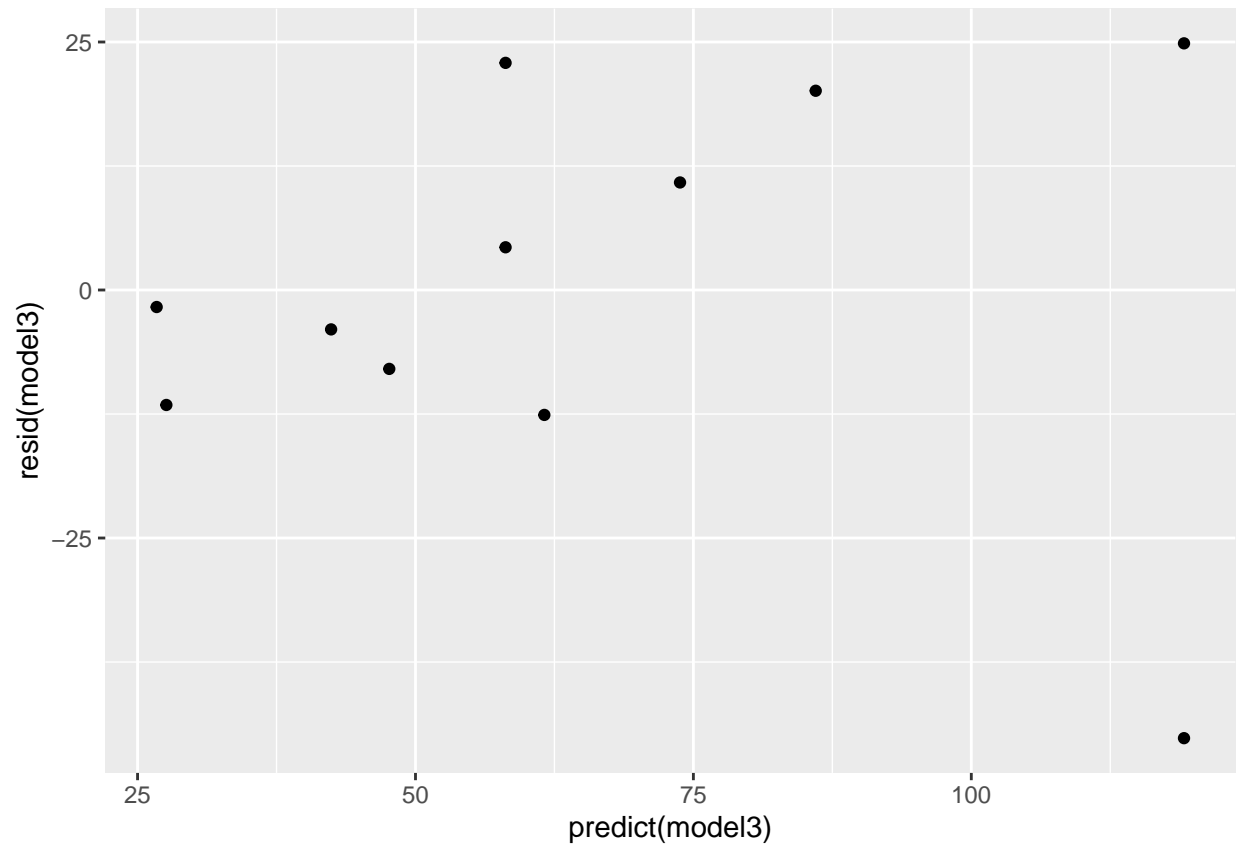
```
ggplot(salamanders) + geom_point(aes(x=predict(model), y=resid(model)))
```



```
ggplot(salamanders) + geom_point(aes(x=predict(model2), y=resid(model2)))
```



```
ggplot(salamanders) + geom_point(aes(x=predict(model3), y=resid(model3)))
```



```
ggplot(salamanders) + geom_point(aes(x=predict(model4), y=resid(model4)))
```

