10/27 Model Diagnostics (Part 2)

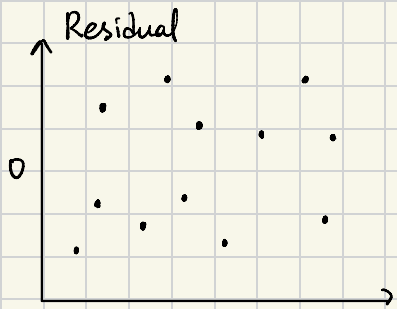
Residual plots when the model is correct

i.e. $E(y_i) = \beta^T x_i$, $\varepsilon_i \overset{iid}{\sim} N(0, \sigma^2)$

1) Linearity

2) Constant variance

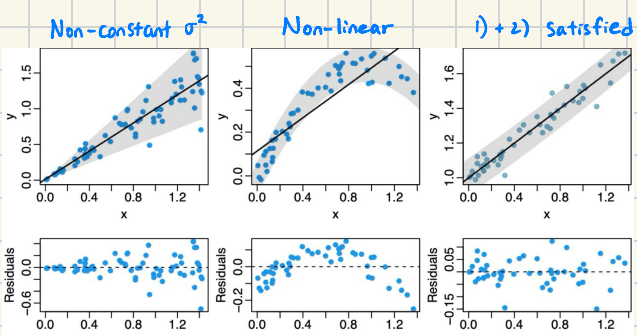
Claim: If 1) + 2) hold, then the residual plot typically looks like:



(Random scatter)

Intuition: If our model captures the true relationship between y and x, then what remains, i.e. $y - \hat{y}$, should not be predictable from x (or any function of x)

e.g. SLR ($p=1$) ⇝ Plot y vs x, and $e = y - \hat{y}$ vs x



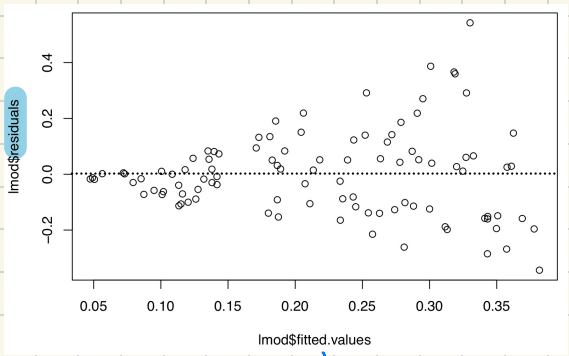
"Null" plot

For MLR ($p > 1$), it can be trickier ...

e.g. Caution data (artificial data set, Cook & Weisberg 1999b)

```
> caution.df <- read.csv("caution.csv")
> head(caution.df)
          x1        x2        y
1  0.8880370  0.453707 0.1560420
2 -0.0975793  0.157144 0.0343178
3 -0.2105760  0.968198 0.0304575
4 -0.6783160 -0.675659 0.8727100
5  0.7905930 -0.496129 0.3411820
6  0.5247750 -0.648466 0.6188230
```

```
> lmod <- lm(y~x1+x2,caution.df)
> plot(lmod$fitted.values,lmod$residuals)
```



$$g(X) = \hat{\beta}_0 + \hat{\beta}_1 X_1 + \hat{\beta}_2 X_2 = \hat{y}$$

Plot shape suggests non-constant variance, but data generated using $E[y|X] = \frac{|x_1|}{2 + (1.5 + x_2)^2}$ (non-linear mean function)

Takeaway: A "non-null" residual plot tells us some assumption fails but not necessarily which one.

Detecting non-linearity in MLR

Q: Is Y linear in X_i, after controlling for the other X_j's?

"Partial residual plot"

$$e + \hat{\beta}_i X_i = y - \sum_{j \neq i} \hat{\beta}_j X_j$$

a.k.a "Residual plus component"

Plot partial residuals versus X_i to check linearity

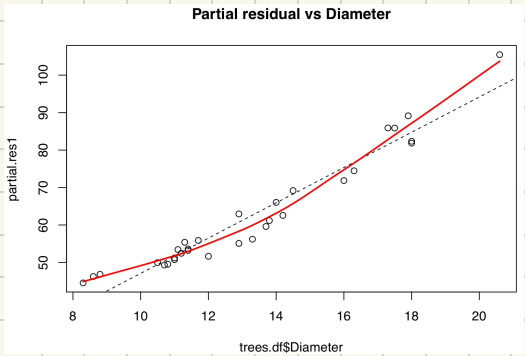
e.g. Cherry tree data (31 felled trees, measure volume, diam, height)

Consider regressing Volume ~ Diam + Height

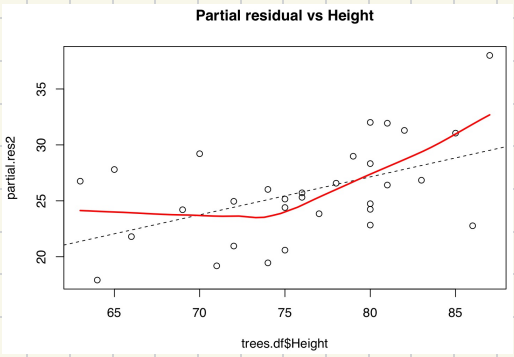
```
> lmod <- lm(Volume~Diameter+Height,trees.df)
> beta <- coefficients(lmod); beta
(Intercept)    Diameter      Height
-57.9876589   4.7081605   0.3392512
> partial.res1 <- lmod$residuals+beta[2]*trees.df$Diameter
> partial.res2 <- lmod$residuals+beta[3]*trees.df$Height
```

Loess: "Locally estimated scatterplot smoothing"

```
plot(trees.df$Diameter,partial.res1,main="Partial residual vs Diameter")
abline(lm(partial.res1~trees.df$Diameter),lty=2)
points(loessLine(trees.df$Diameter,partial.res1,col="red"))
```



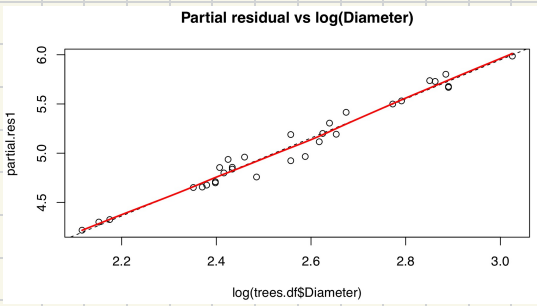

```
plot(trees.df$Height,partial.res2,main="Partial residual vs Height")
abline(lm(partial.res2~trees.df$Height),lty=2)
points(loessLine(trees.df$Height,partial.res2,col="red"))
```



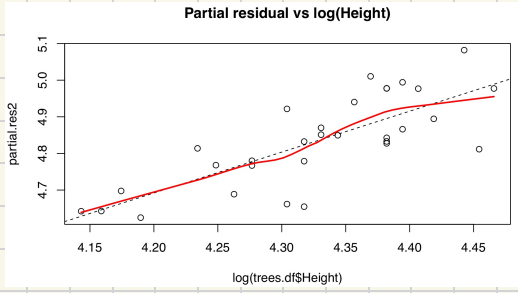
After transformation... log(Volume) ~ log(Diam) + log(Height)

```
> lmod2 <- lm(log(Volume)~log(Diameter)+log(Height),trees.df)
> beta <- coefficients(lmod2); beta
  (Intercept) log(Diameter)   log(Height)
    -6.631617      1.982650      1.117123
> partial.res1 <- lmod2$residuals+beta[2]*log(trees.df$Diameter)
> partial.res2 <- lmod2$residuals+beta[3]*log(trees.df$Height)
```

```
plot(log(trees.df$Diameter),partial.res1,main="Partial residual vs log(Diameter)")
abline(lm(partial.res1~log(trees.df$Diameter)),lty=2)
points(loessLine(log(trees.df$Diameter),partial.res1,col="red"))
```



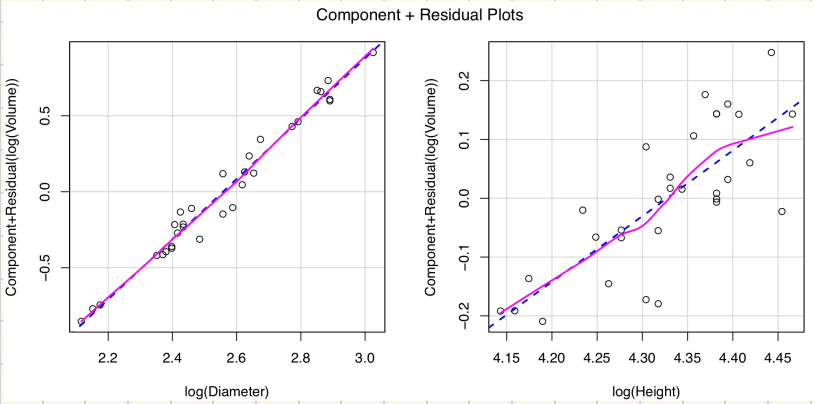

```
plot(log(trees.df$Height),partial.res2,main="Partial residual vs log(Height)")
abline(lm(partial.res2~log(trees.df$Height)),lty=2)
points(loessLine(log(trees.df$Height),partial.res2,col="red"))
```



Shortcut in R: `> crPlots(lmod2)`

Cr Plot

↳ Component + residual



Q: When do we consider taking logs?

Rule of thumb: If a variable is positive and its values range over an order of magnitude, then replacing by its log may help.

e.g. Brain data (mammals, n = 96)

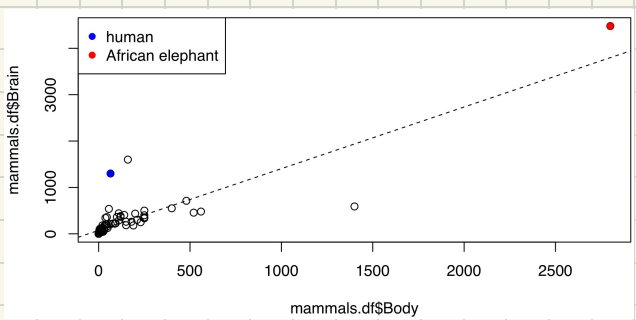
```
> head(mammals.df)
                Species  Brain    Body
1              Aardvark    9.6    2.20
2              Acouchis    9.9    0.78
3 African elephant 4480.0 2800.00
4              Agoutis   20.3    2.80
5            Axis deer  219.0   89.00
6                Badger   53.0    6.00
```

Brain = weight in grams

Body = weight in kg

```
> hum <- which(mammals.df$Species=="Human being")
> af.el <- which(mammals.df$Species=="African elephant")
> plot(mammals.df$Body,mammals.df$Brain)
> abline(lm(Brain~Body,mammals.df),lty=2)
> points(mammals.df$Body[hum],mammals.df$Brain[hum],pch=16,col="blue")
> points(mammals.df$Body[af.el],mammals.df$Brain[af.el],pch=16,col="red")
> legend("topleft",c("human","African elephant"),col=c("blue","red"),pch=16)
```

Scatter plot:

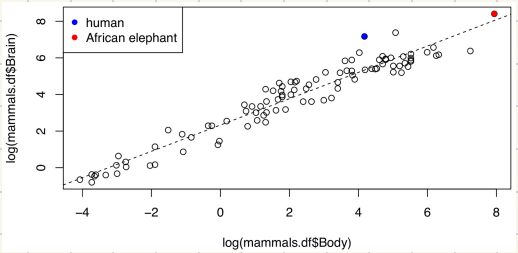


$R^2 = 0.75$

After logging both variables:

$R^2 = 0.93$

$\hat{\beta}_1 = 1.68$, $se(\hat{\beta}_1) \approx 0.04$



Interpretation: A 1% increase in body weight is associated with a $\approx 1.68\%$ increase in brain weight (Allometric interpretation)

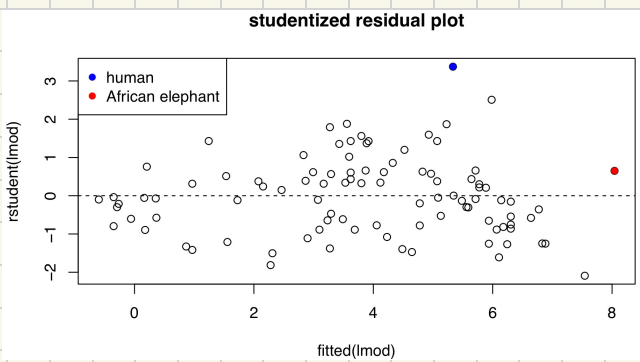
Q: Why? Revert $\log(\text{Brain}) = \beta_0 + \beta_1 \log(\text{Body}) + \varepsilon$ back to original scale $\rightsquigarrow$ $\text{Brain} = \beta_0 (\text{Body})^{\beta_1} \exp(\varepsilon)$ → multiplicative noise

$$\approx \beta_0 (\text{Body})^{\beta_1} \quad \text{(on average)}$$

$$\Rightarrow \frac{d\,\text{brain}}{d\,\text{body}} \approx \beta_0 \beta_1 \,\text{body}^{\beta_1 - 1} = \beta_1 \frac{\text{brain}}{\text{body}} \Rightarrow \beta_1 = \frac{d\text{brain}/\text{brain}}{d\text{body}/\text{body}}$$

□

Plot studentized residuals:




```
> lmod <- lm(log(Brain)~log(Body),mammals.df)
> plot(fitted(lmod),rstudent(lmod),main="studentized residual plot")
> abline(h=0,lty=2)
> points(fitted(lmod)[hum],rstudent(lmod)[hum],pch=16,col="blue")
> points(fitted(lmod)[af.el],rstudent(lmod)[af.el],pch=16,col="red")
> legend("topleft",c("human","African elephant"),col=c("blue","red"),pch=16)
```

$\rightsquigarrow$ Test if humans are an outlier

if not, then $t_i \sim$ Student-t on $n-2-p$ df

$96 - 2 - 1 = 93$

```
> rstudent(lmod)[hum]
      53
3.372468
```

Assuming we'd have been interested in humans before seeing the data, p-value is $2(1-pt(3.37, 93)) = 0.0005$

Otherwise, Bonferroni corrected p-value is $96(0.0005) \approx 0.05$

Interpretation: Human brain weight is too large to be consistent with the model for the data

Takeaway: Transforming can be a remedy for non-constant variance (brain data) and non-linearity (tree data) assumptions, both of which can lead to invalid CI's or hyp. tests

Choosing a transformation for y

Idea: Estimate the transformation needed for y to be consistent with the assumptions of normal errors + constant variance

Consider the model: $y^\lambda = X\beta + \varepsilon \qquad Var(\varepsilon) = \sigma^2 I_n$ ★

$-2 \le \lambda \le 2$, $y^\lambda = (y_1^\lambda, \ldots, y_n^\lambda)^T$, $y_i^\lambda \equiv \log y_i$ if $\lambda = 0$

⇝ Compute $\hat{\beta}(\lambda) = (X^TX)^{-1}X^T y^\lambda$ ⇝ $RSS_\lambda = \|(I-H)y^\lambda\|^2$

Box-Cox method: Choose λ to maximize log-likelihood $L(\lambda)$ defined: $L(\lambda) = n\log(\lambda) - \frac{n}{2}\log(RSS_\lambda) + (\lambda-1)\sum_{i=1}^{n}\log y_i$

(derived from normal pdf in model ★)

$$\hat{\lambda} = \underset{-2 \le \lambda \le 2}{\operatorname{argmax}} L(\lambda)$$ (usually round to nearby fraction)

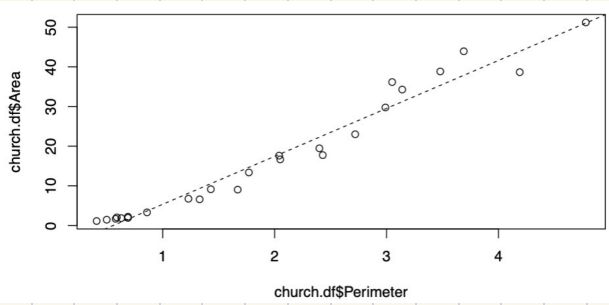
Example (Church data) (Clapham 1934) (Gould 1973)

```
> church.df <- read.csv("churches.csv")
> head(church.df)
      Church Perimeter  Area
1 St. Albans      3.48 38.83
2     Durham      3.69 43.92
3      Blyth      1.43  9.14
4     Binham      2.05 16.66
5 Gloucester      3.05 36.16
6    Norwich      4.19 38.66
```

For $n = 25$ churches, measure

Perimeter in hundreds of meters

Area in hundreds of sq. meters



Slight non-linearity +

non-constant variance

($\sigma^2(x)$ increases as perimeter increases)

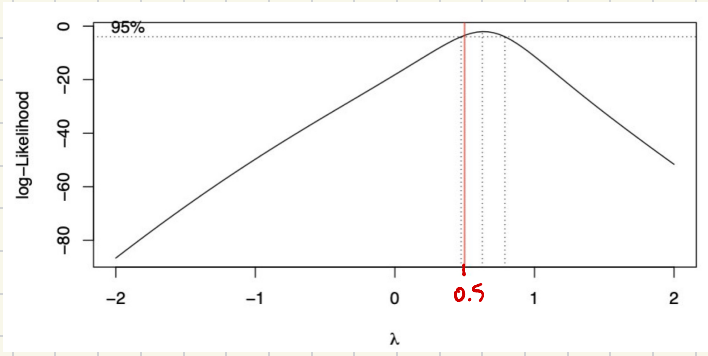
⇝ Plot $L(\lambda)$ and choose λ which makes $L(\lambda)$ large

In R:

```
lmod <- lm(Area~Perimeter,church.df)
library(MASS)
boxcox(lmod)
abline(v=1/2,col="red")
```

95% CI for the "best" λ

$L(\lambda)$



(details for CI beyond the scope of Stat 224)

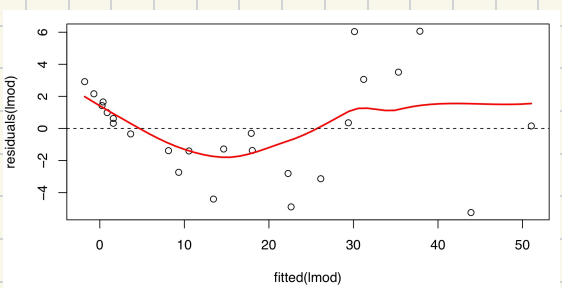
Usually select λ to keep the response interpretable, e.g. in $\{-1, 0$ (log transform), $1/2, 1/3, 1, 2\}$ which is within the 95% region on the x-axis.

Here, choose $\lambda = 1/2$ (area$^{1/2}$ has same units as perimeter and $1/2$ contained in 95% CI)

Q: Does this resolve the issues (non-linearity + increasing variance)?

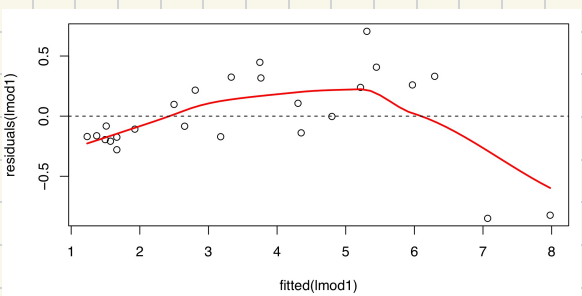
Before

```
lmod <- lm(Area~Perimeter,church.df)
plot(fitted(lmod),residuals(lmod))
points(loessLine(fitted(lmod),residuals(lmod),col="red"))
abline(h=0,lty=2)
```



After

```
lmod1 <- lm(Area^(1/2)~Perimeter,church.df)
plot(fitted(lmod1),residuals(lmod1))
points(loessLine(fitted(lmod1),residuals(lmod1),col="red"))
abline(h=0,lty=2)
```



(Still not great ...)

⇝ Consider transforming X = perimeter

Idea: Let $g_\alpha(X) = \begin{cases} X^\alpha & \text{if } \alpha \neq 0 \\ \log X & \text{if } \alpha = 0 \end{cases}$ and H_0: no transform needed i.e. $\alpha = 1$

$g_1(X) = X$ Goal: Test H_0

$\Rightarrow$ If $\alpha \approx 1$, $g_\alpha(x) \approx x + \frac{\partial}{\partial \alpha} g_\alpha(x)\Big|_{\alpha=1} (\alpha - 1)$

(linear Taylor approx. in α)

Calculus $\Rightarrow$ $\frac{\partial}{\partial \alpha} g_\alpha(x)\Big|_{\alpha=1} = \frac{\partial}{\partial \alpha}[x^\alpha]\Big|_{\alpha=1} = x \log x$

(Proof: $\frac{\partial}{\partial \alpha}[x^\alpha] = \frac{\partial}{\partial \alpha} \exp(\alpha \log x) = (\log x)\, x^\alpha$)

Candidate model: $y = \beta_0 + \beta_1 g_\alpha(x) + \varepsilon$

$\approx \beta_0 + \beta_1 (x + (\alpha - 1) x \log x) + \varepsilon$

$= \beta_0 + \beta_1 x + \underbrace{\beta_1 (\alpha - 1)}_{\text{call this } \eta} x \log x + \varepsilon$

$\rightsquigarrow$ Procedure:

1) Fit MLR with new predictor $x \log x$

2) Test whether it's coefficient η equals 0

(t-test on $n - p - 1 = n - 3$ df)

3) If we reject H_0, choose $\hat{\alpha} = 1 + \hat{\eta}/\hat{\beta}_1$

(or round to nearest simple fraction, e.g. $1/3$, $1/2$, ..)

e.g.

```
lmod2 <- lm(sqrt(Area)~Perimeter+I(Perimeter*log(Perimeter)),church.df)
summary(lmod2)
```

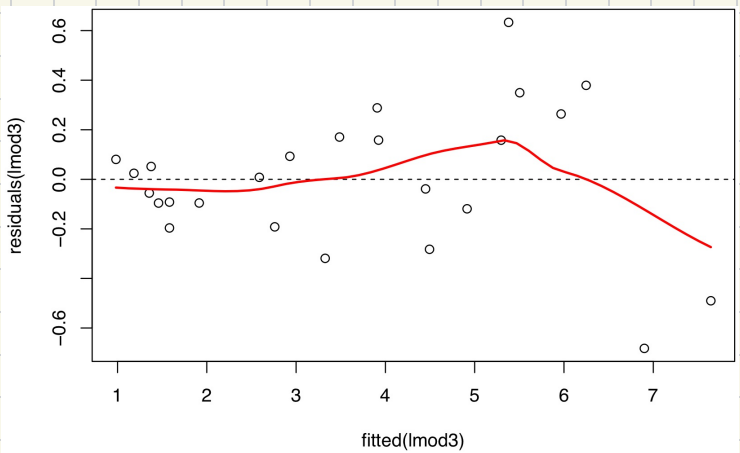
```
Coefficients:
                               Estimate Std. Error t value Pr(>|t|)
(Intercept)                     -0.5036     0.2679  -1.880 0.073409 .
Perimeter                        2.6966     0.2625  10.273 7.38e-10 ***
I(Perimeter * log(Perimeter))   -0.6727     0.1510  -4.454 0.000199 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2719 on 22 degrees of freedom
Multiple R-squared:  0.983,     Adjusted R-squared:  0.9814
F-statistic: 634.2 on 2 and 22 DF,  p-value: < 2.2e-16
```

⇝ Reject H_0

$$\hat{a} = 1 + \frac{(-0.67)}{2.7} \approx 3/4$$

Refit ⇝

```
lmod3 <- lm(sqrt(Area)~I(Perimeter^(3/4)),church.df)
plot(fitted(lmod3),residuals(lmod3))
abline(h=0,lty=2)
points(loessLine(fitted(lmod3),residuals(lmod3),col="red"))
```



Slightly less non-linear than before, but still non-const. variance

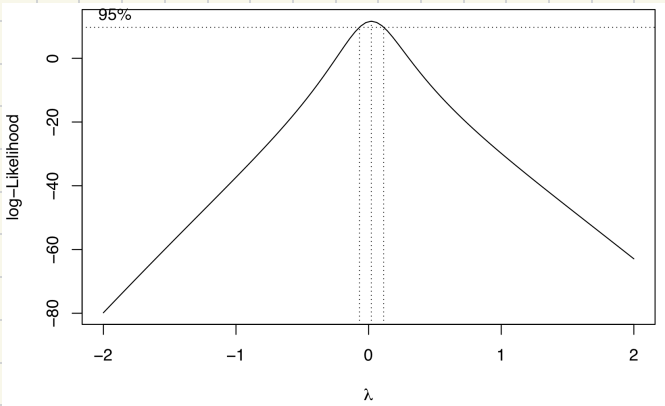
Notice that:

```
> max(church.df$Perimeter)/min(church.df$Perimeter)
[1] 11.65854
```

> 10 (one order of magnitude)

⇝ Consider $g(x) = \log(x)$ & check Box-Cox plot

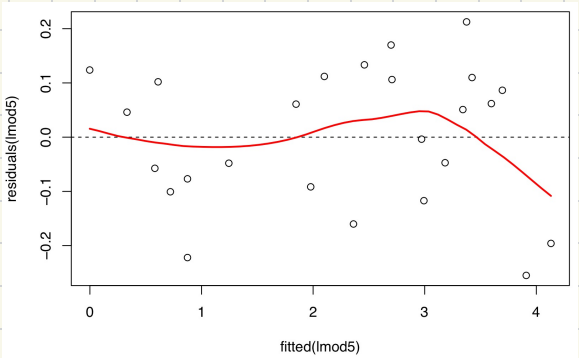
```
lmod4 <- lm(Area~log(Perimeter),church.df)
boxcox(lmod4)
```



⇝ Choose $\lambda = 0$, i.e. $\tilde{y} = \log(\text{Area})$

Refit & check residuals

```
lmod5 <- lm(log(Area)~log(Perimeter),church.df)
plot(fitted(lmod5),residuals(lmod5))
abline(h=0,lty=2)
points(loessLine(fitted(lmod5),residuals(lmod5),col="red"))
```



Satisfactory ...

Const. variance, slight non-linearity, but still interpretable (1% inc. in Perimeter ⇝ 1.68% inc. in Area)

```
Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)      1.49881    0.03068   48.85   <2e-16 ***
log(Perimeter)   1.68281    0.03587   46.92   <2e-16 ***
```

10/31 More on transformation

Q: Why transform?

- To obtain a linear relationship
- To restore constant variability assumption (Box-Cox method)

- Scientific theory suggests a transformation (example today)

Linearity means $E[y]$ is linear in β (not necessarily in X)

e.g. 1) $y = \beta_0 + \beta_1 X + \beta_2 X^2 + \varepsilon$

2) $y = \beta_0 + \beta_1 \log X + \varepsilon$

3) $y = \beta_0 + \beta_1 \sqrt{x} + \varepsilon$

} all linear, since $E(y) = h(x)^T \beta$

e.g. $h(x) = (1, X, X^2)$

Non-linear example: $y = \beta_0 + \exp\{\beta_1 X\} + \varepsilon$ (can't use MLR here)

Some nonlinear models can be made linear:

e.g. 1) $y = \alpha \exp\{\beta X\}$ (Exponential growth / decay)

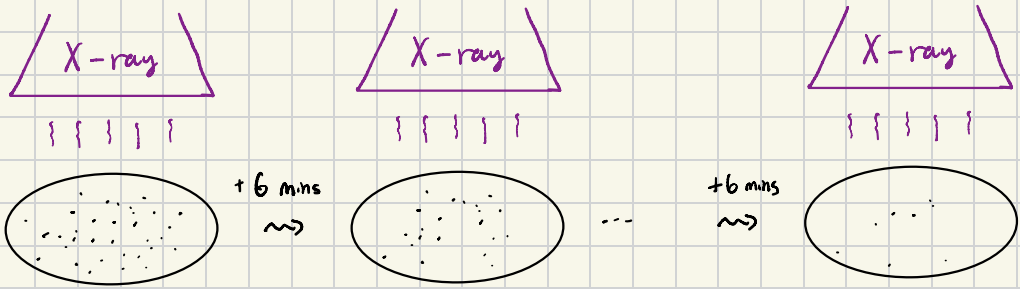
log-transform $\rightsquigarrow$ $\log y = \underbrace{\log \alpha}_{\beta_0} + \underbrace{\beta}_{\beta_1} X$

2) Suppose $y = \dfrac{X}{\alpha X - \beta}$

inverse-power $\rightsquigarrow$ $\underbrace{1/y}_{\tilde{y}} = \dfrac{\alpha X - \beta}{X} = \alpha + \beta \underbrace{(-\tfrac{1}{x})}_{\tilde{x}}$

$\rightsquigarrow$ MLR tools can be applied to some (but not all) non-linear models

e.g. Survival of bacteria (Lorentz & Henshaw, 1941)



Data: t = units of time (1 unit = 6 mins)

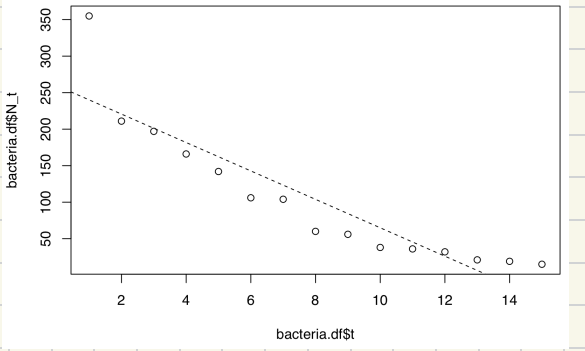
N_t = Number of surviving bacteria (in hundreds) after exposure to 200kV X-ray for t units of time (6t mins)

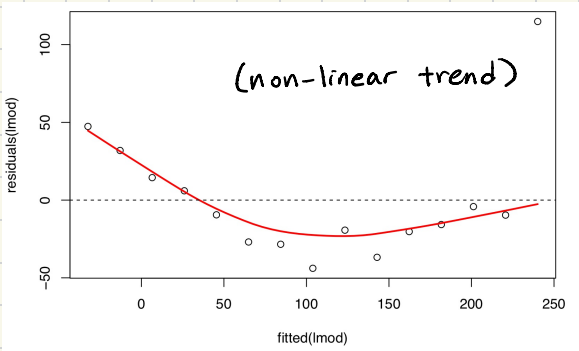
```
> bacteria.df <- read.table("bacteria.txt",header=TRUE)
> head(bacteria.df)
  t N_t
1 1 355
2 2 211
3 3 197
4 4 166
5 5 142
6 6 106
```

$(n = 15)$

```
lmod <- lm(N_t~t,bacteria.df)
plot(bacteria.df$t,bacteria.df$N_t)
abline(lmod,lty=2)
```

```
plot(fitted(lmod),residuals(lmod))
points(loessLine(fitted(lmod),residuals(lmod),col="red"))
abline(h=0,lty=2)
```





Scientific theory suggests the number of survivors decays exponentially

$\rightsquigarrow$ $N_t = N_0 e^{\beta_1 t}$ where N_0 = initial population

β_1 = decay rate

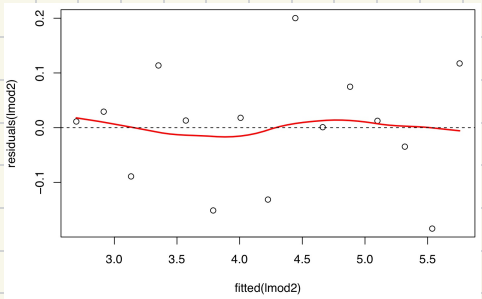
log both sides $\rightsquigarrow$ $\underbrace{\log N_t}_{y} = \underbrace{\log N_0}_{\beta_0} + \beta_1 t$

linear in (β_0, β_1) $\Rightarrow$ SLR can be applied

```
lmod2 <- lm(log(N_t)~t,bacteria.df)
summary(lmod2)
```

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.973160   0.059778   99.92  < 2e-16 ***
t           -0.218425   0.006575  -33.22 5.86e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.11 on 13 degrees of freedom
Multiple R-squared:  0.9884,    Adjusted R-squared:  0.9875
F-statistic:  1104 on 1 and 13 DF,  p-value: 5.86e-14
```

```
plot(fitted(lmod2),residuals(lmod2))
points(loessLine(fitted(lmod2),residuals(lmod2),col="red"))
abline(h=0,lty=2)
```



Interpretation: Every 6 minutes of exposure to X-rays causes the bacterial population to shrink by a factor of $\underbrace{e^{-0.218}}_{\text{estimated prop. of surviving bacteria}} \approx 80\%$

Q: Why? $E[\log N_t \mid t] = \beta_0 + \beta_1 t$

$\Rightarrow E[\log N_{t+1} \mid t+1] = \beta_0 + \beta_1(t+1) = \underbrace{\beta_0 + \beta_1 t}_{E[\log N_t \mid t]} + \beta_1$

$$\Rightarrow \quad N_{t+1} \approx \exp(\log N_t + \beta_1)$$

$$= N_t \exp(\beta_1) \quad \Rightarrow \quad \exp(\beta_1) = \frac{N_{t+1}}{N_t}$$

Q: 95% CI for the half life of bacteria exposed to X-ray?

$N_t = N_0 e^{\beta_1 t}$

(half life = amount of time being exposed to 200kV X-ray until only half the initial population remains alive)

$$e^{\beta_1 t^*} = 1/2 \quad \Rightarrow \quad \beta_1 t^* = \log(1/2) \quad \Rightarrow \quad t^* = \frac{\log(1/2)}{\beta_1}$$

Point estimate: $\hat{t}^* = \dfrac{\log(1/2)}{-0.218} \approx 3.18$ units of time

(19.1 minutes)

95% CI for β_1 $\rightsquigarrow$

```
> confint(lmod2,"t")
        2.5 %     97.5 %
t -0.2326291 -0.2042214
```

$\Rightarrow$ 95% CI for $t^* = \log(1/2)/\beta_1$ given by:

$$\left(\frac{\log(1/2)}{-0.233}, \frac{\log(1/2)}{-0.204}\right) \approx (2.98, 3.39) \times 6 \text{ minutes}$$

$$= (17.9, 20.4) \text{ minutes}$$

Q: 95% PI for N_0? (initial population)

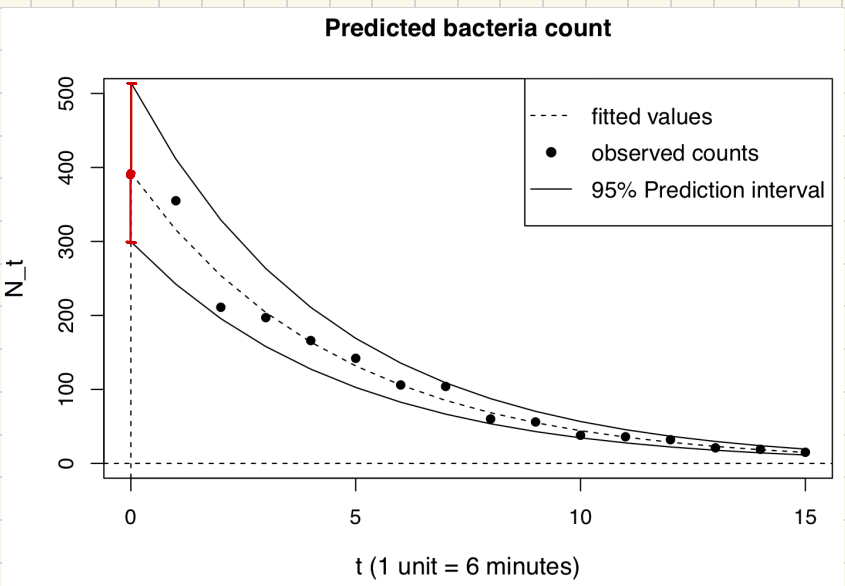
$$E[\log N_t \mid t=0] = \beta_0 + \beta_1(0) = \beta_0 \Rightarrow N_t \approx e^{\beta_0}$$

$$\rightsquigarrow \text{Point estimate: } e^{5.973} \approx 393$$

```
> pred.int <- predict(lmod2,data.frame(t=c(0:15)),interval="prediction",level=0.95)
> head(pred.int)
       fit      lwr      upr
1 5.973160 5.702666 6.243655
2 5.754735 5.489893 6.019577
3 5.536310 5.276467 5.796153
4 5.317885 5.062348 5.573421
5 5.099459 4.847500 5.351418
6 4.881034 4.631893 5.130175
```

t=0
t=1
2
3
⋮

95% PI: $(e^{5.703}, e^{6.244}) \approx (299.7, 514.7)$



Plotting code:

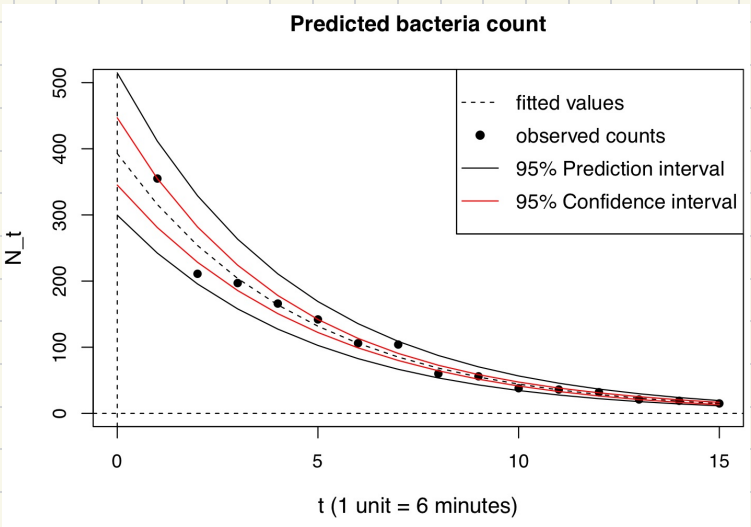
```
nt <- exp(pred.int[,1])
lower <- exp(pred.int[,2])
upper <- exp(pred.int[,3])
plot(c(0:15),nt,ylim=c(0,500),xlim=c(0,15),
     type="l",lty=2,xlab="t (1 unit = 6 minutes)",
     ylab="N_t",main="Predicted bacteria count",cex.lab=1.2)
abline(h=0,lty=2)
abline(v=0,lty=2)
points(c(0:15),lower,type="l",lty=1)
points(c(0:15),upper,type="l",lty=1)
points(bacteria.df$t,bacteria.df$N_t,pch=16)
legend("topright",c("fitted values","observed counts","95% Prediction interval"),
       lty=c(2,NA,1),pch=c(NA,16,NA),cex=1.1)
```

True or false: We are 95% confident that the expected number of bacteria at time 0 is ≈ between 300 and 515 units (1 unit = 100 bacteria)

False. Q: Why? This range covers the realized num. of bacterial units on a new plate (not yet exposed to X ray) with 95% probability

A 95% CI for the mean num. bacterial units at time = 0 is given by (345.2, 446.9), which is narrower than PI.

```
> conf.int <- predict(lmod2,data.frame(t=c(0:15)),interval="confidence",level=0.95)
> exp(conf.int[1,])
     fit      lwr      upr
392.7449 345.1633 446.8858
```



CI doesn't target coverage of observed counts (PI does)