# Diagnostics

## Motivation

- Estimation and inference of the regression model $Y = X\beta + \epsilon$ depend on several assumptions.
- These assumptions should be checked using regression diagnostics.
- The potential problems can be divided into three categories:
  - *Error.* We have assumed that $\epsilon \sim N(0, \sigma^2 I)$ or in words, that the errors are (1) independent, have (2) equal variance and are (3) normally distributed.
  - *Unusual observations.* Sometimes just a few observations do not fit the model. These few observations might change the choice and fit of the model.
  - *Model.* We have assumed that the structural part of the model, $E(Y \mid X) = X\beta$, is correct.

## 1. Checking Error Assumptions

- Goal: check error assumptions on (1) independence, (2) constant variance, and (3) normality.
- Errors are not observable → examine the sample residuals $\hat{\epsilon} = Y - X\hat{\beta}$.

- Similarity: $\mathrm{E}(\hat{\epsilon} \mid X) = 0$
- But there are differences:
  - Note $\hat{\epsilon} = (I - P)Y = (I - P)(X\beta + \epsilon) = (I - P)\epsilon$.
  - $\mathrm{cov}(\epsilon \mid X) = \sigma^2 I_n$ whereas $\mathrm{cov}(\hat{\epsilon} \mid X) = \sigma^2(I - P)$.
    - ∗ Even if the errors have equal variance and are uncorrelated, the residuals do not.
    - ∗ When this impact is small, diagnostics can reasonably be applied to residuals to check the assumptions on errors.

### 1.1 Constant variances

- homoscedasticity: constant symmetrical variation in the verticle $\hat{\epsilon}$ direction.

- heteroscedasticity: nonconstant variance.

**Plot of residuals $\hat{\epsilon}$ against fitted values $\hat{y}$**

- $\text{cov}(\hat{\epsilon}, \hat{Y}) = \text{cov}(PY, (I-P)Y) = \sigma^2 P(I-P) = 0$

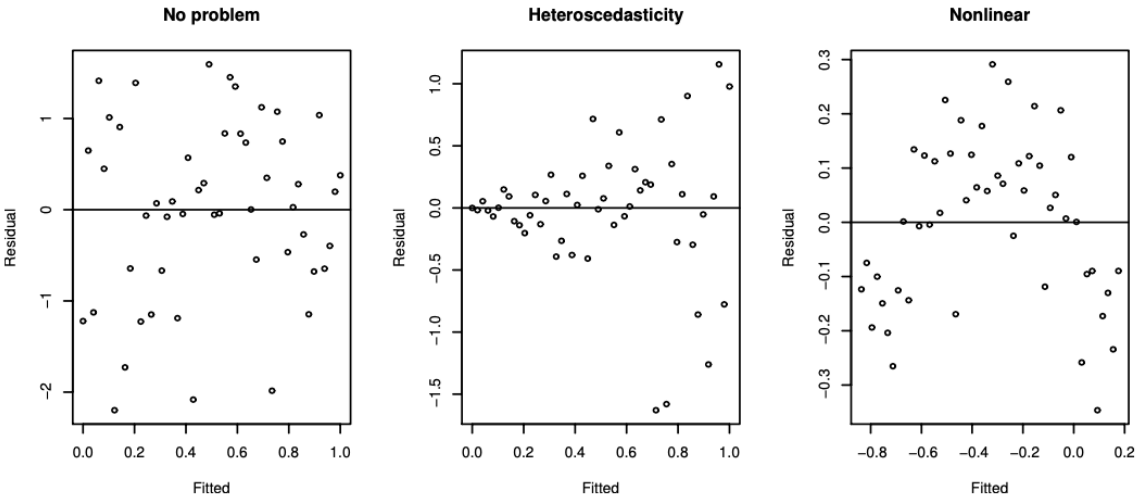
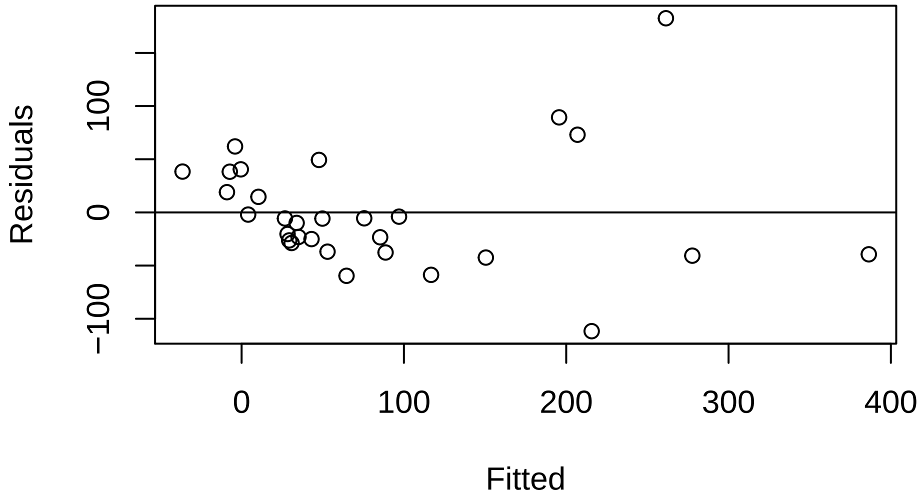


Figure 1: Residuals vs. fitted plots - the first suggests no change to the current model; the second shows nonconstant variance; the third indicates some nonlinearity, which should prompt some change in the structural form of the model.

- Ohter plots $\hat{\epsilon}$ versus $x$, $|\hat{\epsilon}|$ versus $\hat{y}$, or $\sqrt{|\hat{\epsilon}|}$ versus $\hat{y}$ can all be used.
- It could be hard to judge residual plots without prior experience so it is helpful to generate some artificial plots where the true relationship is known. Sample codes provided below.

```
par(mfrow=c(3,3))
for(i in 1:9) plot(1:50,rnorm(50)) #constant variance
for(i in 1:9) plot(1:50,(1:50)*rnorm(50)) #strong non-constant variance
for(i in 1:9) plot(1:50,sqrt((1:50))*rnorm(50)) #mild non-constant variance
for(i in 1:9) plot(1:50,cos((1:50)*pi/25)+rnorm(50)) #nonlinearity
par(mfrow=c(1,1))
```

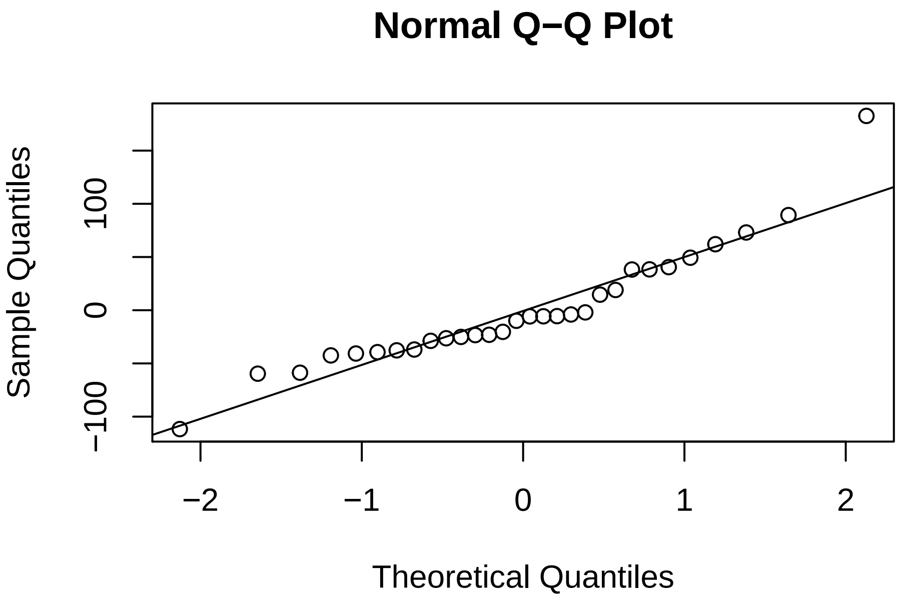
```r
data(gala, package = "faraway")
lmod <- lm(Species ~ Area + Elevation + Scruz + Nearest + Adjacent, gala)
plot(fitted(lmod),residuals(lmod),xlab="Fitted",ylab="Residuals")
abline(h=0)
```



### 1.2 Normality

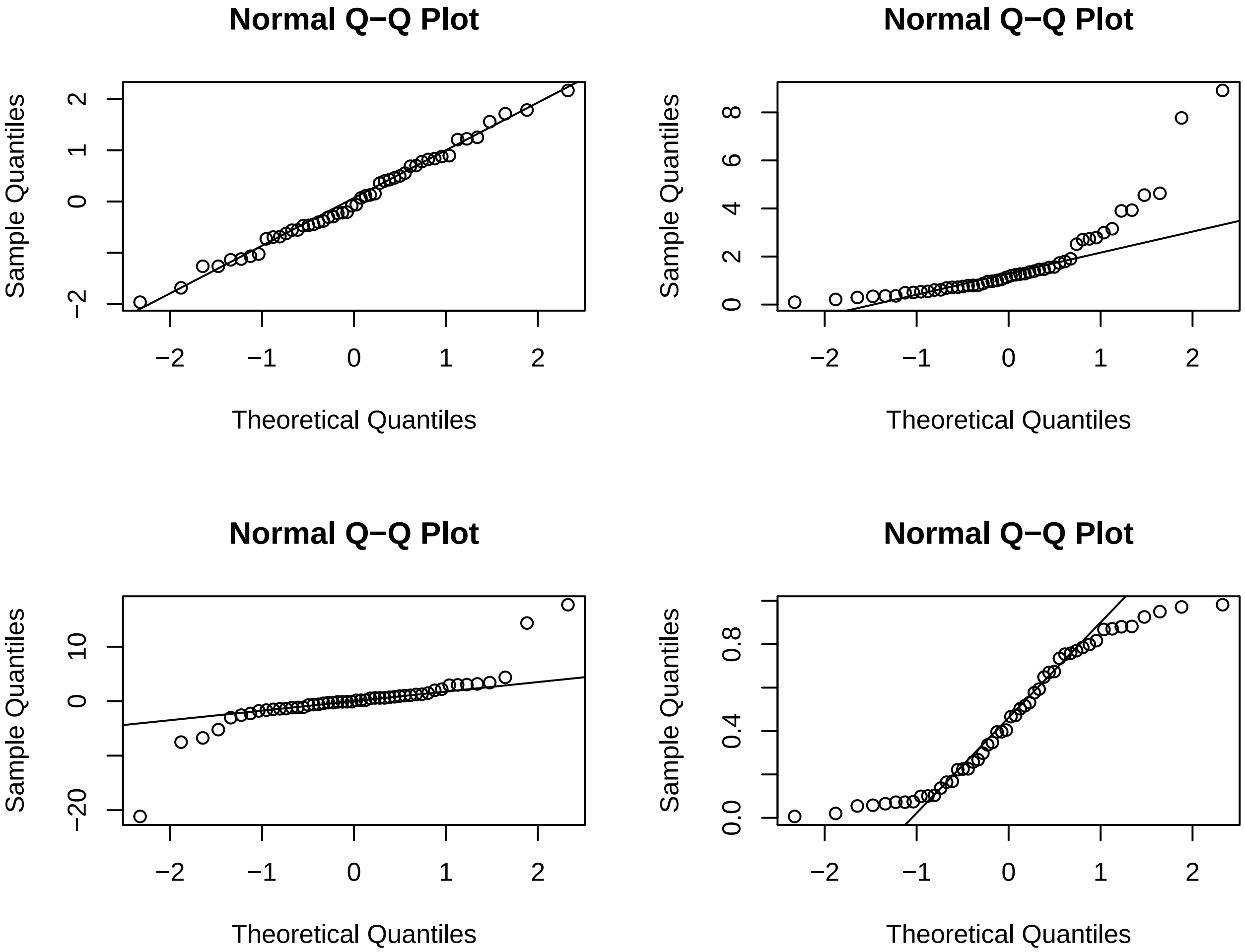
- Tests and confidence intervals are based on the assumption of normal errors.
- The normality of residuals can be assessed by a Q-Q plot. (Histogram is usually not preferred.)
  - Sort sample residuals $\hat{\epsilon}_{(1)} < \ldots < \hat{\epsilon}_{(n)}$.
  - Plot them against $\Phi^{-1}(\frac{1}{n+1}) < \ldots < \Phi^{-1}(\frac{n}{n+1})$ where $\Phi(\cdot)$ denotes the CDF (cumulative distribution fucntion)
  - Normal residuals should follow the line approximately.

```
qqnorm(residuals(lmod))
qqline(residuals(lmod)) #passes through first and third quantiles
```



We can generate sample data from different distributions.

```r
par(mfrow=c(2,2))
set.seed(123)
n = 50
#1. normal
x = rnorm(n); qqnorm(x); qqline(x)
#2. log normal: an example of a skewed distribution
x = exp(rnorm(n)); qqnorm(x); qqline(x)
#3. Cauchy: an example of a long-tailed distribution
x = rcauchy(n); qqnorm(x); qqline(x)
#4. Uniform:  an example of a short-tailed distribution
x = runif(n); qqnorm(x); qqline(x)
```



```
par(mfrow=c(1,1))
```

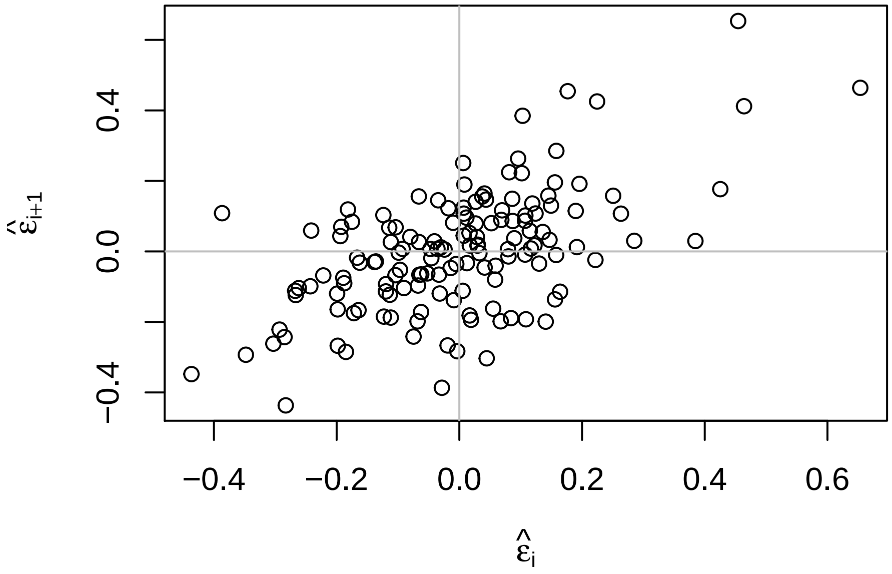
### 1.3 Correlated errors

- It is difficult to check for correlated errors in general because there are just too many possible patterns of correlation that may occur.
  - Data collected over time may have some correlation in successive errors.
  - Spatial data may have correlation in the errors of nearby measurements.
  - Data collected in blocks may show correlated errors within those blocks.

As an example, we consider a serial data on records of annual temperatures.

- The data contains temprature information 1856 through 2000 by Jones and Mann (2004) `Climate over past millennia`.
- We can build a linear model to predict temperature since 1856 and then subsequently use this to predict earlier temperatures based on proxy information.

```
data(globwarm,package="faraway")
lmod <- lm(nhtemp ~ wusa + jasper + westgreen + chesapeake +
tornetrask + urals + mongolia + tasman, globwarm)
n <- length(residuals(lmod))
plot(tail(residuals(lmod),n-1) ~ head(residuals(lmod),n-1), xlab=
expression(hat(epsilon)[i]),ylab=expression(hat(epsilon)[i+1]))
abline(h=0,v=0,col=grey(0.75))
```



- We can see a positive correlation again indicating positive serial correlation. If you have some doubt as to the significance of the correlation.
- We can model the observed correlation directly by linear regression:

```r
summary(lm(tail(residuals(lmod),n-1) ~ head(residuals(lmod),n-1) -1))$coefficients
```

```
##                             Estimate Std. Error  t value     Pr(>|t|)
## head(residuals(lmod), n - 1) 0.5950759 0.06931205 8.585462 1.390651e-14
```

- Note that the the serial correlation is confirmed.

## 2. Finding unusual observations

1. Leverage: A leverage point is extreme in the predictor space.

2. Outliers: Some observations do not fit the model well.
3. Influential: Some observations change the fit of the model in a substantive manner.

### 2.1 Leverage

- $h_{ii} = H_{ii}$, i.e., $i$-th diagonal of hat matrix $H = X(X^\top X)^{-1}X^\top$ are called `leverages`.
- $\mathrm{var}(\hat{\epsilon}_i) = \sigma^2(1 - h_{ii})$
  - a large leverage $h_i$, will make $\mathrm{var}(\hat{\epsilon}_i)$ small.
  - The fit will be attracted toward $y_i$.
  - Large values of $h_i$ are usually due to extreme values in the $X$-space.
- The value of $h_{ii}$ depends only on $X$ and not $y$.
- $\sum_{i=1}^n h_{ii} = \mathrm{tr}(H) = p$ (See notes Sep 26)
  - Average $\sum_{i=1}^n h_{ii}/n = p/n$.
  - A rough rule is that leverages of more than $2p/n$ should be looked at more closely.

```r
data(gala, package = "faraway")
lmod <- lm(Species ~ Area + Elevation + Scruz + Nearest + Adjacent, gala)
hatv <- hatvalues(lmod)
head(hatv)
```

```
##       Baltra    Bartolome     Caldwell     Champion      Coamano Daphne.Major
##   0.07871937   0.09135324   0.06231443   0.07237676   0.16878374   0.07163790
```

```r
sum(hatv) #number of parametrs in the model
```

```
## [1] 6
```

- Standardized residuals:

$$r_i = \frac{\hat{\varepsilon}_i}{\hat{\sigma}\sqrt{1-h_i}}$$

  - Standardization can only correct for the natural non-constant variance in residuals when the errors have constant variance.
  - If there is some underlying heteroscedasticity in the errors, standardization cannot correct for it.
  - When there are unusually large leverages, there could be differences between raw and standardized residuls in their plots.

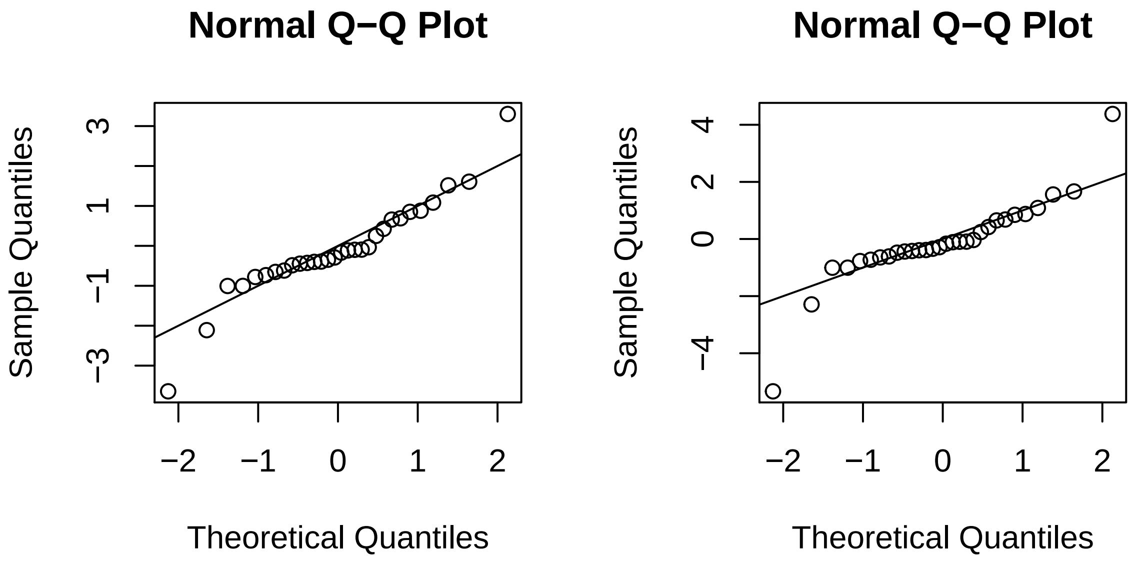
- Leave-one-out studentized residuals:

$$\tilde{r}_i = \frac{\hat{\varepsilon}_i}{\hat{\sigma}_{(i)}\sqrt{1-h_i}}$$

  - $\hat{\sigma}_{(i)}$: estimate $\sigma$ without $i$-th observation.

```r
par(mfrow=c(1,2))
qqnorm(rstandard(lmod))
abline(0,1)

qqnorm(rstudent(lmod))
abline(0,1)
```



```r
par(mfrow=c(1,1))
```

### 2.2 Outliers

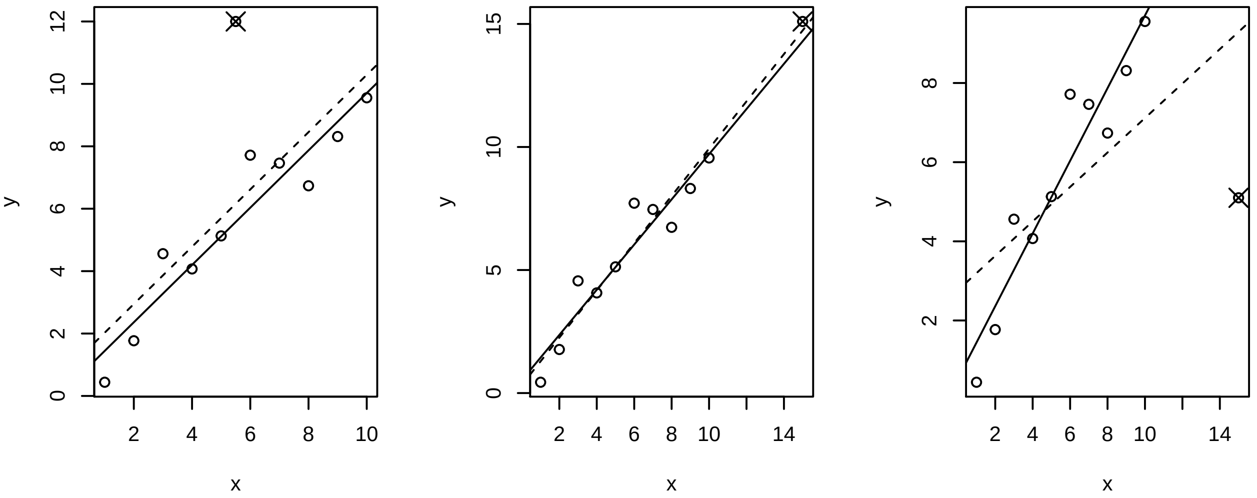
- An outlier is a point that does not fit the current model well.
- Outliers may or may not affect the fit substantially.

```r
set.seed(123)
testdata <- data.frame(x=1:10,y=1:10+rnorm(10))
lmod0 <- lm(y ~ x, testdata)
```

```r
par(mfrow=c(1,3))
p1 <- c(5.5,12)
lmod1 <- lm(y ~ x, rbind(testdata, p1))
plot(y ~ x, rbind(testdata, p1))
points(5.5,12,pch=4,cex=2)
abline(lmod0)
abline(lmod1, lty=2)

p2 <- c(15,15.1)
lmod2 <- lm(y ~ x, rbind(testdata, p2))
plot(y ~ x, rbind(testdata, p2))
points(15,15.1,pch=4,cex=2)
abline(lmod0)
abline(lmod2,lty=2)

p3 <- c(15,5.1)
lmod3 <- lm(y ~ x, rbind(testdata, p3))
plot(y ~ x, rbind(testdata, p3))
points(15,5.1,pch=4,cex=2)
abline(lmod0)
abline(lmod3,lty=2)
```



```r
par(mfrow=c(1,1))
```

- A solid regression line shows the fit without the additional point marked with a cross.
- The dashed line shows the fit with the extra point.
- Left panel:
  - Added point is an outlier.
  - But it does not have large leverage or influential (on the fit).
- Middle panel:
  - Added point has large leverage. (well outside of the range of $X$.)
  - But is not an outlier and is not influential.
- Right panel:
  - Added point changes the fitted line substantially. (influential point.)
  - This is both an outlier and an influential point.

To detect outliers,

- we exclue the point $i$ and recmpute the estimates to get $\hat{\beta}_{(i)}$ and $\hat{\sigma}_{(i)}$.
- Let $\hat{y}_{(i)} = x_i^T \hat{\beta}_{(i)}$
- We have

$$\text{var}\left(y_i - \hat{y}_{(i)}\right) = \sigma^2 \left(1 + x_i^T \left(X_{(i)}^T X_{(i)}\right)^{-1} x_i\right)$$

- Define

$$t_i = \frac{y_i - \hat{y}_{(i)}}{\hat{\sigma}_{(i)} \left(1 + x_i^T \left(X_{(i)}^T X_{(i)}\right)^{-1} x_i\right)^{1/2}},$$

  where $X_{(i)}$ represents the design matrix deleting $i$-th obesrvation.
  - It can be proved $t_i = \frac{\hat{\epsilon}_i}{\hat{\sigma}_{(i)}\sqrt{1-h_{ii}}} = r_i\sqrt{\frac{(n-p-1)}{n-p-r_i^2}}$. (Easy-to-compute. Proof see Theorem 10.1 in Lee and Seber.)
  - If $i$-th case is not outlier, model is correct, and $\epsilon \sim N(0, \sigma^2 I_n)$, $t_i \sim t_{(n-1)-p}$, where $n-1$ is the sample size.
- Test outliers
  - Practically, $|t_i| > 3$ can imply possible outliers.
  - If we want a level $\alpha$ test,
    - $P(\text{all tests accept}) = 1 - P(\text{at least one rejects}) \geqslant 1 - \sum_i P(\text{test } i \text{ rejects }) = 1 - n\alpha$.
    - Each test should use level $\alpha/n$. (Bonferroni correction.)

## 2.3 Influential point

- An influential point is one whose removal from the dataset would cause a large change in the fit.
  - An influential point may or may not be an outlier,
  - and may or may not have large leverage,
  - but it will tend to have at least one of these two properties.
- Measure of influence: Cook's distance statistic Cook (1977)

$$D_i = \frac{\left(\hat{y} - \hat{y}_{(i)}\right)^T \left(\hat{y} - \hat{y}_{(i)}\right)}{p\hat{\sigma}^2}$$

  - It can also be computed as $D_i = \frac{1}{p} r_i^2 \frac{h_{ii}}{1-h_{ii}}$, where $r_i$ represents $i$-th standardized residual.

```r
cook <- cooks.distance(lmod)
summary(lmod)$coefficients
```

```
##                 Estimate  Std. Error      t value     Pr(>|t|)
## (Intercept)  7.068220709 19.15419782  0.369016796 7.153508e-01
## Area        -0.023938338  0.02242235 -1.067610554 2.963180e-01
## Elevation    0.319464761  0.05366280  5.953187968 3.823409e-06
## Scruz       -0.240524230  0.21540225 -1.116628222 2.752082e-01
## Nearest      0.009143961  1.05413595  0.008674366 9.931506e-01
## Adjacent    -0.074804832  0.01770019 -4.226216850 2.970655e-04
```

```r
summary(lmod)$r.squared
```

```
## [1] 0.7658469
```

```r
lmodi <- lm(Species ~ Area + Elevation + Scruz + Nearest + Adjacent,
            gala, subset = (cook<max(cook)))
summary(lmodi)$coefficients
```

```
##                Estimate  Std. Error    t value     Pr(>|t|)
## (Intercept) 22.58614473 13.40191356  1.6852925 1.054542e-01
## Area         0.29574351  0.06186188  4.7807068 8.042013e-05
## Elevation    0.14039023  0.04970484  2.8244782 9.613092e-03
## Scruz       -0.09010457  0.14979821 -0.6015063 5.533860e-01
## Nearest     -0.25518223  0.72167754 -0.3535959 7.268624e-01
## Adjacent    -0.06503051  0.01222732 -5.3184596 2.124483e-05
```

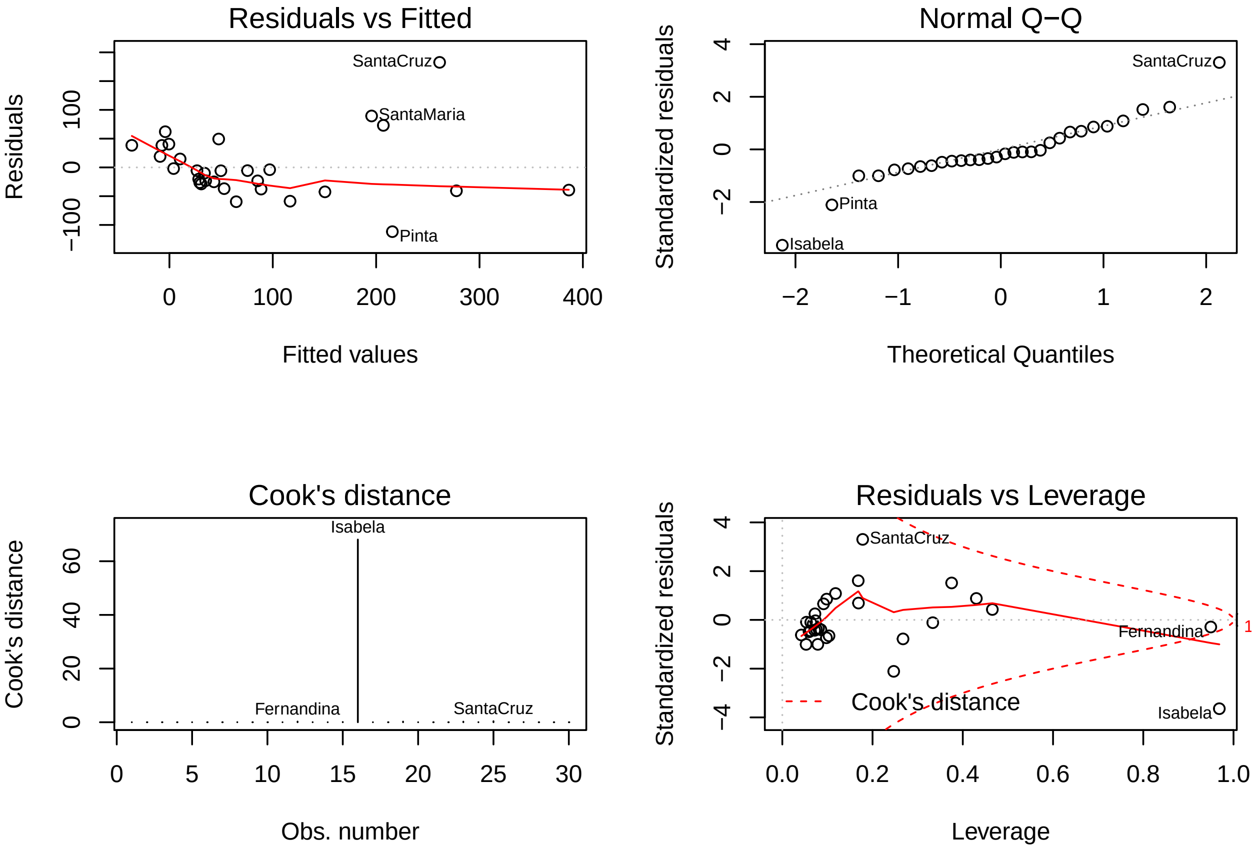
```r
summary(lmodi)$r.squared
```

```
## [1] 0.8714011
```

- p-value of covariate Area changes significantly.

- We usually do not want estimates to be so sensitive to the presence/deletion of just one observation.

```r
par(mfrow=c(2,2))
plot(lmod, which = c(1,2,4, 5), cook.levels = 1) #R codes for multiple diagnostic plots
```



```r
par(mfrow=c(1,1))
```

- As the Cook statistics represent a function of standardized residuals and leverage, we can plot contours (the above plot shows contours with Cook's distance = 1).
- Any point that lies beyond these contours might well be influential and require closer attention.
- A practical guideline: $D_i > 4/n$ can indicate an influential point.

## 3. Checking the Structure of the Model

- Check linear structure $E(Y \mid X) = X\beta$
- Residual plots can suggest transformations of the variables which might improve the structural form of the model.
- A formal lack of test may be conducted in some cases.

# Diagnostics 2

## I. Issues of predictors

### Collinearity of predictors

- When some predictors are linear combinations of others, $X^\top X$ is singular.
  - No unique OLS solution. Theoretical derivations could be problematic.
  - May also be called exact collinearity.
  - A solution could be removing redundant predictors that can be represented by other predictors.
- Another challenging case is $X^\top X$ is nearly singular but not exactly so.
  - Known as `collinearity` or `multicollinearity`.
  - This could lead to imprecise estimates of $\beta$. Intuitively, $(X^\top X)^{-1}$ blows up, and variances are large.
- Detecting (multi)collinearity:
  - Examine correlation matrix of predictors: values close to $-1$ or $+1$ indicate large pairwise correlations/collinearities.
    - ∗ $Y \sim X_1 + X_2 + X_3 + X_4$. If $X^\top X$ is nearly singular, $\mathrm{corr}(X_i, X_j)$ for $i \neq j$ may not be large.
    - ∗ We want to some "generalized correlations".
  - Regressing each $X_j$ on all other predictors $X_{-j}$ for $j = 1, \ldots, p$.
    - ∗ Let $R^2_{X_j|X_{-j}}$ denote the $R^2$ obtained from $X_j \sim X_{-j}$.
    - ∗ Values close to 1 indicate a problem, because it means one predictor can almost be predicted exactly by a linear combination of other predictors.
    - ∗ We can obtain

$$\mathrm{var}(\hat{\beta}_j) = \sigma^2 \frac{1}{1 - R^2_{X_j|X_{-j}}} \times \frac{1}{\sum_{i=1}^n (x_{ij} - \bar{x}_j)^2}$$

- Equivalently, people also use Variance Inflation Factor (VIF)

$$\mathrm{VIF}(\hat{\beta}_j) = \frac{\mathrm{var}(\hat{\beta}_j)_{\text{Full model } Y\sim X_1+\ldots+X_p}}{\mathrm{var}(\hat{\beta}_j)_{\text{Univariate model } Y\sim X_j}} = \frac{1}{1-R^2_{X_j|X_{-j}}}.$$

- If we have the ability to choose the $X$, we can minimize the variance of the regression coefficients by
  - $R^2_{X_j|X_{-j}} = 0$ (Orthogonality design.)
  - Maximizing $\sum_{i=1}^n (x_{ij} - \bar{x}_j)^2 = \widehat{\mathrm{var}}(X_j)$ by spreading $X$ as much as possible.
  - Note this relies on linear model. In pratice, instead of putting all $X$ at maximum/minimum, we also put points in the middle.

**Data Example:**

- Car drivers like to adjust the seat position for their own comfort.
- Car designers would find it helpful to know where different drivers will position the seat depending on their size and age.
- Researchers at the HuMoSim laboratory at the University of Michigan collected data on 38 drivers.
- They measured age in years, weight in pounds, height with shoes and without shoes in centimeters, seated height arm length, thigh length, lower leg length and hipcenter, the horizontal distance of the midpoint of the hips from a fixed location in the car in millimeters.
- We fit a model with all the predictors:

```
data(seatpos, package="faraway")
lmod <- lm(hipcenter ~ ., seatpos)
summary(lmod)$coefficients
```

```
##                 Estimate  Std. Error     t value   Pr(>|t|)
## (Intercept) 436.43212823 166.5716187  2.62008697 0.01384361
## Age           0.77571620   0.5703288  1.36012113 0.18427175
## Weight        0.02631308   0.3309704  0.07950283 0.93717877
## HtShoes      -2.69240774   9.7530351 -0.27605845 0.78446097
## Ht            0.60134458  10.1298739  0.05936348 0.95306980
## Seated        0.53375170   3.7618942  0.14188376 0.88815293
## Arm          -1.32806864   3.9001969 -0.34051323 0.73592450
## Thigh        -1.14311888   2.6600237 -0.42974011 0.67056106
## Leg          -6.43904627   4.7138601 -1.36598163 0.18244531
```

- This model shows the signs of collinearity.
  - S.E. is large. $R^2$ is not small, but none of the individual predictors is significant.
- We can take a look at the pairwise correlations:

```
round(cor(seatpos[,-9]),2)
```

```
##           Age Weight HtShoes    Ht Seated  Arm Thigh   Leg
## Age      1.00   0.08   -0.08 -0.09  -0.17 0.36  0.09 -0.04
## Weight   0.08   1.00    0.83  0.83   0.78 0.70  0.57  0.78
## HtShoes -0.08   0.83    1.00  1.00   0.93 0.75  0.72  0.91
## Ht      -0.09   0.83    1.00  1.00   0.93 0.75  0.73  0.91
## Seated  -0.17   0.78    0.93  0.93   1.00 0.63  0.61  0.81
## Arm      0.36   0.70    0.75  0.75   0.63 1.00  0.67  0.75
## Thigh    0.09   0.57    0.72  0.73   0.61 0.67  1.00  0.65
## Leg     -0.04   0.78    0.91  0.91   0.81 0.75  0.65  1.00
```

- There are several large pairwise correlations between predictors.
- We can check the VIFs.

```
x <- model.matrix(lmod)[,-1]
(rsq_1 <- summary(lm(x[,1] ~ x[,-1]))$r.squared)
```

```
## [1] 0.4994823
```

```
#calculate VIF of first covariate by definition
1/(1-rsq_1)
```

```
## [1] 1.997931
```

```
#By R function
library(faraway)
vif(x)
```

```
##        Age     Weight    HtShoes         Ht     Seated        Arm      Thigh
##   1.997931   3.647030 307.429378 333.137832   8.951054   4.496368   2.762886
##        Leg
##   6.694291
```

# II. Some remedies of error issues

- We have seen that assumptions of errors can be violated and we must then consider alternatives.
- When the errors are dependent, we can use generalized least squares (GLS).
- When the errors are independent, but not identically distributed, we can use weighted least squares (WLS), which is a special case of GLS.

## II.1 Generalized Least Squares

- We have assumed $\epsilon = \sigma^2 I$.

**If $\Sigma$ is known:**

- Suppose instead $\text{var}(\epsilon) = \sigma^2 \Sigma$.
  - $\sigma^2$ is unknown but $\Sigma$ is known.
  - that is, we know the correlation and relative variance between the errors,
  - but we do not know the absolute scale of the variation.
- Write $\Sigma = SS^\top$. We can transform the regression model as

$$
\begin{aligned}
Y &= X\beta + \varepsilon \\
S^{-1}Y &= S^{-1}X\beta + S^{-1}\varepsilon \quad \Rightarrow \quad \tilde{Y} \ = \tilde{X}\beta + \tilde{\varepsilon}
\end{aligned}
$$

- Then OLS can be conducted to the transformed variables $\tilde{Y}$ and $\tilde{X}$.
- Then

$$
\begin{aligned}
\hat{\beta} =& (\tilde{X}^\top \tilde{X})^{-1} \tilde{X}^\top \tilde{Y} = \left( X^T \Sigma^{-1} X \right)^{-1} X^T \Sigma^{-1} Y \\
\text{var}(\hat{\beta}) =& \sigma^2 (\tilde{X}^\top \tilde{X})^{-1} = \sigma^2 (X^\top \Sigma^{-1} X)^{-1}.
\end{aligned}
$$

- Also diagnostics should be applied to the tranformed residuals $S^{-1}\hat{\varepsilon}$, which should be approximately i.i.d.

**If $\Sigma$ is unknown:**

- We need to estimate $\Sigma$. Can be done through R function `gls`.
- Recall the temperature data that we investigated where serial correlation was observed.

```r
data(globwarm,package="faraway")
lmod <- lm(nhtemp ~ wusa + jasper + westgreen + chesapeake
           + tornetrask + urals + mongolia + tasman, globwarm)
sumary(lmod)
```

```
##               Estimate Std. Error t value  Pr(>|t|)
## (Intercept) -0.2425552  0.0270115 -8.9797 1.972e-15
## wusa         0.0773844  0.0429266  1.8027 0.0736475
## jasper      -0.2287948  0.0781074 -2.9292 0.0039859
## westgreen    0.0095839  0.0418405  0.2291 0.8191679
## chesapeake  -0.0321117  0.0340522 -0.9430 0.3473462
## tornetrask   0.0926676  0.0450530  2.0569 0.0416114
## urals        0.1853691  0.0914285  2.0275 0.0445674
## mongolia     0.0419725  0.0457935  0.9166 0.3609955
## tasman       0.1154529  0.0301110  3.8342 0.0001919
##
## n = 145, p = 9, Residual SE = 0.17577, R-Squared = 0.48
```

```r
cor(residuals(lmod)[-1],residuals(lmod)[-length(residuals(lmod))])
```

```
## [1] 0.583339
```

- nlme packageof Pinheiro and Bates (2000) contains a GLS fitting function.
  - restricted maximum likelihood (ReML)
- Consider a autoregressive model $\varepsilon_{i+1} = \phi\varepsilon_i + \delta_i$, where $\delta_i \sim N(0, \tau^2)$.

```r
require(nlme)
glmod <- gls(nhtemp ~ wusa + jasper + westgreen + chesapeake
             + tornetrask + urals + mongolia + tasman,
             correlation=corAR1(form=~year),  data=na.omit(globwarm))
# na.omit(globwarm) drops missing values
summary(glmod)
```

```
## Generalized least squares fit by REML
##   Model: nhtemp ~ wusa + jasper + westgreen + chesapeake + tornetrask +      urals +
##   Data: na.omit(globwarm)
##         AIC       BIC   logLik
##   -108.2074 -76.16822 65.10371
##
## Correlation Structure: AR(1)
##  Formula: ~year
##  Parameter estimate(s):
##       Phi
## 0.7109922
##
## Coefficients:
##                   Value  Std.Error   t-value p-value
## (Intercept) -0.23010624 0.06702406 -3.433188  0.0008
## wusa         0.06673819 0.09877211  0.675678  0.5004
## jasper      -0.20244335 0.18802773 -1.076668  0.2835
## westgreen   -0.00440299 0.08985321 -0.049002  0.9610
## chesapeake  -0.00735289 0.07349791 -0.100042  0.9205
## tornetrask   0.03835169 0.09482515  0.404446  0.6865
## urals        0.24142199 0.22871028  1.055580  0.2930
## mongolia     0.05694978 0.10489786  0.542907  0.5881
## tasman       0.12034918 0.07456983  1.613913  0.1089
##
##  Correlation:
##            (Intr) wusa   jasper wstgrn chespk trntrs urals  mongol
## wusa       -0.517
## jasper     -0.058 -0.299
## westgreen   0.330 -0.533  0.121
## chesapeake  0.090 -0.314  0.230  0.147
## tornetrask -0.430  0.499 -0.197 -0.328 -0.441
## urals      -0.110 -0.142 -0.265  0.075 -0.064 -0.346
## mongolia    0.459 -0.437 -0.205  0.217  0.449 -0.343 -0.371
## tasman      0.037 -0.322  0.065  0.134  0.116 -0.434  0.416 -0.017
##
## Standardized residuals:
```

```
##         Min          Q1         Med          Q3         Max
## -2.31122523 -0.53484054  0.02342908  0.50015642  2.97224724
##
## Residual standard error: 0.204572
## Degrees of freedom: 145 total; 136 residual
```

### II.2 Weighted least squares

- Errors are uncorrelated, but have unequal and unknown variances.
- $\Sigma = \operatorname{diag}(1/w_1, \ldots, 1/w_n)$, where $w_i$ are the weights.
- By GLS, $S = \operatorname{diag}(1/\sqrt{w_1}, \ldots, 1/\sqrt{w_n})$.
- Regress $S^{-1}Y$ on $S^{-1}X$, i.e., $\sqrt{w_i}y_i \sim \sqrt{w_i}x_i$.
- Data example:
  - We consider an experiment studying interactions of unstable elementary particles in collision with proton targets (Weisberg et al., 1978).
  - These particles interact via the so-called strong interaction force that holds nuclei together.
  - The experiment was carried out with beam having various values of incident momentum, or equivalently for various values of $s$, the square of the total energy in the center-of-mass frame of reference system.
  - For each value of $s$, we observe the scattering cross-section $y$, measured in millibarns.
  - A theoretical model of the strong interaction force predicts that

$$\mathrm{E}(y \mid s) = \beta_0 + \beta_1 s^{-1/2} + \text{ relatively small terms}$$

    (leading part is a simple linear regression.)
  - At each value of $s$, a very large number of particles was counted, and as a result the values of $\operatorname{Var}(y \mid s = s_i) = \sigma^2/w_i$ are known almost exactly.

```r
library(alr4)
head(physics1,3)
```

```
##       x   y SD
## 1 0.345 284 13
## 2 0.287 288  9
## 3 0.251 304  9
```

- In this case, variances are not constant but are different for each value of $s$.
- Therefore, WLS should be used.

# III. Transformations

- We have seen that transformations of the response and/or predictors can improve the fit and correct violations of model assumptions, such as non-constant error variance.

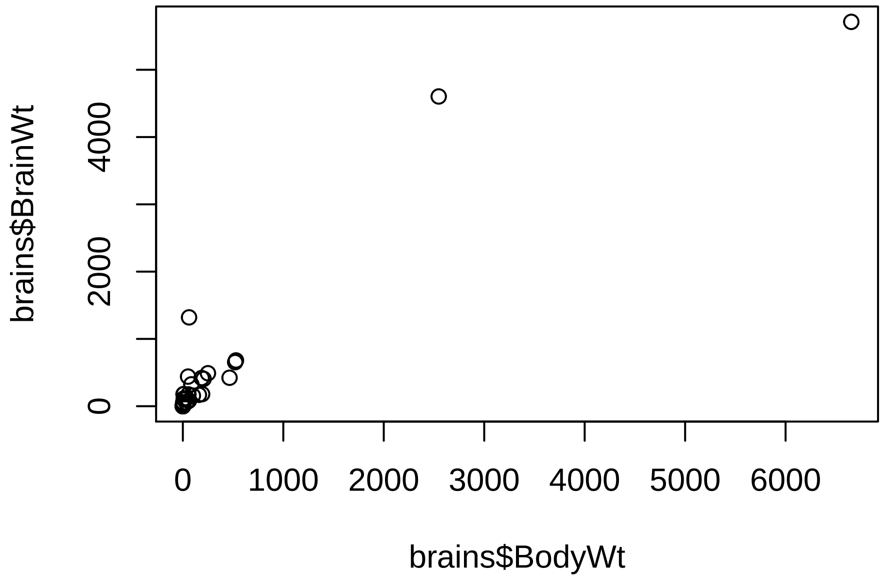
## III.1 Transform the responses

- Transforming responses is to achieve a mean function that is linear in the transformed scale: $\mathrm{E}(\tilde{Y} \mid \tilde{X}) \approx \beta_0 + \beta_1 \tilde{X}$.
  - In simple linear regression:
    - let scatterplot have an approximate straight line mean function
  - In multiple linear regression:
    - Harder, but we can consider one predictor case first.
- A transformation family is a collection of transformations indexed by one or a few parameters that the analyst can select.

```r
library(alr4)
head(brains,3)
```

```
##             BrainWt BodyWt
## Arctic fox   44.500  3.385
## Owl monkey   15.499  0.480
## Beaver        8.100  1.350
```

```r
plot(brains$BodyWt, brains$BrainWt )
```



- Little evidence of linear mean function.

**Box-Cox transformation**

- Named after statisticians George Box and Sir David Roxbee Cox (1964)
- Designed for strictly positive responses and chooses the transformation to find the best fit to the data: $y \to g_\lambda(y)$

$$g_\lambda(y) = \begin{cases} \frac{y^\lambda - 1}{\lambda} & \lambda \neq 0 \\ \log y & \lambda = 0 \end{cases}$$

- Choose $\lambda$ that maximizes

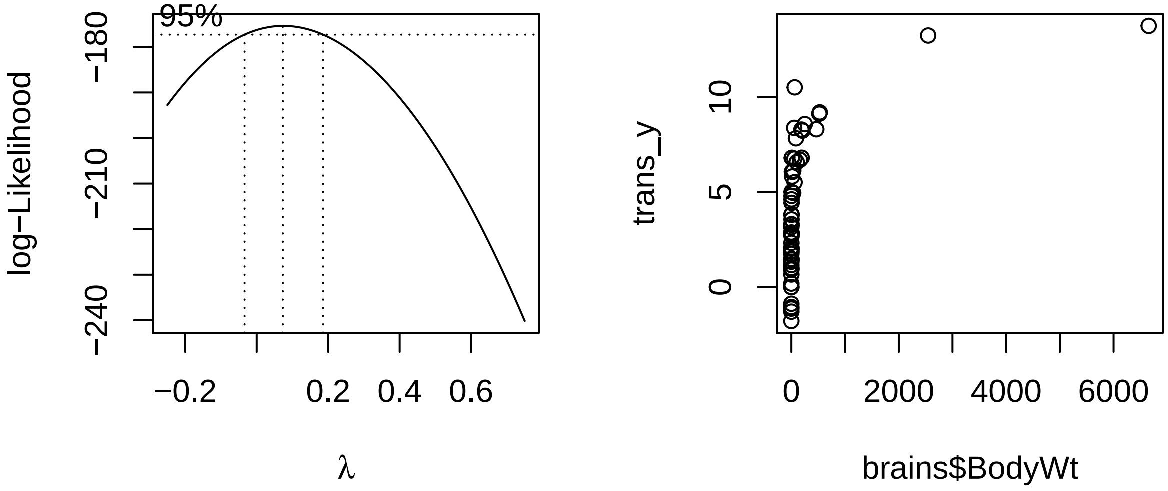
$$L(\lambda) = -\frac{n}{2} \log\left(\text{RSS}_\lambda / n\right) + (\lambda - 1) \sum \log y_i$$

where $\text{RSS}_\lambda$ is the residual sum of squares when $g_\lambda(y)$ is the response.

- A $100(1-\alpha)\%$ confidence interval for $\lambda$ is (from $H_0 : \lambda = \lambda_0$):

$$\left\{\lambda : \quad L(\lambda) > L(\hat{\lambda}) - \frac{1}{2}\chi_1^{2(1-\alpha)}\right\}$$

```r
require(MASS)
par(mfrow=c(1,2))
lmod <- lm( BrainWt ~ BodyWt, brains)
boxcox(lmod, lambda=seq(-0.25,0.75,by=0.05),plotit=T)
lambda_opt <- 0.1
trans_y <- (brains$BrainWt^(lambda_opt)-1)/(lambda_opt)
plot(brains$BodyWt, trans_y )
```



```r
par(mfrow=c(1,1))
```

`Remark`

- Conclusion can be influenced by outliers.
- If some $y_i < 0$, we can add a constant to all the $y_i$.

## III.2 Transform the predictors

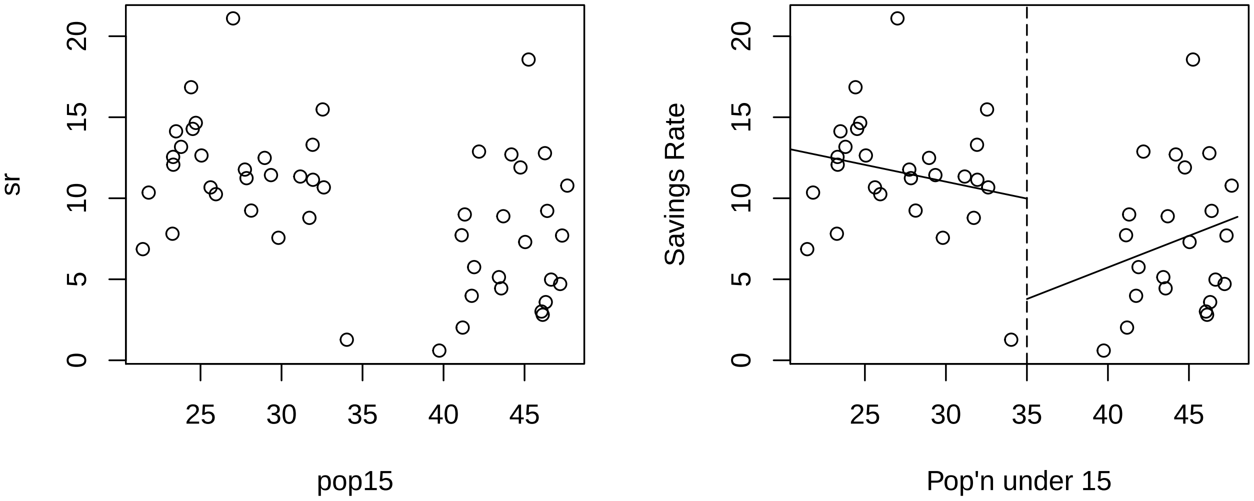
### Broken Stick Regression

- Sometimes we have reason to believe that different linear regression models apply in different regions of the data.
- Analyzing a data on savings rates in 50 countries (averaged over the period 1960- 1970.)
  - sr: savings rate - personal saving divided by disposable income
  - pop15: percent population under age of 15

```
par(mfrow=c(1,2))
plot(sr ~ pop15,savings)

lmod1 <- lm(sr ~ pop15, savings, subset=(pop15 < 35))
lmod2 <- lm(sr ~ pop15, savings, subset=(pop15 > 35))
plot(sr ~ pop15,savings,xlab="Pop'n under 15",  ylab="Savings Rate")

abline(v=35,lty=5)
segments(20,lmod1$coef[1]+lmod1$coef[2]*20,35,  lmod1$coef[1]+lmod1$coef[2]*35)
segments(48,lmod2$coef[1]+lmod2$coef[2]*48,35,  lmod2$coef[1]+lmod2$coef[2]*35)
```



```
lhs <- function(x) ifelse(x < 35,35-x,0)
rhs <- function(x) ifelse(x < 35,0,x-35)
lmod <- lm(sr ~ lhs(pop15) + rhs(pop15), savings)
x <- seq(20,48,by=1)
```

```
par(mfrow=c(1,1))
```

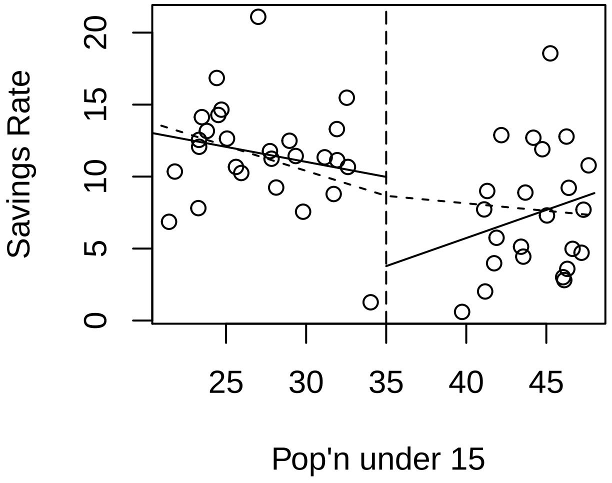
- Subsetted regression fit is that the two parts of the fit do not meet at the join.
- If we believe the fit should be continuous as the predictor varies, we should consider the broken stick regression fit.
- Define

$$B_l(x) = \begin{cases} c - x & \text{if } x < c \\ 0 & \text{otherwise} \end{cases} \qquad B_r(x) = \begin{cases} x - c & \text{if } x > c \\ 0 & \text{otherwise} \end{cases}$$

  where $c$ marks the division between two groups.
- Fit a model of the form:

$$y = \beta_0 + \beta_1 B_l(x) + \beta_2 B_r(x) + \varepsilon$$

```
plot(sr ~ pop15,savings,xlab="Pop'n under 15",  ylab="Savings Rate")
abline(v=35,lty=5)
segments(20,lmod1$coef[1]+lmod1$coef[2]*20,35,  lmod1$coef[1]+lmod1$coef[2]*35)
segments(48,lmod2$coef[1]+lmod2$coef[2]*48,35,  lmod2$coef[1]+lmod2$coef[2]*35)
lhs <- function(x) ifelse(x < 35,35-x,0)
rhs <- function(x) ifelse(x < 35,0,x-35)
lmod <- lm(sr ~ lhs(pop15) + rhs(pop15), savings)
x <- seq(20,48,by=1)
py <- lmod$coef[1] + lmod$coef[2]*lhs(x) + lmod$coef[3]*rhs(x)
lines(x,py,lty=2)
```



**Polynomial regression**

- Generalizing $X\beta$ part by adding polynomial terms:
  - e.g.,
    $$y = \beta_0 + \beta_1 x + \cdots + \beta_d x^d + \varepsilon$$
    allows for a more flexible relationship.
- Adding a quadratic term:
  - e.g., there may be a best temperature for baking bread – a hotter or colder temperature may result in a less tasty outcome.
  - If you believe a predictor behaves in this manner, it makes sense to add a quadratic term.
- Again look at `savings` dataset
  - dpi: per-capita disposable income in dollars
  - ddpi: percent growth rate of dpi
- Choosing $d$:
  - Keep adding terms until the added term is not statistically significant.
  - Start with a large $d$ and eliminate non-statistically significant terms starting with the highest order term.

```
summary(lm(sr ~ ddpi,savings))$coefficients #linear of ddpi is significant
```

```
##             Estimate Std. Error  t value     Pr(>|t|)
## (Intercept) 7.883021  1.0110011 7.797243 4.464697e-10
## ddpi        0.475830  0.2146166 2.217117 3.138509e-02
```

```
summary(lm(sr ~ ddpi+I(ddpi^2),savings))$coefficients #quadratic of ddpi is significant
```

```
##                Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)  5.13038069 1.43471517  3.575888 0.0008211413
## ddpi         1.75751897 0.53772368  3.268443 0.0020258542
## I(ddpi^2)   -0.09298521 0.03612318 -2.574115 0.0132617330
```

```
summary(lm(sr ~ ddpi+I(ddpi^2)+I(ddpi^3),savings))$coefficients  #cubit of ddpi is NOT
```

```
##                  Estimate Std. Error      t value   Pr(>|t|)
## (Intercept)  5.145360e+00 2.19860644  2.340282237 0.02366212
## ddpi         1.746017e+00 1.38045499  1.264812459 0.21230898
## I(ddpi^2)   -9.096724e-02 0.22559835 -0.403226554 0.68864973
## I(ddpi^3)   -8.496955e-05 0.00937393 -0.009064453 0.99280691
```

- You have to refit the model each time a term is removed which is inconvenient and numerically unstable.
- Orthogonal polynomials get around this problem by defining:

$$
\begin{aligned}
z_1 &= a_1 + b_1 x \\
z_2 &= a_2 + b_2 x + c_2 x^2 \\
z_3 &= a_3 + b_3 x + c_3 x^2 + d_3 x^3
\end{aligned}
$$

  where coefficients are chosen so that $z_i^\top z_j = 0$ when $i \neq j$.
- In r, poly() function constructs orthogonal polynomials:

```
lmod <- lm(sr ~ poly(ddpi,4),savings)
summary(lmod)$coefficients
```

```
##                     Estimate Std. Error      t value     Pr(>|t|)
## (Intercept)       9.67100000   0.584602 16.542879686 9.477039e-21
## poly(ddpi, 4)1    9.55899338   4.133760  2.312420904 2.538538e-02
## poly(ddpi, 4)2  -10.49987612   4.133760 -2.540030321 1.460646e-02
## poly(ddpi, 4)3   -0.03737382   4.133760 -0.009041119 9.928263e-01
## poly(ddpi, 4)4    3.61196847   4.133760  0.873773113 3.868811e-01
```

- You can also define polynomials in more than one variable. These are sometimes called response surface models.
- A second degree model would be like:

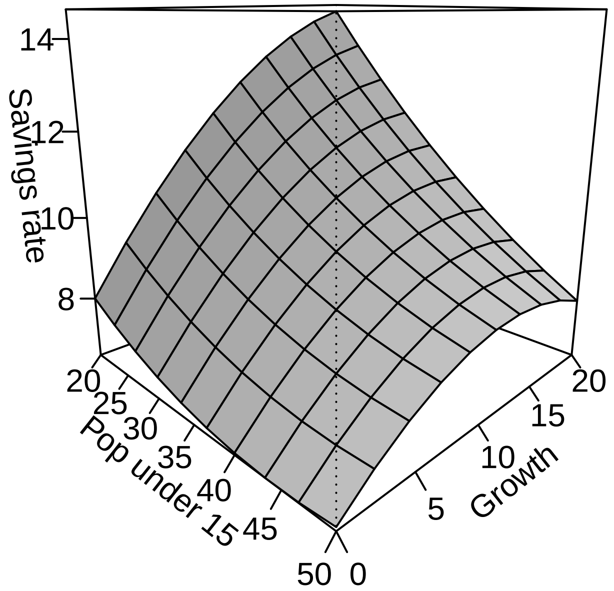
$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{11} x_1^2 + \beta_{22} x_2^2 + \beta_{12} x_1 x_2$$

```
lmod <- lm(sr ~ polym(pop15,ddpi,degree=2),savings)
```

- We can construct a perspective plot of the fitted surface.

```
pop15r <- seq(20, 50, len=10) #choose grid vlaues
ddpir <- seq(0, 20, len=10) #choose grid vlaues
pgrid <- expand.grid(pop15=pop15r, ddpi=ddpir) #construct 2-dim grids
pv <- predict(lmod, pgrid)
```

```
persp(pop15r, ddpir, matrix(pv, 10, 10), theta=45,
      xlab="Pop under 15", ylab="Growth", zlab = "Savings rate",
      ticktype="detailed", shade = 0.25)
```



## IV. Measurement errors

- Errors in both responses $Y$ and covariates $X$.
- Suppose that what we observe is $(Y, X)$.
- But the true relationship is

$$Y = Z\beta + \epsilon \qquad \text{with} \qquad X = Z + \tau,$$

  so $\tau$ denotes errors of measuring true $Z$.
- Suppose we use observed data to calculate least squares estimates

$$\begin{aligned}\hat{\beta} &= (X'X)^{-1} X'Y \\ &= (Z'Z + Z'\tau + \tau'Z + \tau'\tau)^{-1} (Z'Y + \tau'Y) \\ &= \left(\frac{Z'Z}{n} + \frac{Z'\tau}{n} + \frac{\tau'Z}{n} + \frac{\tau'\tau}{n}\right)^{-1} \left(\frac{Z'Y}{n} + \frac{\tau'Z\beta}{n} + \frac{\tau'\epsilon}{n}\right)\end{aligned}$$

  where we plug in $Y = Z\beta + \epsilon$ and $X = Z + \tau$.
- In many cases, it can be assumed that
  - the covariate measurement error is uncorrelated with the covariate levels $Z'\tau/n \to 0$
  - covariate measurement error and observation error are uncorrelated $\tau'\epsilon/n \to 0$
- Then

$$\begin{aligned}\hat{\beta} \to & \left(\frac{Z'Z}{n} + \frac{\tau'\tau}{n}\right)^{-1} Z'Y/n = (M_z + M_\tau)^{-1} \frac{Z'Y}{n} \\ = & \ (M_z + M_\tau)^{-1} \left(\frac{Z'Z}{n}\beta + \frac{Z'}{n}\epsilon\right) \to (M_z + M_\tau)^{-1} M_z\beta,\end{aligned}$$

  where $M_z = Z'Z/n$ and $M_\tau = \tau'\tau/n$.
- The limiting bias is

$$\hat{\beta} - \beta = \left(\left(I + M_z^{-1}M_\tau\right)^{-1} - I\right)\beta.$$

#### SIMEX (SIMulation-EXtrapolation)

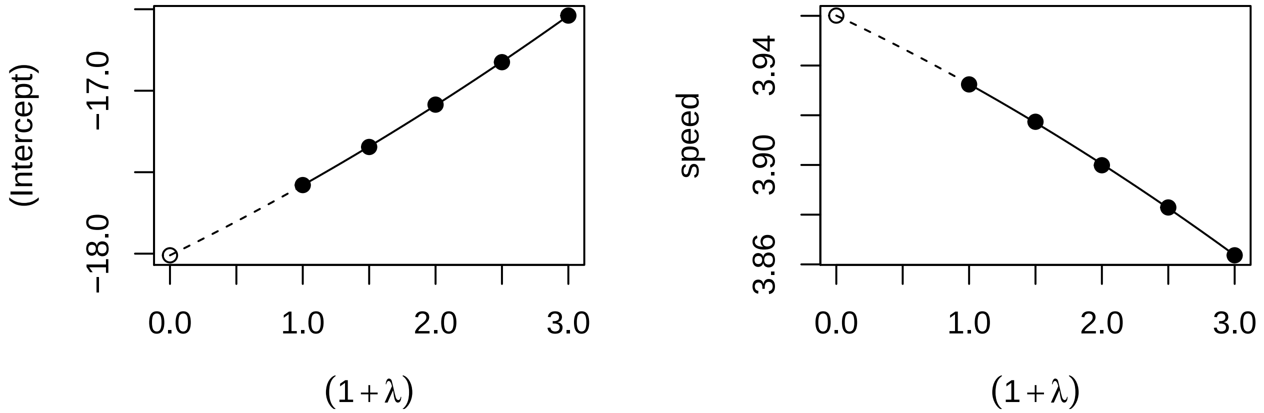
- If the variances and covariances among the measurement errors can be considered known.
- Regress $Y$ on $X + \lambda E$
  - $E$ is simulated noise having the same variance as the assumed measurement error.
- Denote the coefficient vector of this fit as $\hat{\beta}_\lambda$.
- Repeat this for several values of $\lambda \geq 0$, leading to a set of $\hat{\beta}_\lambda$ vectors.
  - Ideally, $\hat{\beta}_{-1}$ would approximate the coefficient estimates under no measurement error.

- By fitting a line or smooth curve to the $\hat{\beta}_\lambda$ values (separately for each component of $\beta$ ), it becomes possible to extrapolate back to $\hat{\beta}_{-1}$.

```r
require(simex)
set.seed(123)
lmod <- lm(dist ~ speed, cars, x=TRUE) #stopping distances vs speed
#Suppose the predictor, speed, was measured with
#a known error standard deviation, say 0.5.
simout <- simex(lmod,"speed",0.5, B=1000)
simout
```

```
##
## Naive model:
## lm(formula = dist ~ speed, data = cars, x = TRUE)
##
## SIMEX-Variables: speed
## Number of Simulations: 1000
##
## Coefficients:
## (Intercept)        speed
##      -18.01         3.96
```

```r
par(mfrow=c(1,2))
plot(simout)
```


(Intercept)
(1 + λ)
speed
(1 + λ)