Lecture 9 and 10 Handout

Elizabeth Colantuoni

2/20/2021

I. Objectives:

Upon completion of lectures 9 and 10, you will be able to do the following:

- Understand the bias introduced into models when key variables are excluded
- Check the mean model by plotting and smoothing residuals versus predicted values
- Check the mean model using added variable plots
- Check the variance assumption by plotting and smoothing squared residuals against predicted values
- Check the independence assumption using autocorrelation functions
- Apply the general least squares or weighted least squares to account for heteroskedasticity
- Derive the robust variance estimator for linear regression
- Checking for undue influence using DBETAS and DFITS

II. General form of linear model

We have established the multiple linear regression model:

$$Y_{n\times 1} = X_{n\times(p+1)}\beta_{(p+1)\times 1} + \epsilon_{n\times 1}, \epsilon_{n\times 1} \sim MVN(0_{n\times 1}, \sigma^2 I_{n\times n})$$

III. Linear regression model assumptions

The key assumptions from the model by order of importance are:

1. We **assume** that the mean of Y is given by $X_{n\times(p+1)}\beta_{(p+1)\times 1}$. There can be violations of this assumption including missing predictors, wrong functional form (e.g. linear vs. non-linear functions), missing interactions and errors in predictors. **Violations of this assumption affect/bias β**
2. $Cov(\epsilon_i, \epsilon_j) = 0$ i.e. errors are independent of each other. There can be violations of this assumption if the data is generated via a clustered sampling design (e.g. family members nested within households, household nested within villages) or a longitudinal design (e.g. participants followed over time and repeated measures of the outcome recorded). **Violations of this assumption affect the variance estimates of $\hat{\beta}$ and therefore our inferences about β**
3. $Var(Y_i|X) = v_i$ is not constant, but may depend on X, i.e. $Var(Y_i|X) = v_i(X)$. **Violations of this assumption affect the variance estimates of $\hat{\beta}$ and therefore our inferences about β**

4. ϵ_i not normally distributed. We saw this violation in the medical expenditure example. **Violations of this assumption affect the variance estimates of $\hat{\beta}$ and therefore our inferences about β**

5. A small fraction of data has high influence on the model fit. **Violations of this assumption can affect estimation and inference on β**

A. Omitted variable bias

What happens if we leave out important variables in our linear model? **The estimates of the β for the covariates included may be biased.**

We will derive the bias in the estimate of the regression coefficient corresponding to X_1 in a simple linear regression model, when in fact there should be another variable X_2 included in the model.

$$\text{True Model: } Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \epsilon_i$$

$$\text{Model Fit: } Y_i = \alpha_0 + \alpha_1 X_{1i} + u_i$$

Based on the least squares solution, we know that

$$\begin{aligned}
\hat{\alpha_1} &= \tfrac{Cov(Y_i, X_{1i})}{Var(X_{1i})} \\
&= \tfrac{Cov(\beta_0+\beta_1 X_{1i}+\beta_2 X_{2i}+\epsilon_i, X_{1i})}{Var(X_{1i})} \text{ by substituting what is true for } Y_i \\
&= \tfrac{Cov(\beta_0, X_{1i})+Cov(\beta_1 X_{1i}, X_{1i})+Cov(\beta_2 X_{2i}, X_{1i})+Cov(\epsilon_i, X_{1i})}{Var(X_{1i})} \text{ but } Cov(\beta_0, X_{1i}) = 0 \\
&= \tfrac{\beta_1 Cov(X_{1i}, X_{1i})+\beta_2 Cov(X_{2i}, X_{1i})+Cov(\epsilon_i, X_{1i})}{Var(X_{1i})} \text{ but } Cov(\epsilon_i, X_{1i}) = 0 \text{ under the true model} \\
&= \tfrac{\beta_1 Var(X_{1i})+\beta_2 Cov(X_{2i}, X_{1i})}{Var(X_{1i})} \\
&= \beta_1 + \tfrac{\beta_2 Cov(X_{2i}, X_{1i})}{Var(X_{1i})} \\
\hat{\alpha_1} &= \beta_1 + \beta_2 \hat{\delta}_1 \text{ where } \hat{\delta} = \tfrac{Cov(X_{2i}, X_{1i})}{Var(X_{1i})} \text{ from a simple linear regression of } X_{2i} = \delta_0 + \delta_1 X_{1i} + v_i
\end{aligned}$$

Therefore, the bias, $\hat{\alpha_1} - \beta_1$, will be 0 when

- $\beta_2 = 0$ i.e. X_2 is not important
- $\hat{\delta_1} = 0$ i.e. when X_1 and X_2 are uncorrelated.

Note that the bias may be positive or negative and the direction of the bias will be determined by the relationship between X_1 and X_2 and the adjusted relationship between Y and X_2.

B. Functional form of Xs

One part of the assumption $E(Y|X) = X\beta$ requires that the functional form we specified to describe $E(Y|X)$ as a function of a continuous X is correct. Again, if we assume that $E(Y|X)$ is linear but in fact it is non-linear, then our slope estimate based on the linear assumption is misleading/biased.

To explore the assumption that $E(Y|X) = X\beta$, you can make the following plots:

- Plot $\hat{R}$ vs. $X_j, j = 1, ..., p$. Recall that the residuals are independent of X if the model is correctly specified

- Plot $\hat{R}$ vs. $\hat{Y}$. The residuals and predicted values are independent if the model is correctly specified
- Never plot $\hat{R}$ vs. Y because these two are correlated!

Based on the plot of $\hat{R}$ vs. X_j or $\hat{Y}$ then modify the model to increase/decrease the complexity of the functional form of the variables.

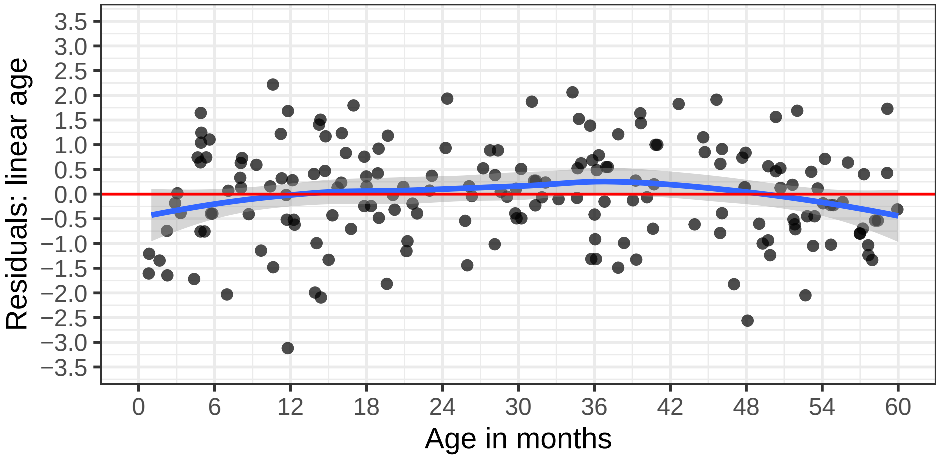
1. Example using Nepali Anthropometry data

```
load("NepalAnthro.rdata")
d = nepal.anthro %>%filter(.,num==1)
d = mutate(d,
agesp6=ifelse(age-6>0, age-6,0),
agesp12=ifelse(age-12>0,age-12,0)
)
cc=complete.cases(d[,c("age","arm")])
d.cc=filter(d,cc)
d.cc = arrange(d.cc,age)
reg0<-lm(data=d.cc, arm~age)
d.cc$residuals = residuals(reg0)
```

```
ggplot(d.cc,aes(x=age, y=residuals)) +
    geom_jitter(alpha = 0.7) +
    theme_bw() +
    geom_smooth() +
    geom_hline(yintercept=0,color="red") +
    labs(y="Residuals: linear age",x="Age in months") +
    scale_y_continuous(breaks=seq(-3.5,3.5,0.5),limits=c(-3.5,3.5)) +
  scale_x_continuous(breaks=seq(0,60,6),limits=c(0,60))
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```

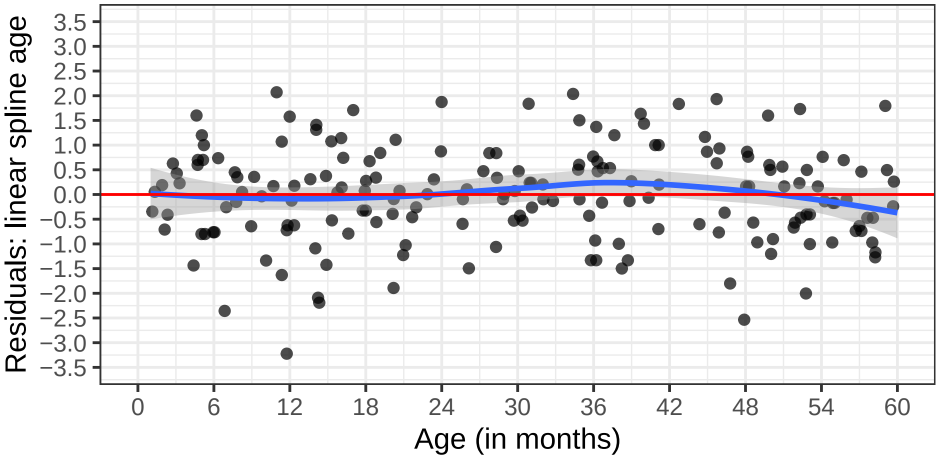


Update the model to include a smooth function of age via linear splines or natural splines.

```
reg1 = lm(arm~age + agesp6 + agesp12,data=d.cc)
reg2 = lm(arm~ns(age,3),data=d.cc)
d.cc$residuals1 = residuals(reg1)
d.cc$residuals2 = residuals(reg2)
```

```
ggplot(d.cc,aes(x=age, y=residuals1)) +
    geom_jitter(alpha = 0.7) +
    theme_bw() +
    geom_smooth() +
    geom_hline(yintercept=0,color="red") +
    labs(y="Residuals: linear spline age",x="Age (in months)") +
    scale_y_continuous(breaks=seq(-3.5,3.5,0.5),limits=c(-3.5,3.5)) +
  scale_x_continuous(breaks=seq(0,60,6),limits=c(0,60))
```

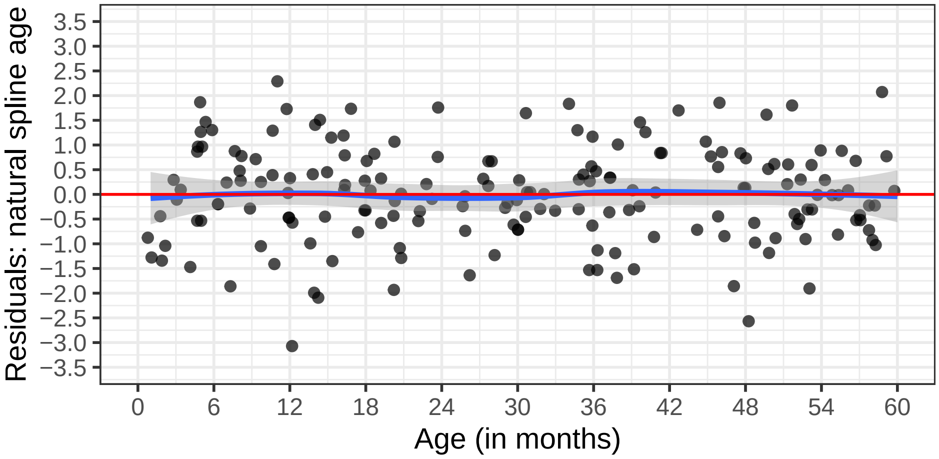
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



```
ggplot(d.cc,aes(x=age, y=residuals2)) +
    geom_jitter(alpha = 0.7) +
    geom_smooth() +
    theme_bw() +
    geom_hline(yintercept=0,color="red") +
    labs(y = "Residuals: natural spline age",x="Age (in months)") +
    scale_y_continuous(breaks=seq(-3.5,3.5,0.5),limits=c(-3.5,3.5)) +
  scale_x_continuous(breaks=seq(0,60,6),limits=c(0,60))
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



C. Independence assumption

The independence assumption is typically established based on the study / sampling design that generated the data. Most common examples of designs that violate the independence assumption are longitudinal studies and clustered studies.

1. Longitudinal study

In a longitudinal study, study participants are followed over time and the outcome is measured at predetermined time points (e.g. fixed design: at enrollment and every 3 months for a total of 2 years, or clinical cohort:at diagnosis and at each subsequent clinical appointment where the number of appointments and appointment times may vary).

In the longitudinal study, the data is expressed as (Y_{ij}, X_{ij}) where $i = 1, ..., m$ subjects and $j = 1, ..., n_i$ observations per subject. In such designs X_{ij} may contain variables describing characteristics of the subject (i.e. X_i) or characteristics of the subject at the specific assessment time (i.e. X_{ij}).

The data vector and covariate information for subject i within a longitudinal design is defined as $(Y_i,\ X_i)$ where Y_i is the $m \times 1$ vector of responses for subject i, and X_i is the $m \times p$ matrix of covariates for subject i.

2. Clustered study

In a clustered study, many participants are sampled from clusters. For example, all the children from the same household are included in a study OR a random sample of children from a selected school are enrolled in a study OR villages nested within a district are selected for inclusion. The clusters may carry important meaning (i.e. the context and characteristics of the cluster may influence the outcome of interest) or they may be utilized for convenience of reaching the target population of interest (i.e. it may be easier to identify children if we first recruit schools and then sample children within schools).

Similar to the longitudinal design, the data is expressed as (Y_{ij}, X_{ij}) where $i = 1, ..., m$ clusters and $j = 1, ..., n_i$ units per cluster. In such designs X_{ij} may contain variables describing characteristics of the cluster (i.e. X_i) or characteristics of the units within the cluster (i.e. X_{ij}).

Similar to the longitudinal design, in the cluster design the data for a cluster i is $(Y_i,\ X_i)$.

3. Why do we care?

Later, we will show that if your goal is solely to estimate $E(Y|X) = X\beta$ then you can obtain unbiased estimates of β by ignoring the design and using the least squares solution.

However, if you want to make inference about β, then you will need to appropriately account for the design in the analysis.

4. Example: Nepali Anthropometry data

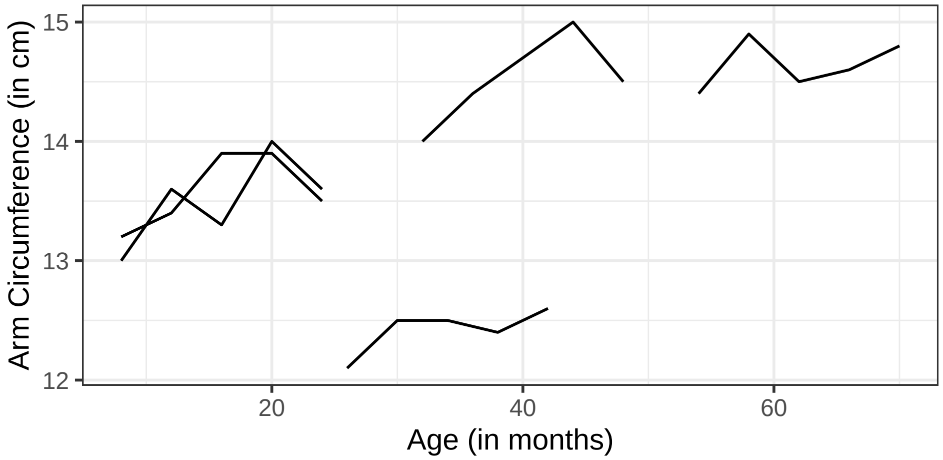
In the Nepali Anthopometry study, children between the ages of 0 to 60 months were recruited. Anthropometry data was collected for children at enrollment and then at 4 follow-up ups, once every 4 months.

```
load("NepalAnthro.rdata")
d5 = nepal.anthro[nepal.anthro$id %in% unique(nepal.anthro$id)[c(3,7,8,9,10)],c("id","arm","age","fuvisi
head(d5)
```

```
##        id  arm age fuvisit
## 11 120021 13.0   8       0
## 12 120021 13.6  12       1
## 13 120021 13.3  16       2
## 14 120021 14.0  20       3
## 15 120021 13.6  24       4
## 31 120051 13.2   8       0
```

The figure below displays the arm circumfernece as a function of age for 5 randomly selected children. What do you notice when looking at the data within a child over time? What do you notice when comparing the data across children?

```
ggplot(d5,aes(x=age,y=arm,group = factor(id))) +
    geom_line() +
    theme_bw() +
    labs(x='Age (in months)', y ='Arm Circumference (in cm)')
```



5. How do we check the independence assumption?

Consider the Nepali Anthropometry data where we have data for $i = 1, ..., m = 200$ children each measured at baseline ($j = 1$) and then every 4 months for 4 follow-up visits ($j = 2, 3, 4, 5$).

- Step 1: Regress Y on X assuming independence to estimate β and R.
- Step 2: Plot $\hat{R}_{ij}$ vs. $\hat{R}_{ik}$ for all j, k.
- Step 3: Compute $Cov(\hat{R}_{ij}, \hat{R}_{ik}) = \sqrt{Var(\hat{R}_{ij})} \times \sqrt{Var(\hat{R}_{ik})} \times Corr(\hat{R}_{ij}, \hat{R}_{ik})$

NOTE: You will learn how to do this in the lab 5.

6. How do we "fix" the model?

We have been working under the assumption that $Y = MVN(X\beta, \sigma^2 I)$ but now we recognize that $Var(Y) = \sigma^2 I$ is wrong.

We can now think of the data structure as:

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_m \end{bmatrix} \quad \text{MVN} \left(\begin{bmatrix} X_1\beta \\ X_2\beta \\ \vdots \\ X_m\beta \end{bmatrix}, \begin{bmatrix} V_1 & 0 & ... & 0 \\ 0 & V_2 & ... & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & ... & V_m \end{bmatrix} \right),$$

where Y_i and X_i are the vector of responses, and the design matrix for subject i, respectively, and V_i is the variance matrix that has $Var(Y_{ij})$ as the diagonal elements and $Cov(Y_{ij}, Y_{ik})$ on the off diagonal.

So that we have $Y \sim MVN(X\beta, \Sigma)$ where Σ is the block diagonal matrix above.

7. What if we use the least squares solution?

Assume the truth is: $Y \sim MVN(X\beta, \Sigma)$ and $\hat{\beta_{ls}} = (X'X)^{-1}X'Y$.

Is $\hat{\beta}_{ls}$ biased?

$$\begin{aligned} E(\hat{\beta_{ls}}) &= E\left[(X'X)^{-1}X'Y\right] \\ &= (X'X)^{-1}X'E[Y] \\ &= (X'X)^{-1}X'X\beta \\ &= \beta \end{aligned}$$

What is the $Var(\hat{\beta_{ls}})$?

$$\begin{aligned} Var(\hat{\beta}_{ls}) &= Var\left[(X'X)^{-1}X'Y\right] \\ &= \left[(X'X)^{-1}X'\right] Var(Y) \left[(X'X)^{-1}X'\right] \\ &= (X'X)^{-1}X'\Sigma X(X'X)^{-1} \end{aligned}$$

NOTE: The expression above will only be equal to $(X'X)^{-1}\sigma^2$ when $Var(Y) = \Sigma = \sigma^2 I$.

Implications:

- If you use the least squares estimate of variance then you are ignoring the information about the covariance structure in the data.
- Confidence intervals based on the least squares solution won't be reliable because the variance estimate is wrong. The confidence intervals can be either too wide or too narrow.

8. Weighted least squares

How about identifying a transformation of the data such that the transformed data satisfies our preferred assumptions?

Define $\Sigma^{-1/2}$ as a square weight matrix such that:

$$\Sigma^{-1/2}Y = \Sigma^{-1/2}X\beta + \Sigma^{-1/2}\epsilon \ , \ \Sigma^{-1/2}\epsilon \sim \text{ MVN}(0, I)$$

If we could define this weight matrix, then our solution for β is:

$$\hat{\beta}_{wls} = (X'\Sigma^{-1}X)^{-1}X'\Sigma^{-1}Y$$

$$Var(\hat{\beta}_{wls}) = (X'\Sigma^{-1}X)^{-1}$$

If $\Sigma = \begin{bmatrix} V_1 & 0 & ... & 0 \\ 0 & V_2 & ... & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & ... & V_m \end{bmatrix}$, then $\Sigma^{-1} = \begin{bmatrix} V_1^{-1} & 0 & ... & 0 \\ 0 & V_2^{-1} & ... & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & ... & V_m^{-1} \end{bmatrix}$.

This derives the weighted least squares estimator:

$$\hat{\beta}_{wls} = \left(\sum_{i=1}^{m} X_i' V_i^{-1} X_i \right)^{-1} \left(\sum_{i=1}^{m} X_i' V_i^{-1} Y_i \right)$$

where $V_i = Var(Y_i) = Var(\epsilon_i)$ can be estimated by $(Y_i - X_i\hat{\beta})(Y_i - X_i\hat{\beta})'$.

8. Robust variance estimate

Recall the least squares solution:

$$\hat{\beta}_{ls} = (X'X)^{-1}X'Y = \left(\sum_{i=1}^{m} X_i' X_i \right)^{-1} \left(\sum_{i=1}^{m} X_i' Y_i \right)$$

With variance given by:

$$\begin{aligned} Var(\hat{\beta}_{ls}) &= \left[(X'X)^{-1}X'\right] Var(Y) \left[(X'X)^{-1}X'\right]' \\ &= \left[(X'X)^{-1}\right] X'VX \left[(X'X)^{-1}\right] \\ &= \left(\textstyle\sum_{i=1}^{m} X_i'X_i\right)^{-1} \left(\textstyle\sum_{i=1}^{m} X_i'V_iX_i\right) \left(\textstyle\sum_{i=1}^{m} X_i'X_i\right)^{-1} \end{aligned}$$

Again, if we approximate $V_i = Var(Y_i) = Var(\epsilon_i)$ by $(Y_i - X_i\hat{\beta})(Y_i - X_i\hat{\beta})'$, then we have:

$$Var_{robust}(\hat{\beta}_{ls}) = \left(\sum_{i=1}^{m} X_i' X_i \right)^{-1} \left(\sum_{i=1}^{m} X_i'(Y_i - X_i\hat{\beta})(Y_i - X_i\hat{\beta})'X_i \right) \left(\sum_{i=1}^{m} X_i' X_i \right)^{-1}$$

This is the Huber-White estimator – sandwich estimator – "information sandwich" (Moulton).

D. Constant variance assumption

Now, let $Y_1, Y_2, \ldots, Y_n$ be independent observations from n units, but $Var(Y_i) = Var(\epsilon_i) = \sigma_i^2$. Then the regression model is given by:

$$Y = X\beta + \epsilon, \epsilon \sim MVN(0, \Sigma), \Sigma = \text{ diag}(\sigma_j^2, \text{ j = 1, ..., n})$$

Then apply the weighted least squares solution:

$$\begin{aligned} \hat{\beta}_{wls} &= (X'\Sigma^{-1}X')^{-1}X'\Sigma^{-1}Y \\ &= \left[(\Sigma^{-1/2}X)'(\Sigma^{-1/2}X)\right]^{-1} (\Sigma^{-1/2}X)'(\Sigma^{-1/2}Y) \end{aligned}$$

where $\Sigma^{-1/2} = \text{ diag}(1/\sigma_j, \text{ j = 1, ..., n})$.

$$\begin{bmatrix} 1/\sigma_1 & 0 & ... & 0 \\ 0 & 1/\sigma_2 & ... & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & ... & 1/\sigma_n \end{bmatrix} \begin{bmatrix} 1 & X_{11} & ... & X_{p1} \\ 1 & X_{12} & ... & X_{p2} \\ \vdots & \vdots & ... & \vdots \\ 1 & X_{1n} & ... & X_{pn} \end{bmatrix} = \begin{bmatrix} 1/\sigma_1 & X_{11}/\sigma_1 & ... & X_{p1}/\sigma_1 \\ 1/\sigma_2 & X_{12}/\sigma_2 & ... & X_{p2}/\sigma_2 \\ \vdots & \vdots & ... & \vdots \\ 1/\sigma_n & X_{1n}/\sigma_n & ... & X_{pn}/\sigma_n \end{bmatrix}$$

1. What if we use the least squares solution?

See Section C7; estimate of β will be unbiased but inference for β can be misleading!

2. One proposal for $\Sigma^{-1/2}$

What about estimating Σ by:

$$\hat{\Sigma}^{-1/2} = \begin{bmatrix} 1/\mid y_1 - X_1\hat{\beta} \mid & 0 & ... & 0 \\ 0 & 1/\mid y_2 - X_2\hat{\beta} \mid & ... & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & ... & 1/\mid y_n - X_n\hat{\beta} \mid \end{bmatrix}$$

This is not a great idea!

If $y_i \approx X_i\hat{\beta}$ then you are creating very large weights!

You want to smooth!

3. Build a model for $\Sigma^{-1/2}$

An alternative approach for estimating $\Sigma^{-1/2}$ is based on a model.

Regress $(y_i - X_i\hat{\beta})^2$ on X_i or $X_i\hat{\beta}$. Assume a smooth function, i.e. use natural splines and fit the model using a Gamma regression (with log link).

- NOTE: we will discuss this Gamma regression in more detail next term. Gamma random variables are positive and have tails to the right. It is common to assume variances follow a Gamma distribution.

You can show that:

$$\frac{(y_i - X_i\hat{\beta})^2}{\sigma_j} \sim \chi^2_1 = Gamma(\mu = 1, shape = 1/2)$$

Suppose you fit a model for $log(\sigma_j^2) = \gamma_0 + \gamma_1 Z_1 + ... + \gamma_p Z_l$ then $(y_i - X_i\hat{\beta})^2 \sim Gamma(\mu = e^{Z_i\hat{\gamma}}, shape = 1/2)$ and you can estimate σ_j^2 via $e^{Z_i\hat{\gamma}}$ and σ_j via $e^{Z_i\hat{\gamma}/2}$.

4. Two-step algorithm for fitting the weighted least squares solution

One option is to do a two-step estimation:

- Step 1: Fit the model for $E(Y|X) = X\beta$ using least squares
- Step 2: Obtain the residuals $\hat{r}_i = y_i - X_i\hat{\beta}$ and fit the regression of $\hat{r}_i^2$ on $ns(X_i\hat{\beta}, df = k)$ using a Gamma regression (log link) to estimate σ_i^2.
- Step 3: Compute $w_i = 1/\sqrt{\hat{\sigma}_i^2} = e^{-Z_i\hat{\gamma}/2}$ and fit the linear regression model using the weights.

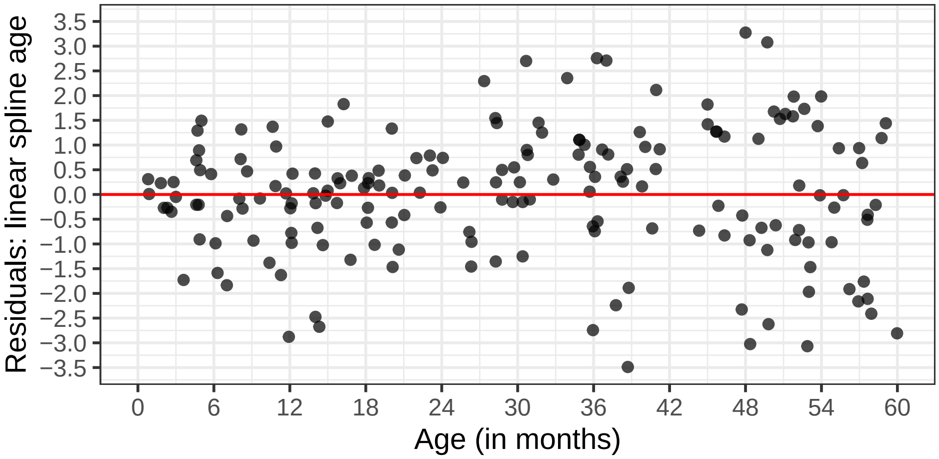
5. Example using two-step algorithm on Nepal data

Going back to the Nepali data but looking at the weight vs. age relationship.

```
load("NepalAnthro.rdata")
d = nepal.anthro %>% select(names(.)[1:16]) %>% filter(.,num==1)
d = mutate(d,
agesp6=ifelse(age-6>0, age-6,0)
)
cc=complete.cases(select(d,age,wt))
d.cc=filter(d,cc)
d.cc = arrange(d.cc,age)
reg<-lm(data=d.cc, wt~age+agesp6)
d.cc$residuals = residuals(reg)
```

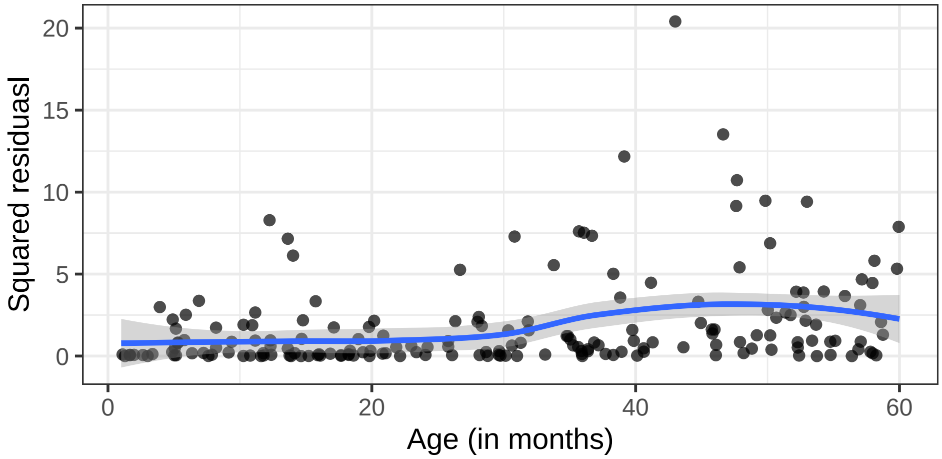
```
ggplot(d.cc,aes(x=age, y=residuals)) +
    geom_jitter(alpha = 0.7) +
    theme_bw() +
    geom_hline(yintercept=0,color="red") +
    labs(y="Residuals: linear spline age",x="Age (in months)") +
    scale_y_continuous(breaks=seq(-3.5,3.5,0.5),limits=c(-3.5,3.5)) +
  scale_x_continuous(breaks=seq(0,60,6),limits=c(0,60))
```

```
## Warning: Removed 3 rows containing missing values (geom_point).
```



```
d.cc = mutate(d.cc,r2 = residuals^2)
ggplot(d.cc,aes(x=age, y=r2)) +
    geom_jitter(alpha = 0.7) +
    theme_bw() +
    geom_smooth() +
    labs(y="Squared residuasl",x="Age (in months)")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Now, get the weighted least squares solution.

```
v=predict.glm(glm(r2 ~ ns(age,3),data=d.cc, family=Gamma(link="log")),type="response")
regw = lm(wt~age+agesp6,data=d.cc,weights=1/sqrt(v))
summary(reg)
```

```
##
## Call:
## lm(formula = wt ~ age + agesp6, data = d.cc)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.6768 -0.7575  0.0366  0.8998  4.5174
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.2112     0.7168   4.480 1.32e-05 ***
## age           0.5793     0.1303   4.445 1.53e-05 ***
## agesp6       -0.4307     0.1328  -3.243  0.00141 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.366 on 182 degrees of freedom
## Multiple R-squared:  0.8087, Adjusted R-squared:  0.8066
## F-statistic: 384.7 on 2 and 182 DF,  p-value: < 2.2e-16
```

```
summary(regw)
```

```
##
## Call:
## lm(formula = wt ~ age + agesp6, data = d.cc, weights = 1/sqrt(v))
##
## Weighted Residuals:
##     Min      1Q  Median      3Q     Max
## -2.9081 -0.6539  0.0860  0.7535  3.4893
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.2750     0.5904   5.547 1.01e-07 ***
## age           0.5554     0.1080   5.142 6.99e-07 ***
## agesp6       -0.4042     0.1104  -3.662 0.000328 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.154 on 182 degrees of freedom
## Multiple R-squared:  0.827,  Adjusted R-squared:  0.8251
## F-statistic:   435 on 2 and 182 DF,  p-value: < 2.2e-16
```

6. Iteratively re-weighted least squares

The general estimation procedure requires an interative process.

The algorithm is:

1. Regress Y on X to obtain $\hat{\beta}_{ls} = \hat{\beta}^{(0)}$.
2. Regress $\hat{r_i}^2 = (y_i - X_i\hat{\beta}^{(k)})^2$ on Z_i using a Gamma regression with log link. Obtain $\hat{\gamma}^{(k)}$.
3. Regress Y_i on X_i with weights $1/\hat{\sigma}_j^{(k)} = e^{-Z_i\hat{\gamma}^{(k)}/2}$.
4. Repeat steps 2. and 3. until convergence:

$$\frac{(\hat{\beta}^{(k+1)} - \hat{\beta}^{(k)})'(\hat{\beta}^{(k+1)} - \hat{\beta}^{(k)})}{\hat{\beta}^{(k)\prime}\hat{\beta}^{(k)}} << \delta$$

E. Normality assumption

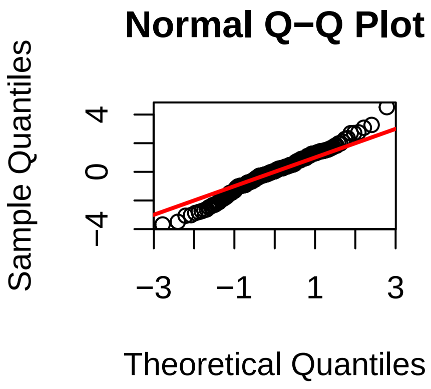
To assess the normality assumption of the residuals, you can make a quantile-quantile plot comparing the quantiles from the distribution of the estimated residuals to that of a standard normal random variable.

If you find an extreme departure from the straight line, then you can use a bootstrapping procedure to make inference.

1. Example using Nepali data

Back to the analysis of child weight vs. age.

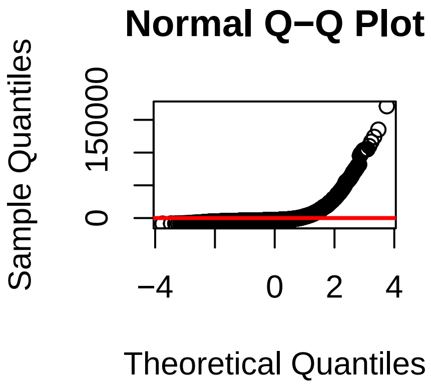
```r
qqnorm(d.cc$residuals)
abline(0,1,col="red",lwd=2)
```



The normality assumptions seems to be reasonable. There is some departure in the tails of the distribution of the residuals.

2. Example using medical expenditures

```
load("nmes.rdata")
d = nmes %>% select(names(.)[c(1,2,3,15)]) %>% filter(.,lastage>=65)
d = mutate(d,
agec=lastage-65,
agesp1 = ifelse(lastage-75>0, lastage-75,0),
agesp2 = ifelse(lastage-85>0, lastage-85,0)
)
reg = lm(totalexp~(agec+agesp1+agesp2)*male,data=d)
qqnorm(reg$residuals);abline(0,1,col="red",lwd=2)
```



As we expected, the residuals have a large departure from normality due to the skewed nature of the medical expenditures. In this case, inferences can be based on the bootstrap procedure.

F. Leverage and Influence

1. Leverage

Leverage is a measure of how far away an individual i's values of predictors (X_i) are from the center or average values of the predictors.

High-leverage points are those observations of X_i that are far from other Xs such that the lack of neighboring Xs means that the fitted regression model will pass close to that particular observation X_i.

Recall,

$$\hat{y}_i = \sum_{j=1}^{n} h_{ij} y_j$$

where h_{ii} is the weight given to y_i in predicting $\hat{y}_i$ at X_i.

2. Influence

Here we want to understand the impact of each observation on the fit of the model; i.e. how much do the regression statistics change if a given observation is removed?

There are several influence statistics that are used in practice:

- $DBETA_{ij} = \hat{\beta}_j - \hat{\beta}_{j(-i)}$
- $DBETAS_{ij} = \frac{DBETA_{ij}}{\hat{se}(\hat{\beta}_{j(-i)})}$
- $DFIT_i = \hat{Y}_i - \hat{Y}_{i(-i)}$
- $DFITS_i = \frac{DFIT_i}{\hat{se}(\hat{Y}_{i(-i)})}$

There are rules of thumb or cut-off values associated with each of these influence statistics; but generally, you should further explore observations that have large values of these influence statistics.

3. Example of Nepali Anthropometry data

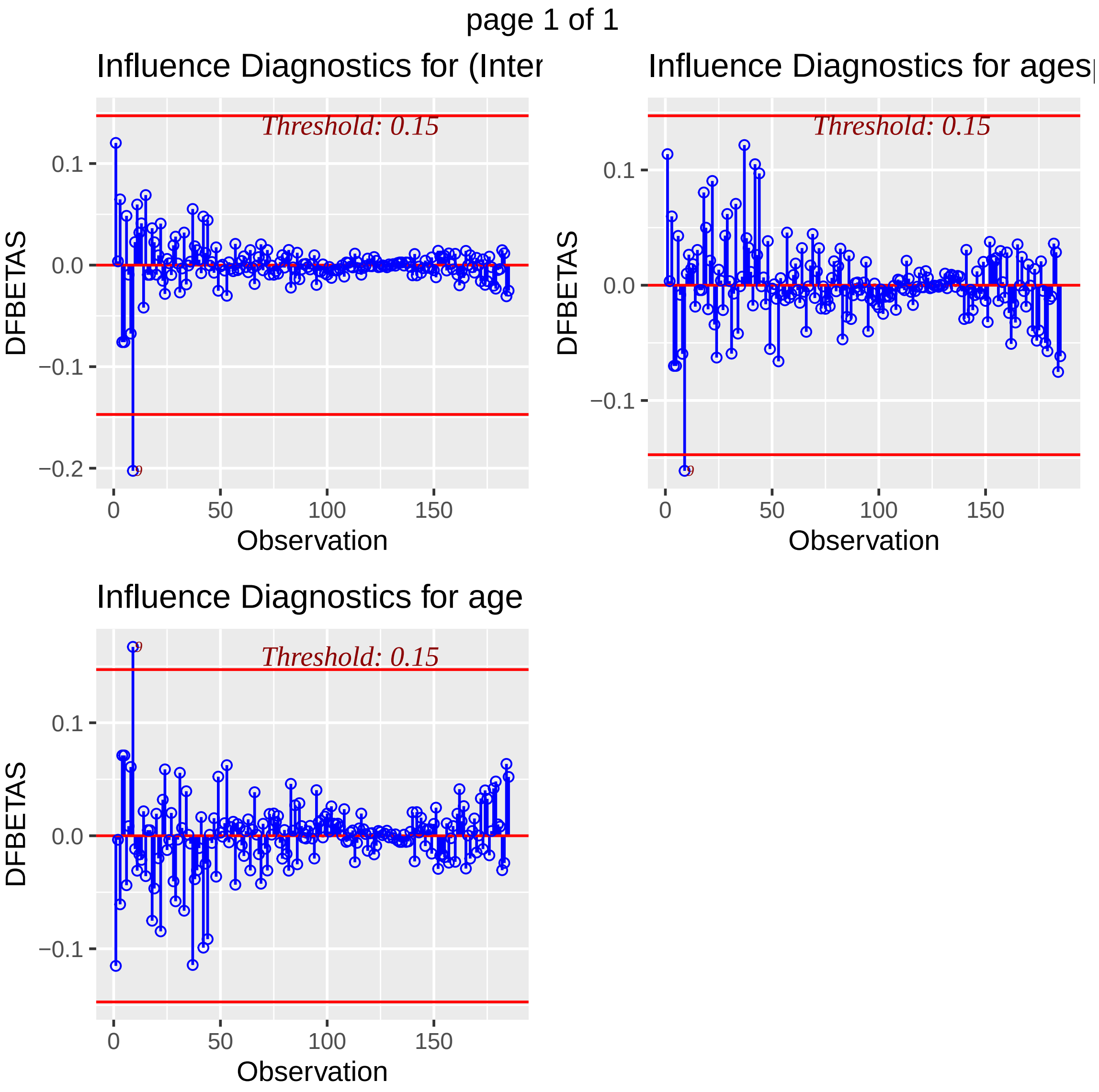
Back to the analysis of children's weight as a function of age.

Compute the influence statistics for the model:

$$wt_i = \beta_0 + \beta_1 age_i + \beta_2 (age_i - 6)^+ + \epsilon_i, \epsilon_i \sim N(0, \sigma^2)$$

```
d = nepal.anthro %>% select(names(.)[1:16]) %>% filter(.,num==1)
d = mutate(d,
agesp6=ifelse(age-6>0, age-6,0)
)
cc=complete.cases(select(d,age,wt))
d.cc=filter(d,cc)
d.cc = arrange(d.cc,age)
reg<-lm(data=d.cc,wt~age+agesp6)
```

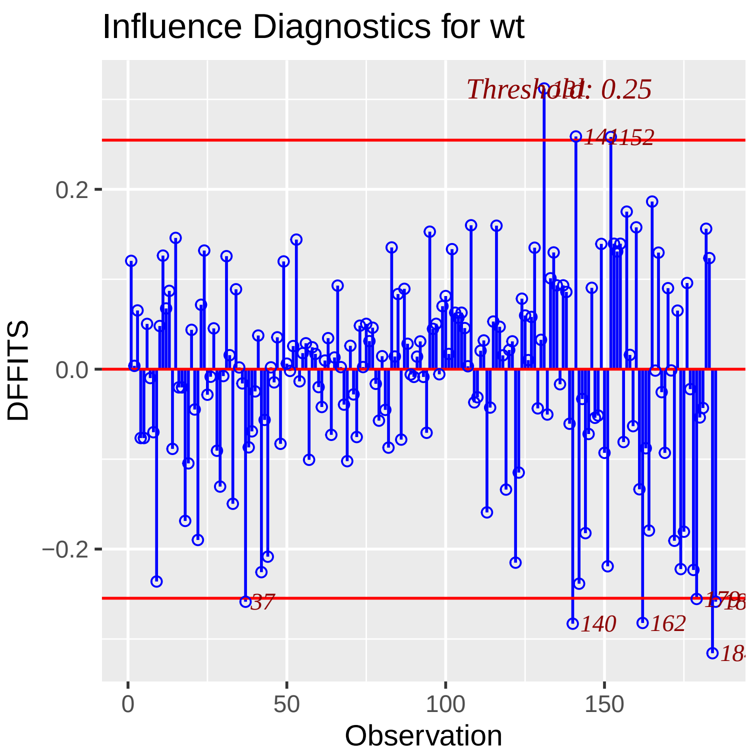
```
par(mfrow=c(2,2))
ols_plot_dfbetas(reg)
```



```
d.cc[9,c("wt","age")]
```

```
##    wt age
## 9 3.8   4
```

```
ols_plot_dffits(reg)
```



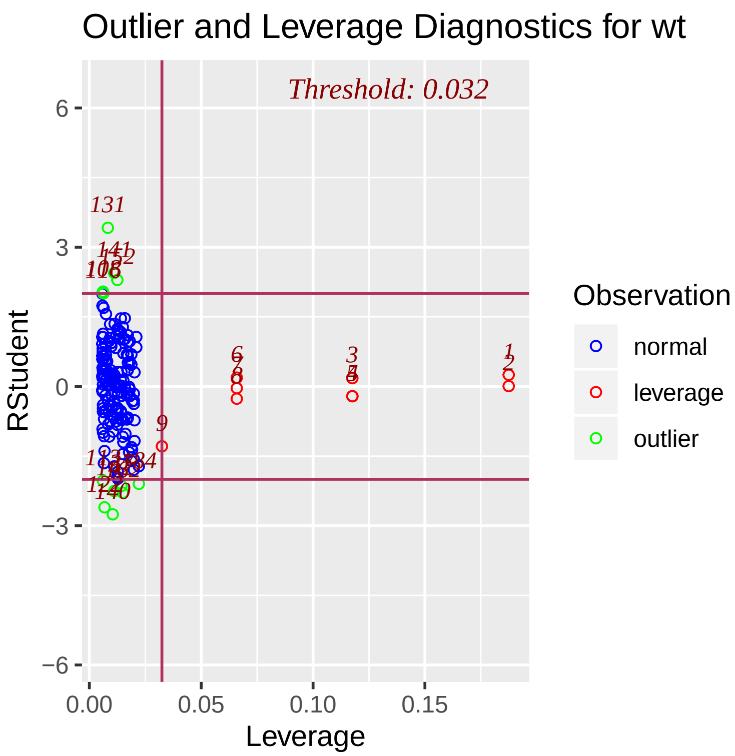
```r
d.cc[131,c("wt","age")]
```

```
##       wt age
## 131 16.7  43
```

```r
d.cc[180,c("wt","age")]
```

```
##       wt age
## 180 13.9  58
```

```r
ols_plot_resid_lev(reg)
```



```
d.cc[1:7,c("wt","age")]
```

```
##    wt age
## 1 4.1   1
## 2 3.8   1
## 3 4.6   2
## 4 4.1   2
## 5 4.1   2
## 6 5.2   3
## 7 4.9   3
```