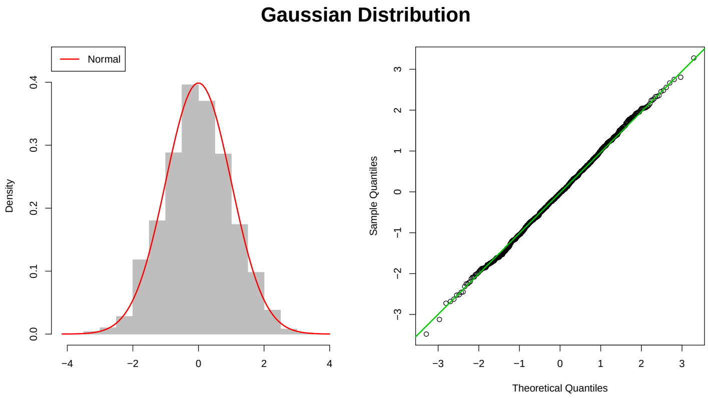
Chapter 4
Model Adequacy Checking

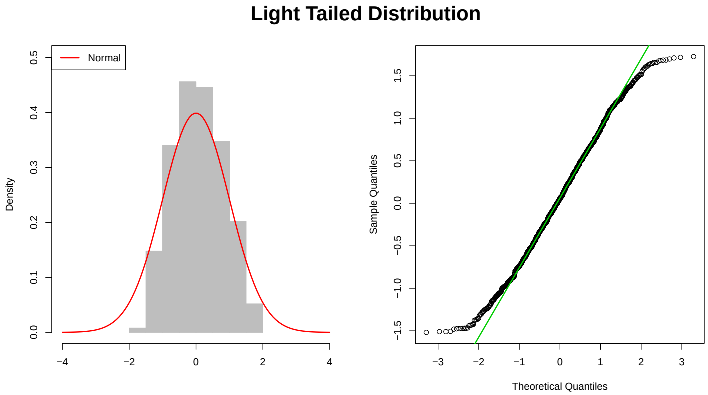
Patterns of Q-Q plot (Normal probability plot)

1. Gaussian distribution



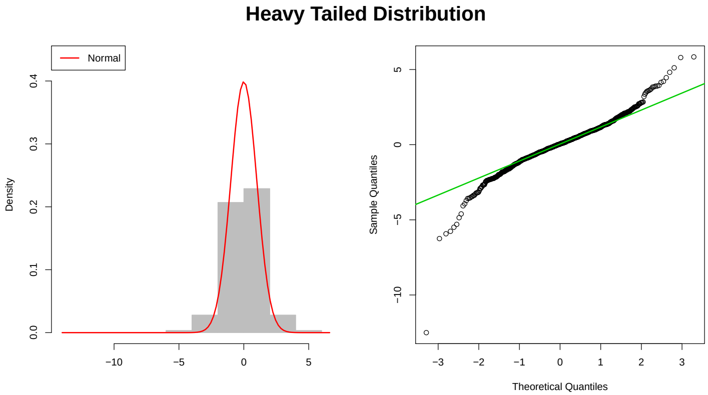
Patterns of Q-Q plot (Normal probability plot) (cont.)

2. Light tailed distribution



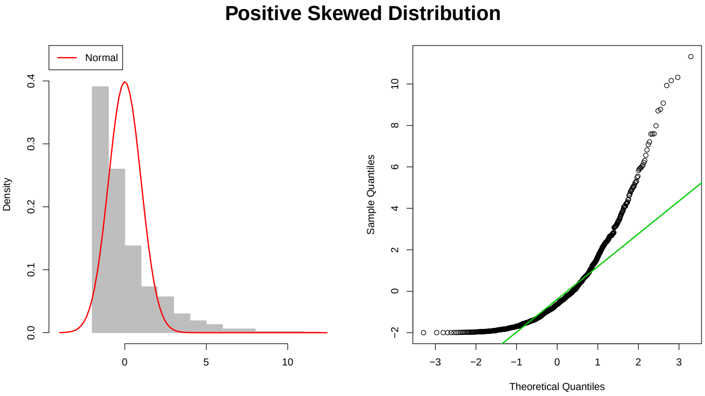
Patterns of Q-Q plot (Normal probability plot) (cont.)

3. Heavy tailed distribution



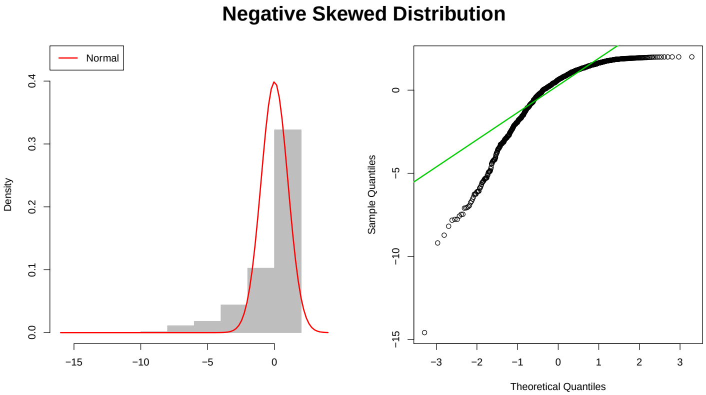
Patterns of Q-Q plot (Normal probability plot) (cont.)

4. Positive skewed distribution



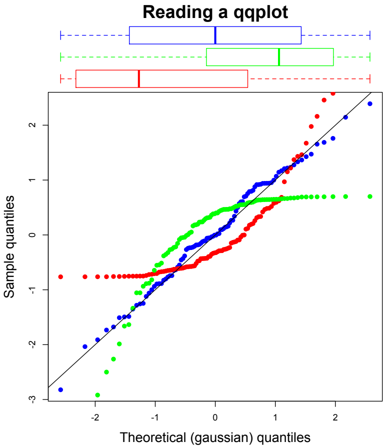
Patterns of Q-Q plot (Normal probability plot) (cont.)

5. Negative skewed distribution



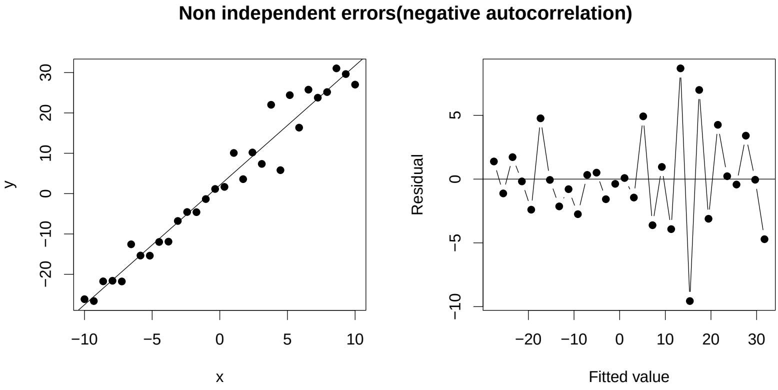
Patterns of Q-Q plot (Normal probability plot) (cont.)

6. Three types of Q-Q plot



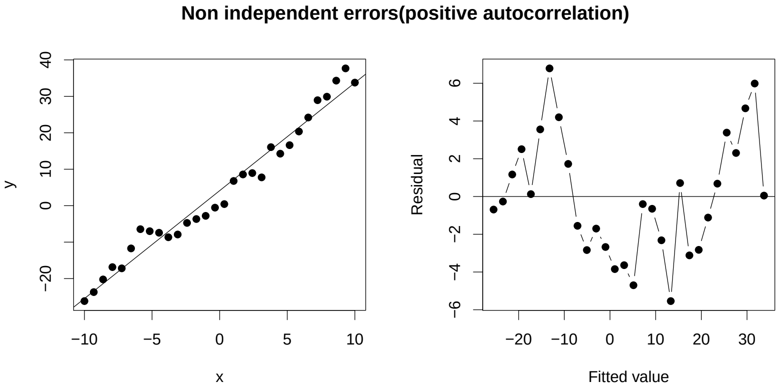
Patterns of residual plots

1. Non independent errors (negative autocorrelation)



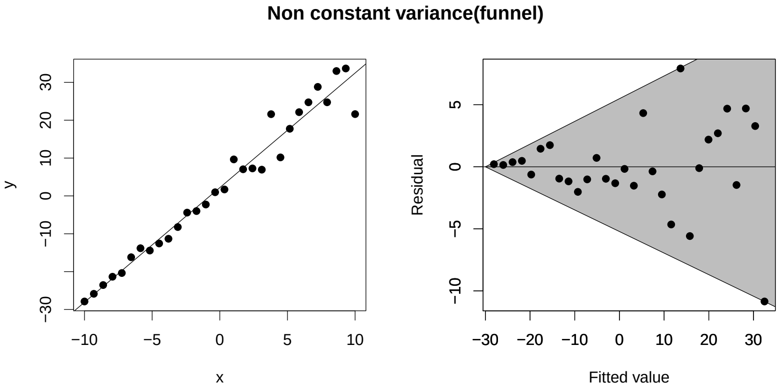
Patterns of residual plots (cont.)

2. Non independent errors (positive autocorrelation)



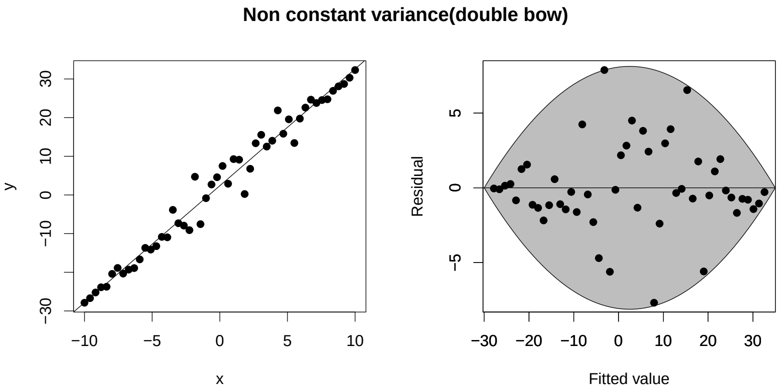
Patterns of residual plots (cont.)

3. Non constant variance (funnel)



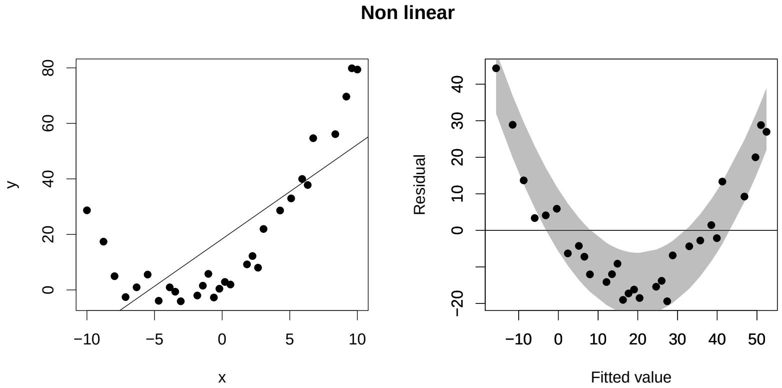
Patterns of residual plots (cont.)

4. Non constant variance (double bow)



Patterns of residual plots (cont.)

5. Non linear



Example 4.2 The Delivery Time Data

1. Various types of residuals

```
> url <- "https://raw.github.com/dongikjang/regression/master/"
> rfun <- getURL(paste(http, "scaled.R",sep=""))
> eval(parse(text=rfun))
>
> scaled
function(model, type="standardized")
  UseMethod("scaled")
>
> scaled.lm
function(model, type="standardized"){
  switch(type,
         studentized = rstandard(model),

         rstudent = rstudent(model),

         standardized = residuals(model)/summary(model)$sigma
  )
}
```

Example 4.2 The Delivery Time Data (cont.)

```
> # Data download
> rfun <- getURL(paste(http, "read.xls2.r",sep=""))
> eval(parse(text=rfun))
> # If OS is Windows then install "xlsReadWrite" package
> # If OS is Mac or Linux then install "gdata" package
>
> library(RCurl)
> tf <- paste(tempfile(), "xls", sep = ".")
> download.file(paste(url, "Dataset/data-ex-3-1.xls", sep=""), tf, met
  % Total    % Received % Xferd  Average Speed   Time    Time     Time
                                 Dload  Upload   Total   Spent    Left
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:
> data_3.1 <- read.xls2(tf, header=TRUE)
> View(data_3.1)
> colnames(data_3.1) <- c("obs", "d_time", "n_case", "dista")
```

Example 4.2 The Delivery Time Data (cont.)

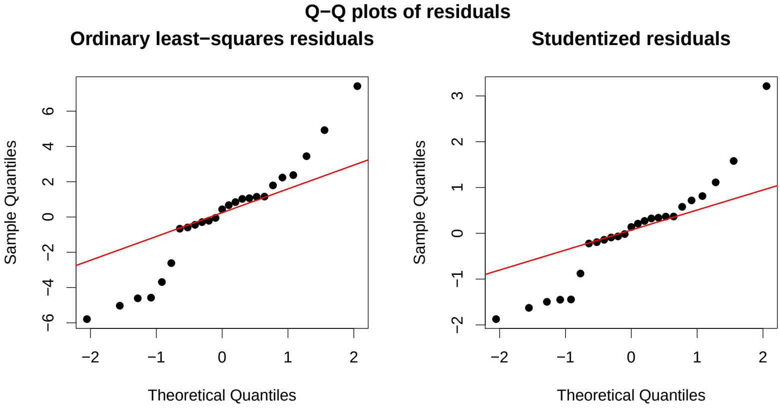
```
> # Linear fit
> lmfit <- lm(d_time~n_case+dista)
> # standardized residuals
> scaled(lmfit)
          1           2           3           4           5           6
-1.54260631  0.35170879 -0.01527661  1.51078203 -0.13634053 -0.0888408
          9          10          11          12          13          14
 2.27635117  0.72907878  0.68645843 -0.18194377  0.31508443  0.3275178
         17          18          19          20          21          22
 0.13387449  1.05803019  0.55014821 -1.77573772 -0.80202492 -1.1310194
         25
-0.06522033
```

Example 4.2 The Delivery Time Data (cont.)

```
> # standardized, studentized and  rstudentized residuals
> residual_mat <- cbind(residuals(lmfit), scaled(lmfit),
                        scaled(lmfit, "studentized"),
                        scaled(lmfit, "rstudent"))
> colnames(residual_mat) <- c("residual", "stadardized",
                              "studentized", "rstudent")
> head(residual_mat)
    residual stadardized studentized    rstudent
1 -5.0280843 -1.54260631 -1.62767993 -1.69562881
2  1.1463854  0.35170879  0.36484267  0.35753764
3 -0.0497937 -0.01527661 -0.01609165 -0.01572177
4  4.9243539  1.51078203  1.57972040  1.63916491
5 -0.4443983 -0.13634053 -0.14176094 -0.13856493
6 -0.2895743 -0.08884082 -0.09080847 -0.08873728
```

Example 4.2 The Delivery Time Data (cont.)

2. Q-Q plots of residuals

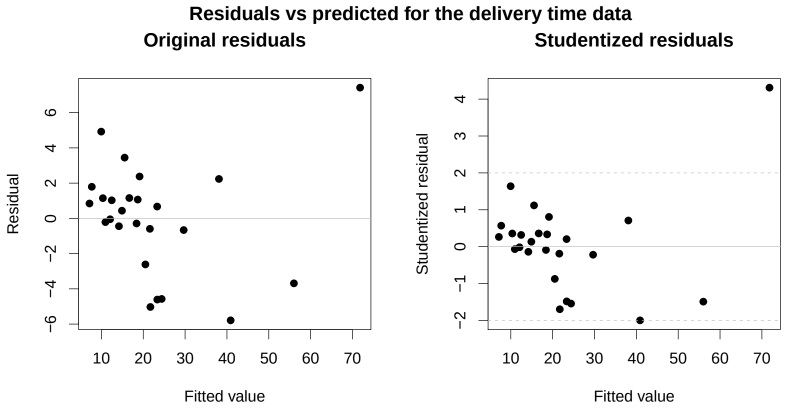


Example 4.2 The Delivery Time Data (cont.)

```
> par(mfrow=c(1,2), cex.main=1.2, pch=19, cex=1.5)
>
> qqnorm(residuals(lmfit), main='Ordinary least-squares residuals')
> qqline(residuals(lmfit), col=2, lwd=2)
>
> qqnorm(scaled(lmfit, "studentized"), main='Studentized residuals')
> qqline(scaled(lmfit, "studentized"), col=2, lwd=2)
>
> title(main='Q-Q plots of residuals',line=-1,outer=T)
```

Example 4.2 The Delivery Time Data (cont.)

2. Residuals vs Predicted

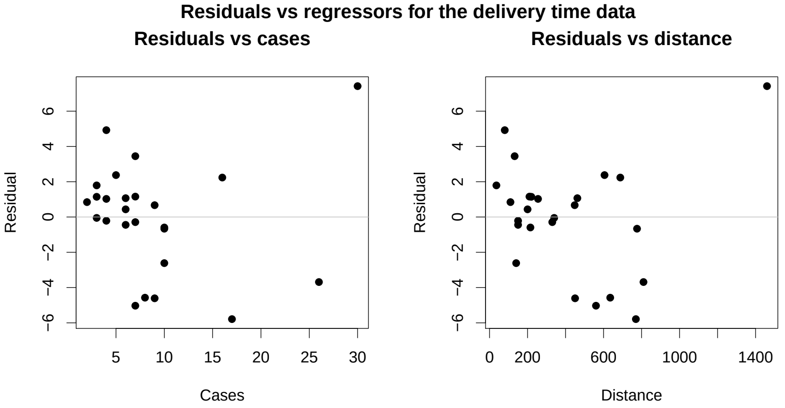


Example 4.2 The Delivery Time Data (cont.)

```
> par(mfrow=c(1,2), cex.main=1.2, pch=19, cex=1.5)
>
> fit_val <- fitted(lmfit)
>
> plot(fit_val, residuals(lmfit), xlab="Fitted value",
+       ylab="Residual", main="Original residuals")
> abline(h=0, lty=1, col="grey")
>
> plot(fit_val, scaled(lmfit, "rstudent"), xlab="Fitted value",
+      ylab="Studentized residual", main="Studentized residuals")
> abline(h=c(0,-2,2), lty=c(1,2,2), col="grey")
>
> title(main='Residuals vs predicted for the delivery time data',
+       line=-1,outer=T)
```

Example 4.2 The Delivery Time Data (cont.)

3. Residuals vs Regressors



Example 4.2 The Delivery Time Data (cont.)

```
> par(mfrow=c(1,2), cex.main=1.2, pch=19, cex=1.5)
>
> fit_val <- fitted(lmfit)
>
> plot(n_case, residuals(lmfit), xlab="Cases",
+      ylab="Residual", main="Residuals vs cases")
> abline(h=0, lty=1, col="grey")
>
> plot(dista, residuals(lmfit), xlab="Distance",
+      ylab="Residual", main="Residuals vs distance")
> abline(h=0, lty=1, col="grey")
>
> title(main='Residuals vs regressors for the delivery time data',
+       line=-1,outer=T)
```

Example 4.2 The Delivery Time Data (cont.)

4. Partial regression plots
   - Model:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon$$

   - Partial residual 1:

$$\begin{aligned} \hat{y}_i(x_2) &= \hat{\theta}_0 + \hat{\theta}_1 x_{i2} \\ e_i(y|x_2) &= y_i - \hat{y}_i(x_2), \quad i = 1, 2, \ldots, n \end{aligned}$$

   - Partial regressor 2:

$$\begin{aligned} \hat{x}_{i1}(x_2) &= \hat{\alpha}_0 + \hat{\alpha}_1 x_{i2} \\ e_i(x_1|x_2) &= x_{i1} - \hat{x}_{i1}(x_2), \quad i = 1, 2, \ldots, n \end{aligned}$$

   - Partial regression plots: plotting $e_i(y|x_2)$ against $e_i(x_1|x_2)$.

Example 4.2 The Delivery Time Data (cont.)

Example 4.2 The Delivery Time Data (cont.)

```
> par(mfrow=c(1,2), cex.main=1.2, pch=19, cex=1.5)
>
> plot(lm(n_case~dista)$resi,lm(d_time~dista)$resi,xlab='Cases',
+      ylab='Time', main='Time vs Cases', pch=16, cex=1.3)
>
> plot(lm(dista~n_case)$resi,lm(d_time~n_case)$resi,xlab='Distance',
       ylab='Time', main='Time vs Distance', pch=16, cex=1.3)
>
> title(main='Partial regression plots for the delivery time data',
        line=-1,outer=T)
```

Example 4.2 The Delivery Time Data (cont.)

5. Partial residual plots
   - The partial residual for regressor x_j:

$$e_i^*(y|x_j) = e_i + \hat{\beta}_j x_{ij}, \quad i = 1, 2, \ldots, n$$

     where the e_i are the residuals from the model with all k regressors included.
   - Partial residual plots: plotting $e_i^*(y|x_j)$ against x_j.

Example 4.2 The Delivery Time Data (cont.)

Example 4.2 The Delivery Time Data (cont.)

```
> partial <- function(model, part)
+ UseMethod("partial")
>
> partial <- function(model, part){
+         x <- model$model[, part]
+         coeff <- model$coefficients[part]
+         resi <- c(residuals(model) + x*coeff)
+         return(resi)
+ }
>
> par(mfrow=c(1,2), cex.main=1.2, pch=19, cex=1.5)
> plot(n_case, partial(lmfit, "n_case"), pch=16,cex=1.3,
+      xlab='Cases',ylab='Time',main='Time vs Cases')
> plot(dista, partial(lmfit, "dista"), pch=16, cex=1.3,
+ xlab='Distance',ylab='Time',main='Time vs Distance')
> title(main='Partial residual plots for the delivery time data',
+       line=-1,outer=T)
```

Example 4.2 The Delivery Time Data (cont.)

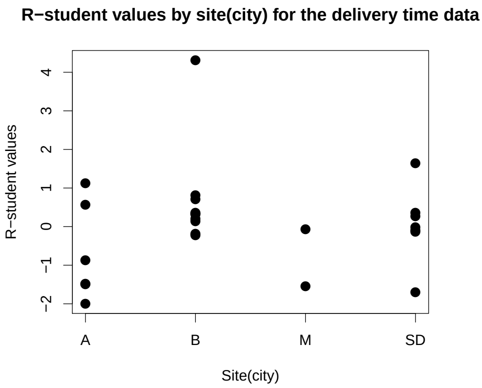
6. Regressor vs Regressor

Example 4.2 The Delivery Time Data (cont.)

```
> par(mfrow=c(1,1), pch=16,cex=1.4)

> plot(dista,n_case,xlab="Distance",ylab="Cases",
+      pch=16,main="Cases vs Distance")

> identify(dista ,n_case, 1:length(n_case))

> title(main="Regressor vs regressor for the delivery time data",
+       line=-1,outer=T)
```

Example 4.2 The Delivery Time Data (cont.)

7. R-student values by site(city)



Example 4.2 The Delivery Time Data (cont.)

```
> par(mfrow=c(1,1), pch=16, cex=1.4)
>
> i <- 1:length(d_time)
> site <- ifelse(i<=7, "SD", ifelse(i<=17, "B",
+                                    ifelse(i<=23, "A", "M")))
> site <- factor(site)
> stripchart(scaled(lmfit, "rstudent")~site, pch=16, vertical=T,
+            cex=1.5, xlab="Site(city)", ylab="R-student values")
>
> title(main="R-student values by site(city)
+ for the delivery time data")
```

Example 4.2 The Delivery Time Data (cont.)

8. PRESS statistics

$$PRESS = \sum_{i=1}^{n}[y_i - \hat{y}_{(i)}]^2 = \sum_{i=1}^{n}\left(\frac{e_i}{1-h_{ii}}\right)^2$$

```
> press <- function(obj){
+     sum((resid(obj)/(1-hatvalues(obj)))^2)
}
```

Example 4.2 The Delivery Time Data (cont.)

- R^2 for prediction based on PRESS

$$R^2_{prediction} = \frac{1 - PRESS}{SS_T}$$

```
> 1-press(lmfit)/sum((d_time-mean(d_time))^2)
[1] 0.9206438
```

- Using PRESS to compare Models

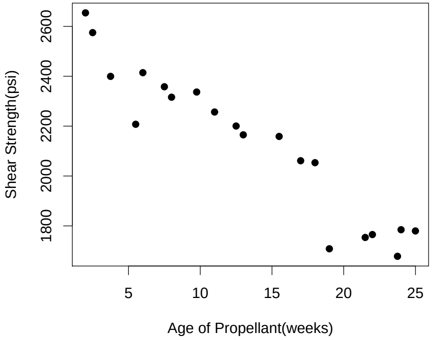
```
> press(lm(d_time ~ n_case))
[1] 733.55
> press(lm(d_time ~ n_case + dista))
[1] 459.0393
```

Example 4.7 The Rocket Propellant Data

1. Data and Plots

obs	yi	xi
1	2158.70	15.50
2	1678.15	23.75
3	2316.00	8.00
4	2061.30	17.00
5	2207.50	5.50
⋮	⋮	⋮
19	2654.20	2.00
20	1753.70	21.50

Example 4.7 The Rocket Propellant Data (cont.)

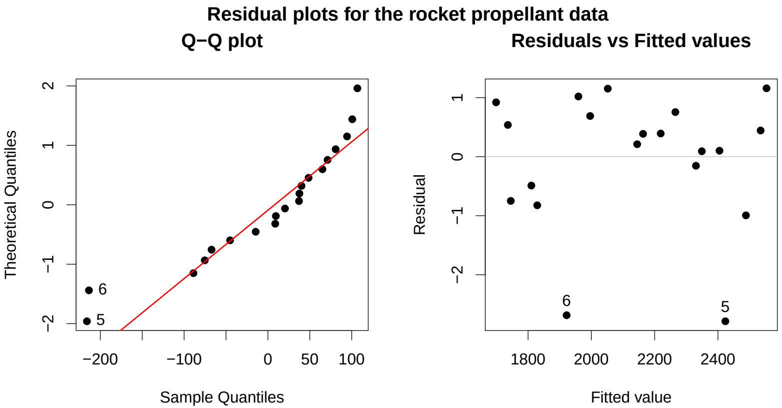


Example 4.7 The Rocket Propellant Data (cont.)

```
> tf <- paste(tempfile(), "xls", sep = ".")
> download.file(paste(url, "Dataset/data-ex-2-1.xls", sep=""), tf, met
  % Total    % Received % Xferd  Average Speed   Time    Time     Time
                                 Dload  Upload   Total   Spent    Left
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:
> data_2.1 <- read.xls2(tf, header=TRUE)
> colnames(data_2.1) <- c("obs", "yi", "xi")
> attach(data_2.1)
>
> par(mfrow=c(1,1), pch=16, cex=1.4)
> plot(xi, yi, pch=19, xlab="Age of Propellant(weeks)",
+      ylab="Shear Strength(psi)")
```

Example 4.7 The Rocket Propellant Data (cont.)

2. Detection and treatment of outliers

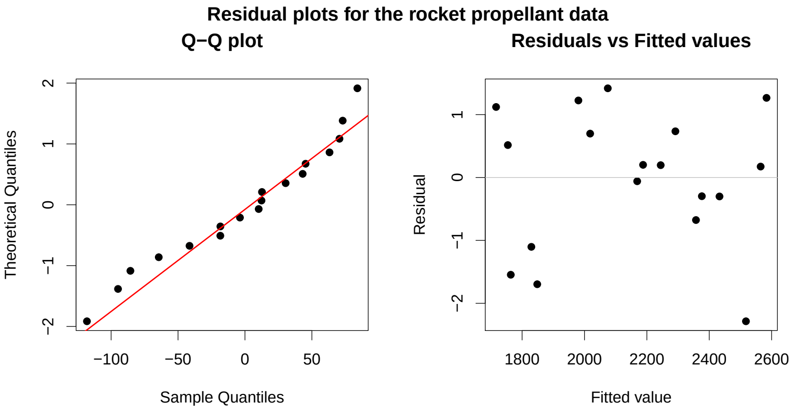


Example 4.7 The Rocket Propellant Data (cont.)

```
> lmfit <- lm(yi~xi)

> par(mfrow=c(1,2), cex.main=1.2, pch=19, cex=1.5)

> qqnorm(residuals(lmfit), datax=TRUE, main="Q-Q plot")
> qqline(residuals(lmfit), datax=TRUE, col=2, lwd=2)
> identify(sort(residuals(lmfit)), qnorm(1:length(xi)/length(xi)),
+          (1:length(xi))[order(residuals(lmfit))])

> fit_val <- fitted(lmfit)
> plot(fit_val, scaled(lmfit, "rstudent"), xlab="Fitted value",
+      ylab="Residual", main="Residuals vs Fitted values")
> identify(fit_val, scaled(lmfit, "rstudent"), 1:length(xi))
> abline(h=0, lty=1, col="grey")

> title(main="Residual plots for the rocket propellant data",
+       line=-1,outer=T)
```

Example 4.7 The Rocket Propellant Data (cont.)

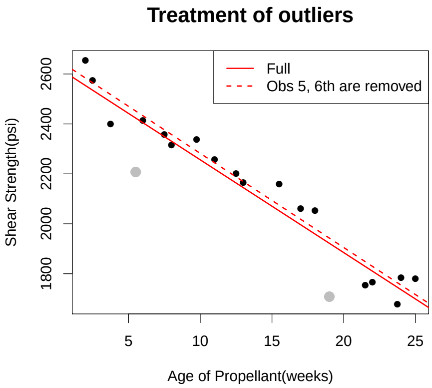
3. Observations 5 and 6 are removed



Example 4.7 The Rocket Propellant Data (cont.)

```
> lmfit <- lm(yi[-c(5,6)]~xi[-c(5,6)])

> par(mfrow=c(1,2), cex.main=1.2, pch=19, cex=1.5)

> qqnorm(residuals(lmfit), datax=TRUE, main="Q-Q plot")
> qqline(residuals(lmfit), datax=TRUE, col=2, lwd=2)
> identify(sort(residuals(lmfit)), qnorm(1:length(xi)/length(xi)),
+          (1:length(xi))[order(residuals(lmfit))])
> fit_val <- fitted(lmfit)

> plot(fit_val, scaled(lmfit, "rstudent"), xlab="Fitted value",
+      ylab="Residual", main="Residuals vs Fitted values")
> abline(h=0, lty=1, col="grey")

> title(main="Residual plots for the rocket propellant data",
+       line=-1,outer=T)
```

Example 4.7 The Rocket Propellant Data (cont.)



Example 4.7 The Rocket Propellant Data (cont.)

```
> par(mfrow=c(1,1), pch=16, cex=1.4)

> lmfit <- lm(yi~xi)

> plot(xi, yi, xlab="Age of Propellant(weeks)",
+      ylab="Shear Strength(psi)")
> abline(lmfit, col=2, lwd=2)
> points(xi[5:6], yi[5:6], col="grey", cex=1.5, pch=19)

> lmfit <- lm(yi[-c(5,6)]~xi[-c(5,6)])
> abline(lmfit, col=2, lwd=2, lty=2)

> legend("topright", legend=c("Full", "Obs 5, 6th are removed"),
+        col=2, lty=1:2, lwd=2)
```

Lack of Fit of the Regression Model

$$\begin{aligned}\sum_{i=1}^{m}\sum_{j=1}^{n_i}(y_{ij}-\hat{y}_i)^2 &= \sum_{i=1}^{m}\sum_{j=1}^{n_i}(y_{ij}-\bar{y}_i)^2+\sum_{i=1}^{m}\sum_{j=1}^{n_i}(\bar{y}_i-\hat{y}_i)^2 \\ SS_{RES} &= SS_{PE} + SS_{LOF}\end{aligned}$$

$$F_0=\frac{SS_{LOF}/(m-2)}{SS_{PE}/(n-m)}=\frac{MS_{LOF}}{MS_{PE}}\sim F_{(m-2,n-m)}$$

Lack of Fit of the Regression Model (cont.)

```
SSpe <- function(model, lof){  #SSpe function
        lmfit <- lm(model)
        y <- model.response(lmfit$model)
        x <- factor(lof)
        SSpe <- sum(xtabs(y^2~x)-xtabs(y~x)^2/table(x))
        SSres <- sum(residuals(lmfit)^2)
        SSlof <- SSres- SSpe
        out <- matrix(NA, 3, 5)
        colnames(out) <- c("Sum Sq", "Df", "Mean Sq", "F value", "Pr(>F)")
        rownames(out) <- c("SSlof", "SSpe", "SSres")
        out[,1] <- c(SSlof, SSpe, SSres)
        out[,2] <- c(length(levels(x))-2, length(x)-length(levels(x)),
                     length(x)-2)
        out[1:2,3] <- out[1:2,1]/out[1:2,2]
        out[1,4] <- out[1,3]/out[2,3]
        out[1,5] <- pf(out[1,4], out[1,2], out[2,2], lower.tail=F)
        printCoefmat(out, digits=4, na.print="")
}
```

Lack of Fit of the Regression Model (cont.)

1. Using SSpe function

```
> x <- c(1,1,2,3.3,3.3,4,4,4,4.7,5,5.6,5.6,5.6,6,6,6.5,6.9)
> y <- c(10.84,9.30,16.35,22.88,24.35,24.56,25.86,29.16,24.59,
         22.25,25.90,27.2,25.61,25.45,26.56,21.03,21.46)
>
> SSpe(y~x, x)
       Sum Sq      Df Mean Sq F value  Pr(>F)
SSlof 234.571   8.000  29.321   13.19 0.00139 **
SSpe   15.563   7.000   2.223
SSres 250.134  15.000
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
>
```

Lack of Fit of the Regression Model (cont.)

2. Using anova function (restricted method)

```
> f1 <- lm(y ~ x)
> f2 <- lm( y ~ factor( x ) )
> anova( f1, f2 )
Analysis of Variance Table

Model 1: y ~ x
Model 2: y ~ factor(x)
  Res.Df     RSS Df Sum of Sq      F   Pr(>F)
1     15 250.134
2      7  15.563  8    234.57 13.188 0.001389 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```