

Lecture 13: Regression Diagnostics II

Ailin Zhang

2023-06-13

Week 6 - 8 Plans

- Week 6: Lec 13, 14 (in person); Lec 15 (online)
- Week 7 (Dragon Boat Festival): Lec 16 (+quiz), Lec 17 (Scope of Midterm)
- Week 8: Tue (RC by TA), Lec 18 (online), Midterm on Friday
- Distribute project guidelines after the midterm

Recap: Regression Diagnostics

- Regression diagnostics are used after fitting to check if a fitted mean function ($E(Y|X)$) and assumptions are consistent with observed data.
- To fit a model with least square estimates and perform statistical analysis rely on the regression assumptions:
 - Assumptions about the model form
 - The relationship between the response (Y) and the predictors (X_1, \dots, X_p) is linear
 - Assumptions about the errors
 - The errors are independent, with mean zero and constant variance. The errors can be normally distributed (optional)
 - Assumptions about the predictors
 - The predictors are non-random, measured without error, and linearly independent
 - Assumptions about the observations
 - All observations are equally reliable and have an approximately equal role in determining the regression results and in influencing conclusions

Recap: Leverage

$\hat{Y} = X\hat{\beta} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} = \mathbf{H}\mathbf{Y}$ (where \mathbf{H} is called the hat matrix or the projection matrix)

- Recall in SLR, we can solve \mathbf{H} analytically:

$$h_{ij} = \frac{1}{n} + \frac{(x_i - \bar{x})(x_j - \bar{x})}{\sum_{k=1}^n (x_k - \bar{x})^2}$$

And the leverage in SLR is given by

$$h_{ii} = \frac{1}{n} + \frac{(x_i - \bar{x})^2}{\sum_{k=1}^n (x_k - \bar{x})^2}$$

Recap: Leverage

- If the leverage of i -th observation, h_{ii} , is large (close to 1), then this i -th observation is called a leverage point. It means the prediction of \hat{y}_i depends a lot on the observation y_i itself and relatively less on other observations.

Other properties:

- 1 $\frac{1}{n} \leq h_{ii} \leq 1$
- 2 $\sum h_{ii} = p + 1$
- 3 Thus, on average, $h_{ii} \approx (p + 1)/n$

We can look for values far from this as rough screen for high leverage points.

Recap: Three Types of Residuals

- The raw residual of the i -th observation is defined to be

$$e_i = y_i - \hat{y}_i = \text{observed } y_i - \text{predicted } y_i$$

We call this as (raw) residuals.

- Standardized residual or internally studentized residuals:

$$r_i = \frac{e_i}{\hat{\sigma} \sqrt{1 - h_{ii}}}$$

- Externally studentized residuals or studentized residuals are defined as:

$$r_i^* = \frac{e_i}{\hat{\sigma}_{(i)} \sqrt{1 - h_{ii}}}$$

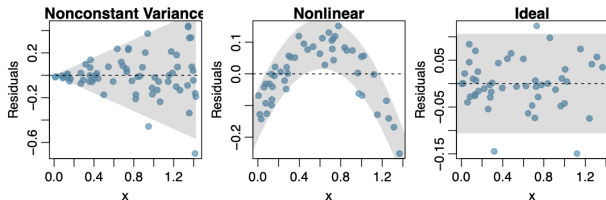
Recap: Comparisons of 3 Types of Residuals

- e_i 's add up to 0, r_i 's and r_i^* 's do not add up to 0
- e_i 's have unequal variance, r_i 's and r_i^* 's have variance 1
- r_i^* 's has a t-distribution with $df=n-p-2$, but r_i does not have a t-distribution.
- With a large enough sample, r_i 's and r_i^* 's are approximately $N(0, 1)$
- None of the 3 types of residuals are strictly independent, but the dependence can be ignored with large enough samples

Recap: Various Kinds of Residual Plots

- Residuals v.s. fitted values
- Residuals v.s. each predictor
- Residuals v.s. potential predictors not yet included in the model
- Residuals v.s. several predictors using `ggplot()`
- Residuals v.s. time if the data are collected over time
- Residuals v.s. ...

In all the plots above, points should scatter evenly above and below the zero line in a band of constant width.



Agenda

- Revisit Regression Assumptions
- Leverage
- Types of Residuals
- Residual Plots
- Graphical Methods (Today)
- Influential Points and Outliers
- Measure of Influence

Checking Assumptions with Graphs

Graphs are useful to:

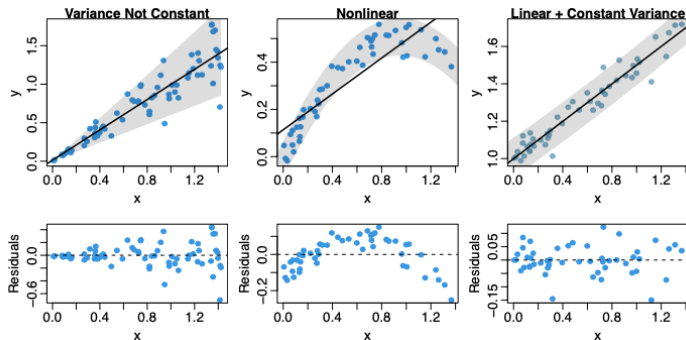
- Detect errors in the data
- Recognize patterns in the data (e.g., clusters, outliers, gaps, etc.)
- Explore relationships among variables
- Discover new phenomena
- Confirm or negate assumptions
- Assess the adequacy of a fitted model
- Suggest remedial actions (e.g., transform the data, redesign the experiment, collect more data, etc.)
- Enhance numerical analysis in general

Checking Assumptions for SLR Using Graphs

For SLR, one can plot

Y against X, Residuals against X, or Residuals against Y

and spot problems (nonlinearity, nonconstant variability, outliers)



Checking Assumptions for MLR Using Graphs

For MLR, it is more difficult to check whether the linear form $Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$ is correctly specified.

We need to check the following:

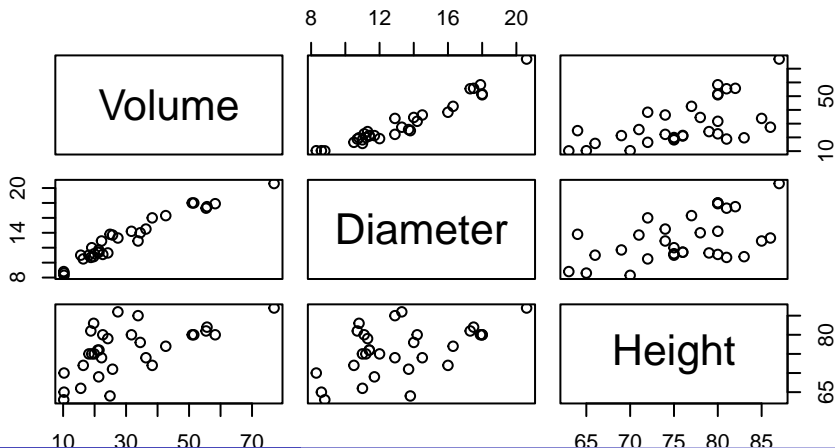
- Is Y linear X_j given other X_k 's?
- Do we miss any important predictor, any interactions?
- Does each X_i have linear effect on Y given other predictors?

Tools:

- Pairwise scatterplots = Scatterplot Matrix
- Multi-panel scatterplots using facet feature in ggplot
- Plotting residuals against each predictors and potential predictors not in the model
- Normal probability plot
- Added variable plots
- Residual plus component plots

Pairwise Scatterplots

```
data("trees")  
colnames(trees) = c("Diameter", "Height", "Volume")  
pairs(Volume ~ Diameter + Height, data=trees)
```

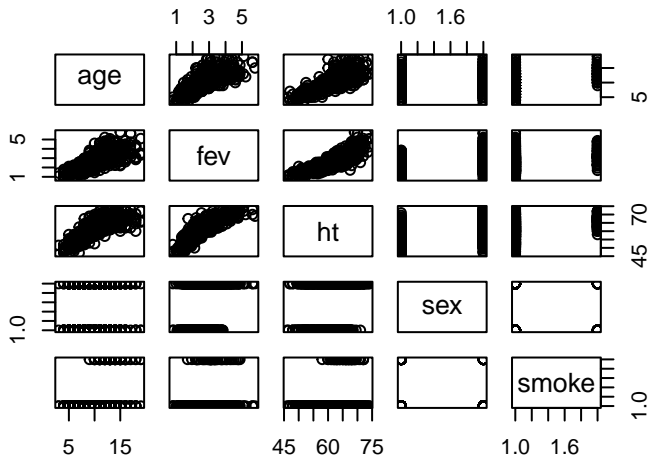


Pairwise Scatterplots

- Explore the relationships between each pair of variables and to identify general patterns
- Pairwise scatterplots only allow us to inspect the relations between variables pairwise, but not 3 or more variables at the same time.
- When there are many predictors, people usually pick the ones with large correlation between the response to begin with.

Pairwise Scatterplots?

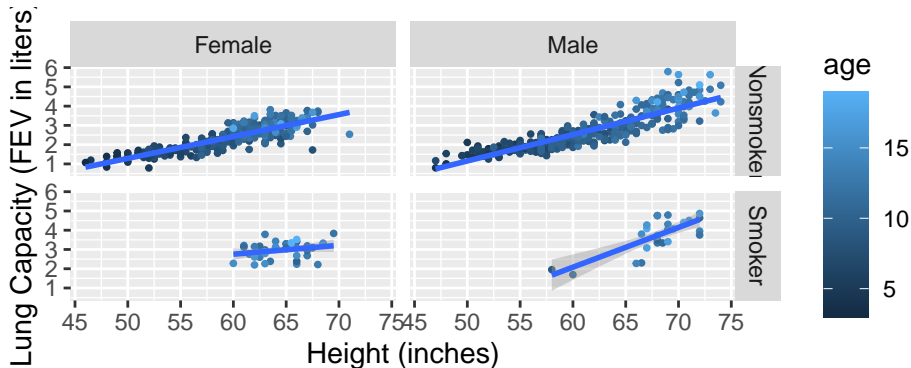
```
fevdata = read.table("fevdata.txt", header=TRUE)
fevdata$sex = factor(fevdata$sex, labels=c("Female", "Male"))
fevdata$smoke = factor(fevdata$smoke, labels=c("Nonsmoker", "Smoker"))
pairs(fevdata, oma = c(2,2,2,2))
```



Multi-panel scatterplots

```
library(ggplot2)
ggplot(fevdata, aes(x = ht, y = fev, color=age)) +
  geom_point(cex=0.7) + facet_grid(smoke~sex) + geom_smooth(method='lm') +
  labs(x="Height (inches)", y="Lung Capacity (FEV in liters)")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

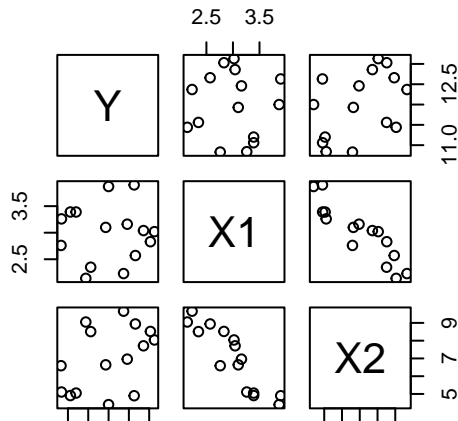


Multi-panel scatterplots with ggplot

- ggplot are more useful in inspecting the relations between 3 or more variables as the the plot on the previous page
- In multiple regression, the scatter plots of Y versus each predictor variable may or may not show linear patterns. Where the presence of a linear pattern is reassuring, the absence of such a pattern does not imply that our linear model is incorrect.

Hamilton's Data (Regression Analysis By Example page 103)

```
hamilton = read.table("hamilton.txt", h = T)
pairs(hamilton, oma=c(2,2,2,2))
```



Hamilton's Data

```
summary(lm(Y ~ X1, data=hamilton))$coef
```

```
##              Estimate Std. Error    t value    Pr(>|t|)
## (Intercept) 11.988755072  1.2668907  9.463132626 3.399903e-07
## X1           0.003747477  0.4160825  0.009006571 9.929506e-01
```

```
summary(lm(Y ~ X1, data=hamilton))$r.squared
```

```
## [1] 6.239832e-06
```

```
summary(lm(Y ~ X2, data=hamilton))$coef
```

```
##              Estimate Std. Error    t value    Pr(>|t|)
## (Intercept) 10.6319366  0.8109425 13.110593 7.178282e-09
## X2           0.1954562  0.1125087  1.737254 1.059593e-01
```

```
summary(lm(Y ~ X2, data=hamilton))$r.squared
```

```
## [1] 0.1884157
```

When X1 or X2 is the only predictor, neither of them is significant

Hamilton's Data

```
summary(lm(Y ~ X1+X2, data=hamilton))$coef
```

##	Estimate	Std. Error	t value	Pr(> t)
## (Intercept)	-4.515414	0.061141807	-73.85149	2.527961e-17
## X1	3.097008	0.012274433	252.31373	1.010771e-23
## X2	1.031859	0.003684173	280.07891	2.888447e-24

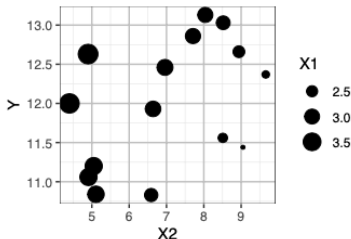
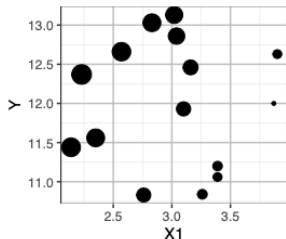
```
summary(lm(Y ~ X1+X2, data=hamilton))$r.squared
```

```
## [1] 0.999847
```

X1 and X2 become highly significant when both are included

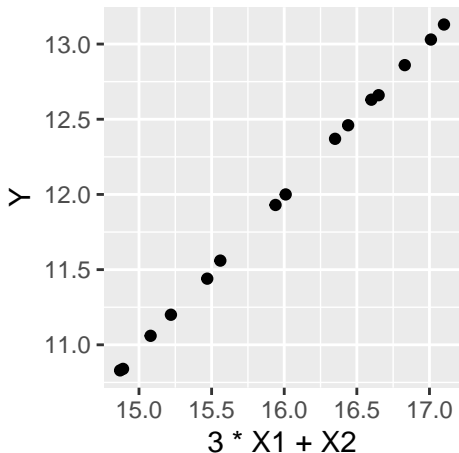
ggplots Can Show X2 Effect on Y After Accounting For X1

```
ggplot(hamilton, aes(x = X1, y = Y, size=X2)) + geom_point()  
ggplot(hamilton, aes(x = X2, y = Y, size=X1)) + geom_point()
```



- For points with similar X1, points with larger size (higher values of X2) have larger Y values. Hence we can see X2 has an effect on Y after accounting for X1.
- Recall β_2 is the effect of X2 on Y when X1 is hold constant.

```
ggplot(hamilton, aes(x = 3*X1+X2, y = Y)) + geom_point()
```



In fact, Y and $3*X1+X2$ are highly correlated.

Check Interactions of Two Numerical Variables

The model

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \epsilon$$

No interaction assumes that

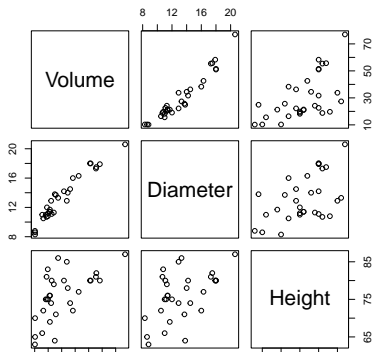
- Y is linear in X_1 for each given X_2 and the slope of X_1 doesn't change with X_2
- Y is linear in X_2 for each given X_1 and the slope of X_2 doesn't change with X_1

Revist the trees data

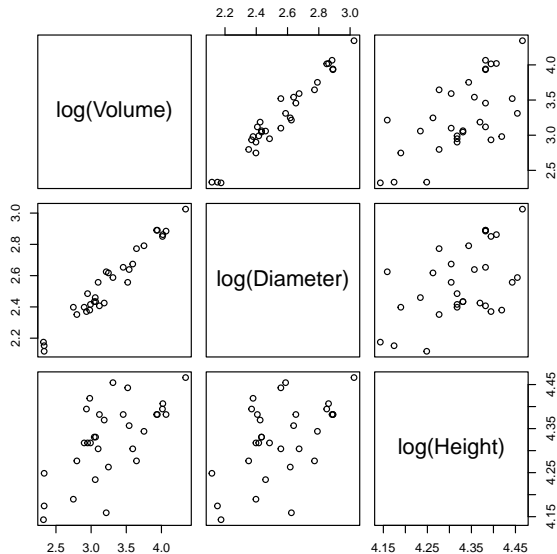
Recall the trees data are measurements of the diameter, height and volume of timber in 31 felled black cherry trees. The variables are

- Girth: Tree diameter in inches measured at 4 ft 6 in above the ground
- Height: Height in ft
- Volume: Volume of timber in cubic ft

```
data("trees")
trees$Diameter = trees$Girth
pairs(Volume ~ Diameter + Height, data=trees, oma = c(2,2,2,2))
```




```
pairs(log(Volume) ~ log(Diameter) + log(Height),  
      data=trees, oma = c(2,2,2,2))
```



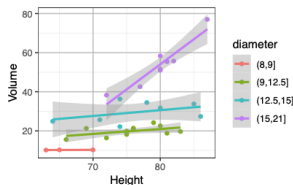
Model Comparison

From the pairwise scatter plots above, the two models below don't seem to differ too much.

- The model $V = \beta_0 + \beta_1 D + \beta_2 H + \epsilon$ implies the slope of H stay the same for each level of D.

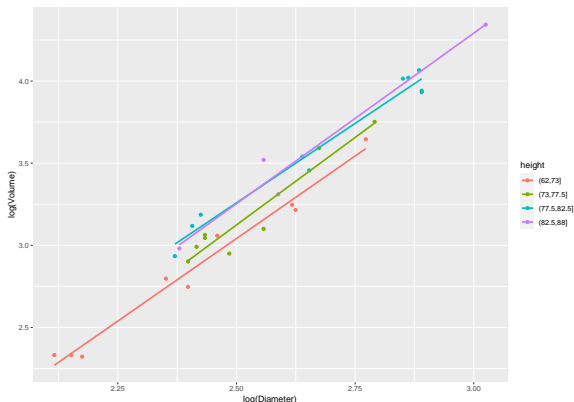
```
trees$diameter = cut(trees$Diameter, breaks=c(8,9,12.5,15,21))
```

```
ggplot(trees, aes(x=Height, y=Volume, color=diameter)) +  
  geom_point() + geom_smooth(method='lm', formula='y~x')
```



Observe the slopes of Height increases as Diameter increases, which means the model $V = \beta_0 + \beta_1 D + \beta_2 H + \epsilon$ isn't appropriate.

```
trees$height = cut(trees$Height, breaks=c(62,73,77.5,82.5,88))
ggplot(trees, aes(x=log(Diameter), y=log(Volume), color=height))
```



The model $\log(V) = \beta_0 + \beta_1 \log(D) + \beta_2 \log(H) + \epsilon$

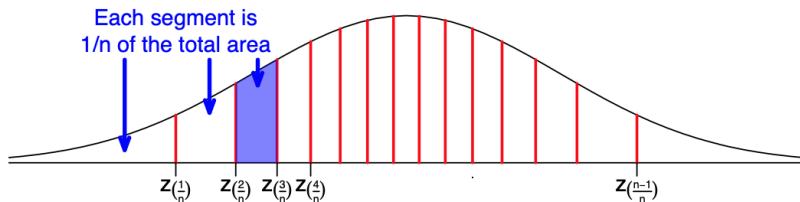
is more appropriate since the slopes of $\log(\text{Height})$ don't change with Diameter as much as before the log-transformation

Normal Probability Plot

- Histogram of the residuals: if normal, should be bell-shaped
 - Pros: simple, easy to understand
 - Cons: for a small sample, histogram may not be bell-shaped even though the sample is from a normal distribution
- Normal probability plot of the residuals
 - also called normal QQ plot, QQ stands for “quantile-quantile”
 - best tool to assess normality

Theory of Normal Probability Plot

- Data: y_1, y_2, \dots, y_n
- Sorted Data: $y_{(1)} \leq y_{(2)} \leq \dots \leq y_{(n)}$, which are called the sample Quantiles
- Theoretical Quantiles of the $N(0, 1)$: $z_{(1/n)}, z_{(2/n)}, \dots, z_{(n-1)/n}$ where, $z_{(k/n)}$ is a value such that $P(Z \leq z_{(k/n)}) = k/n$ for $Z \sim N(0, 1)$.



Theory of Normal Probability Plot

- If $Y \sim N(\mu, \sigma^2)$ and we observed Y_1, Y_2, \dots, Y_n
- We expected k/n of the observations to be $\leq \mu + \sigma z_{(k/n)}$
- If the data are indeed $N(\mu, \sigma^2)$, we will expect that

$$y_{(k)} \approx \mu + \sigma z_{(k/n)}$$

- If one plots the Sample Quantiles $y = \{y_{(k)}\}$ against the Theoretical Quantiles $z = \{z_{(k/n)}\}$, points would fall on the straight line:

$$y = \mu + \sigma z$$

A Technical Remark

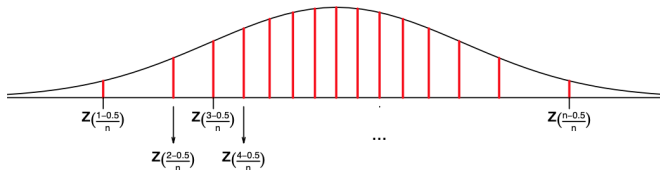
R actually uses the Theoretical Quantiles:

$$Z_{\frac{1-0.5}{n}}, Z_{\frac{2-0.5}{n}}, \dots, Z_{\frac{n-0.5}{n}}$$

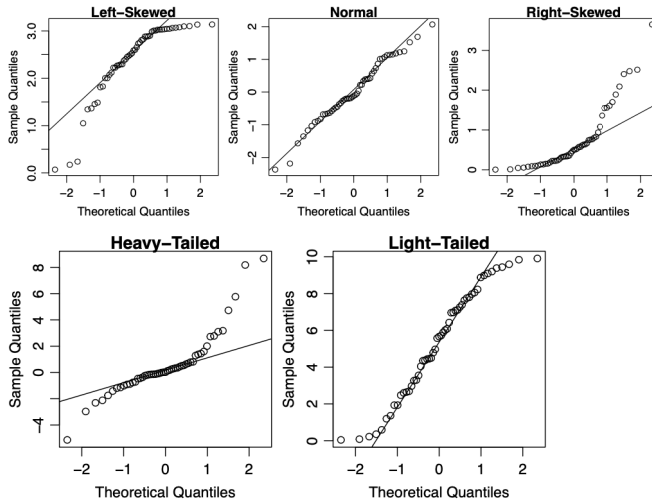
instead of

$$Z_{\frac{1}{n}}, Z_{\frac{2}{n}}, \dots, Z_{\frac{n}{n}}$$

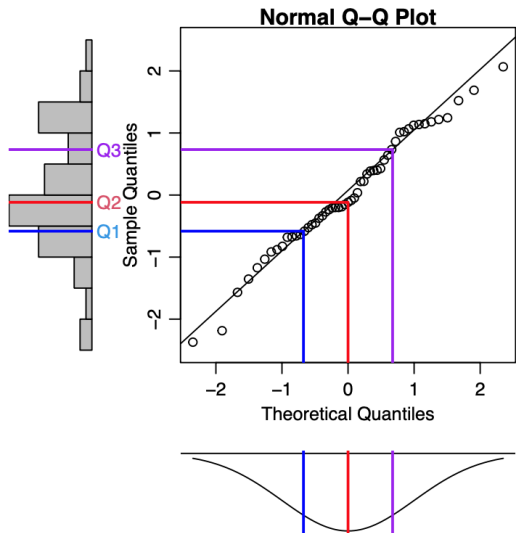
since $z_{(n/n)} = \infty$.



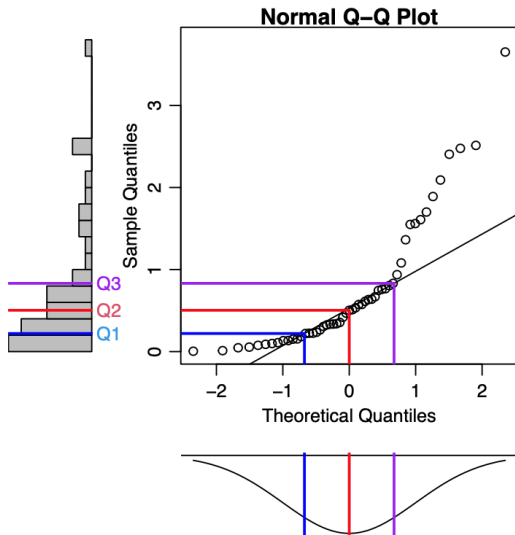
Different QQ Plots



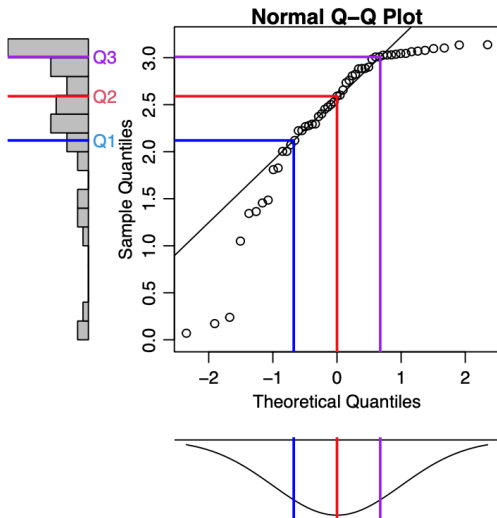
Normal QQ Plot — Normal Data



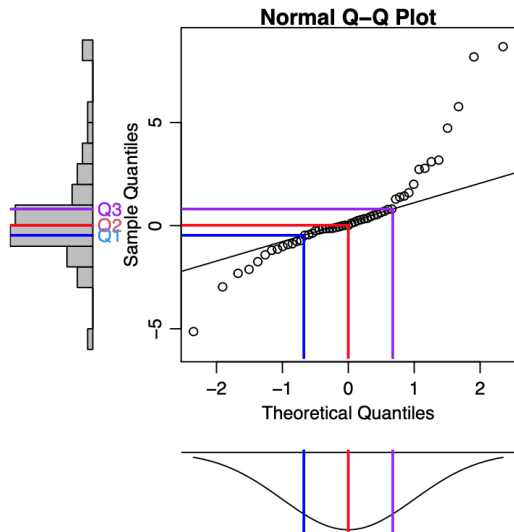
Normal Q-Q Plot — Right-Skewed Data



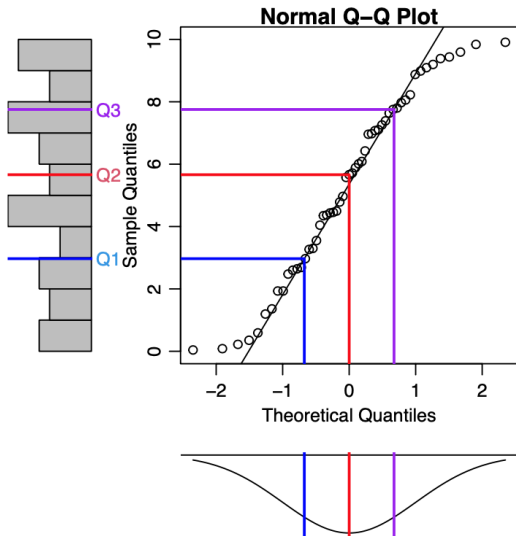
Normal QQ Plot — Left-Skewed Data



Normal QQ Plot — Heavy-Tailed Data



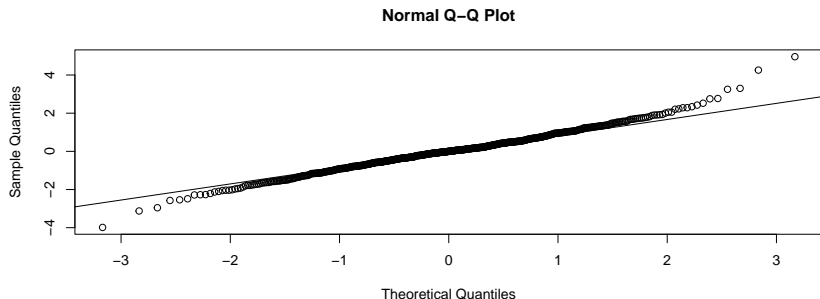
Normal QQ Plot — Light-Tailed Data



Normal QQ Plots in R

The R command `qqnorm()` can make normal QQ plots. The `qqline()` command will add a straight line to the normal QQ plot to help gauging normality.

```
lm1=lm(fev ~ ht + sex, data = fevdata)
qqnorm(rstudent(lm1))
qqline(rstudent(lm1))
```



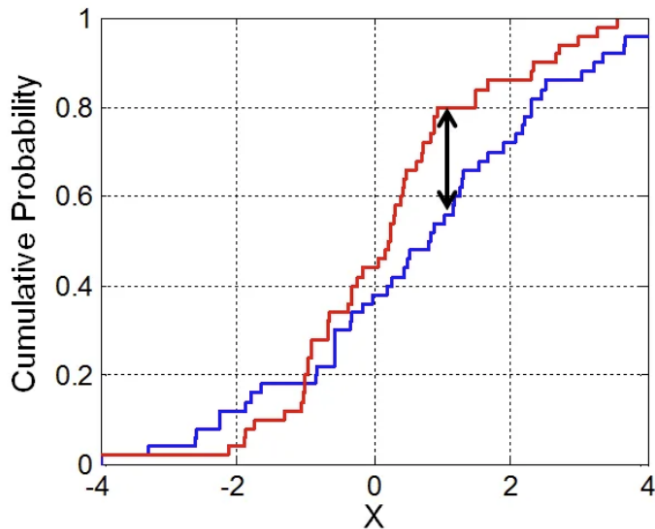
Other Normality Tests

- Shapiro–Wilk test Compute test statistics $W = \frac{b^2}{SS}$, where
$$b = \sum_{i=1}^m a_i(x_{n+1-i} - x_i)$$
 weight a_i can be referenced from the table.
$$SS = \sum_{i=1}^n (x_i - \bar{x})^2$$
- Kolmogorov–Smirnov test

Measure the greatest vertical distance between the two distributions and calculate the test statistic where m and n are the sample sizes

$$D_{n,m} = \sup_x |F_{1,n}(x) - F_{2,m}(x)|$$

Kolmogorov–Smirnov Test



Residual Plus Component Plot

A Residual Plus Component Plot is a scatterplot of

$$(e + \hat{\beta}_j X_j) \quad \text{versus} \quad X_j$$

where e and $\hat{\beta}_j$ are from the regression of Y on all predictors, including X_j .

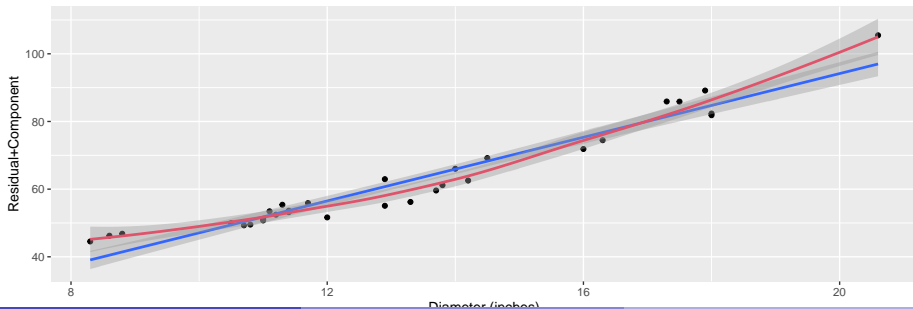
- The slope of the graph is $\hat{\beta}_j$
- This plot is useful to detect non-linearity in the partial relationship between Y and X_j .
- It adds a line indicating where the line of best fit lies

Residual Plus Component Plot (Trees Data)

For the model `lmtrees1 = lm(Volume ~ Diameter + Height, data=trees)`, the Residual Plus Component plot below for Diameter shows clear **nonlinearity**.

```
lmtrees1 = lm(Volume ~ Diameter + Height, data=trees)
ggplot(trees, aes(x=Diameter,
                  y=lmtrees1$res+lmtrees1$coef[2]*Diameter))+
  geom_point() + geom_smooth(method='lm') +
  geom_smooth(method='loess', col=2) +
  labs(x="Diameter (inches)", y="Residual+Component")
```

```
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
```

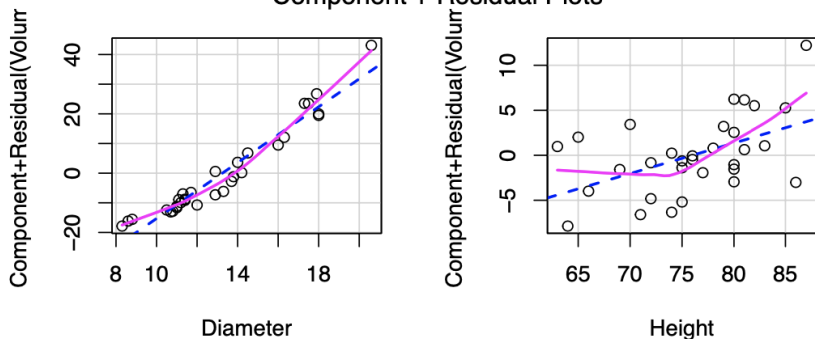


crPlots() in the car Library

The `crPlots()` function in the `car` library can produce residual plus component plots automatically

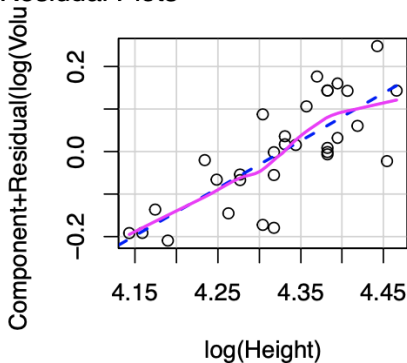
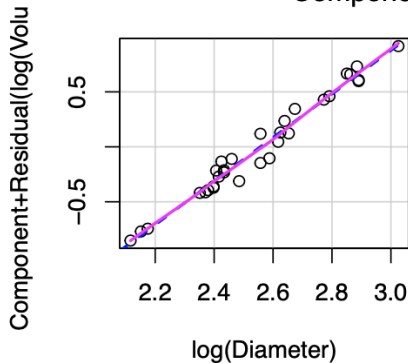
```
library(car)
crPlots(lmtrees1)
```

Component + Residual Plots



```
lmtrees2 = lm(log(Volume) ~ log(Diameter) + log(Height), data=crPlots(lmtrees2))
```

Component + Residual Plots



Both plots look linear, meaning the model

$\log(V) = \beta_0 + \beta_1 \log(D) + \beta_2 \log(H) + \epsilon$ is appropriate.

crPlots() doesn't work if the model includes interactions

```
lm2 = lm(fev ~ age*smoke + age*sex + ht, data=fevdata)
#crPlots(lm2, "ht")
# not work since lm2 includes interactions
```