Linear and Generalized Linear Models in R

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stat
##
##     filter, lag
## The following objects are masked from 'package:base
##
##     intersect, setdiff, setequal, union
```

Introduction

- The family of linear models offers fundamental statistical approaches for investigating the relationship between variables. For example:
  - Determine the relationship between annal sales and advertisement for a company
  - Identify the correlates obesity based on various socio-demographic factors

Linear Models I

Simple linear models

- The simplest form of regression models
- Assumes a linear relationship between the response and predictor variable
- The general form of the simple linear model is:

$$y_i = \beta_0 + x_i\beta_1 x_i + \epsilon_i$$

- **NB**:
  - It is linear because the parameters enter the model linearly – the predictors themselves do not have to be linear
  - Non-linear predictors can be linearised by applying some form of transformation

Multiple linear models

- While Simple Linear Regression models the relationship between a dependent variable and a single predictor, Multiple Linear Regression (MLR) extends this to multiple predictors:

$$y_i = \beta_0 + x_i\beta_1 x_i + \cdots, +\beta_p x_{i,p} + \epsilon_i$$

- Where:
  - y_i is the dependent (response) variable
  - $x_{i,1}, x_{i,2}, \cdots, x_{i,p}$ represent the independent (predictor) variables
  - β_0 is the intercept represent the mean of y_i if all x_i are 0
  - $\beta_1, \cdots, \beta_p$ are regression coefficients representing the effect of each predictor
  - ϵ_i is the random error term assumed to be normally distributed, $\epsilon \sim N(0, \sigma^2)$

Matrix representation

- A convenient representation of this model is the matrix form

$$Y = X\beta + \epsilon$$

- Where:
  - $Y = (y_1, \cdots, y_n)^T$
  - $\epsilon = (\epsilon_1, \cdots, \epsilon_n)^T$
  - $\beta = (\beta_0, \cdots, \beta_p)^T$
  - $X = (1\mathbf{X})$

Estimating β

- The aim is to choose β so that the model explains as much as of the response as possible
  - Problem is finding β so that $X\beta$ is close to Y as possible – the best estimate is $\hat{(\beta)}$
  - The difference between the actual response and the estimated/fitted response is denoted by $\hat{\epsilon}$ and is called the **residual**

Linear Models V

Ordinary Least Squares Method

- We need the estimate of β, $\hat{\beta}$, which minimizes the **sum of squared errors**

$$\sum \epsilon^2 = \epsilon^T \epsilon = (Y - X\beta)^T (Y - X\beta)$$

- We can show that:

$$\begin{aligned} \hat{\beta} &= (X^T X)^{-1} X^T y \\ X\hat{\beta} &= X(X^T X)^{-1} X^T y \\ \hat{y} &= Hy \end{aligned}$$

- Where:
  - $H = X(X^T X)^{-1} X^T$ is called the **hat matrix**
  - $\hat{\epsilon} = y - X\hat{\beta} = y - \hat{y} = (1 - H)y$

Linear Models VI

Estimating $var(\hat{\beta})$

- Provided $var(\epsilon) = \sigma I$, $\hat{\beta}$ is unbiased estimate of β and has a variance of:

$$(X^T X)^{-1}\sigma^2$$

Estimating σ^2

- Since

$$E(\hat{\epsilon}^T \hat{\epsilon}) = \sigma^2(n-p)$$

$$\implies \hat{\sigma}^2 = \frac{\hat{epsilon}^T \hat{\epsilon}}{n-p} = \frac{RSS}{n-p}$$

is unbiased estimate of σ^2, with $n-p$ degrees of freedom.

Goodness of Fit I

- Measures how well the model fits the data
- Once common choice is the R^2, i.e., the coefficient of determination or percentage of variance explained

$$R^2 = 1 - \frac{\sum(\hat{y}_i - y_i)^2}{\sum(y_i - \bar{y})^2} = \frac{\text{RSS}}{\text{TotalSS(Correctedformean)}}$$

- $0 \leq R^2 \leq 1$ - values closer to 1 indicate better fit
- For simple model $R^2 = r^2$, where r is the correlation between x and y.

Example 1 I

Now let's look at an example concerning the number of species of tortoise on the various Galápagos Islands. There are 30 cases (Islands) and seven variables in the dataset.

$$Species = \beta_0 + \beta_1 \times Area + \beta_2 \times Elevation$$
$$+ \beta_3 \times Nearest + \beta_4 \times Scruz$$
$$+ \beta_5 \times Adjacent + \epsilon$$

```
##
## Call:
## lm(formula = Species ~ Area + Elevation + Nearest +
##     data = gala)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -111.679  -34.898   -7.862   33.460  182.584
```

Example 1 II

```
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.068221  19.154198   0.369 0.715351
## Area        -0.023938   0.022422  -1.068 0.296318
## Elevation    0.319465   0.053663   5.953 3.82e-06 *
## Nearest      0.009144   1.054136   0.009 0.993151
## Scruz       -0.240524   0.215402  -1.117 0.275208
## Adjacent    -0.074805   0.017700  -4.226 0.000297 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.
##
## Residual standard error: 60.98 on 24 degrees of fre
## Multiple R-squared:  0.7658, Adjusted R-squared:  0
## F-statistic:  15.7 on 5 and 24 DF,  p-value: 6.838e
```

Example 1 III

```
##                        [,1]
## (Intercept)  7.068220709
## Area        -0.023938338
## Elevation    0.319464761
## Nearest      0.009143961
## Scruz       -0.240524230
## Adjacent    -0.074804832
```

```
### Some important model components
names(mod1)
```

```
##  [1] "coefficients"  "residuals"      "effects"
##  [5] "fitted.values" "assign"         "qr"
##  [9] "xlevels"       "call"           "terms"
```

```
mod1_summ = summary(mod1)
names(mod1_summ)
```

Example 1 IV

```
##  [1] "call"          "terms"         "residuals"
##  [5] "aliased"       "sigma"         "df"
##  [9] "adj.r.squared" "fstatistic"    "cov.unscaled"
```

```r
### Sigma estimate
sqrt(deviance(mod1)/df.residual(mod1))
```

```
## [1] 60.97519
```

```r
mod1_summ$sigma
```

```
## [1] 60.97519
```

```r
### Compute standard error of beta hat
xtxi = mod1_summ$cov.unscaled
sqrt(diag(xtxi)) * mod1_summ$sigma
```

```
## (Intercept)         Area    Elevation      Nearest
## 19.15419782  0.02242235  0.05366280  1.05413595  0.
```

```
### We can also get them directly
mod1_summ$coef[, 2]
```

```
## (Intercept)         Area    Elevation      Nearest
## 19.15419782  0.02242235  0.05366280  1.05413595  0.
```

```
### Compute R square
1 - deviance(mod1)/sum((y - mean(y))^2)
```

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

```
### In-built
mod1_summ$r.squared
```

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

Model diagnostics I

- We make several assumptions to estimate the model parameters
  - $\epsilon \sim N(0, \sigma^2 I)$
    - Independent errors
    - Equal variance
    - Normality
  - We need to check some of these assumptions
- Unusual observations – not all observations can fit the model

Model diagnostics II

Constant variance

- We need to check whether the variance in the residuals is related to some other quantities
  - One way is to look at the fitted values vs the residuals
- Non-constant variance can be handled through:
  - Weighted least squares
  - Transformation

Normality

- All the test and confidence intervals are based on the assumption of normal errors
- We can use Q-Q plots to assess this assumption
- Shapiro test is another alternative test

Model diagnostics III

Correlated errors

- We assume that the errors are uncorrelated. However,
  - Might not be true for temporally or spatially related data
- We can check ϵ_i against ϵ_{i-1} or conduct Durbin-Watson test

Model diagnostics IV

Finding unusual observations

- Some observations do not fit the model well – we call these outliers
- Some may change the model in a substantive manner – we call these influential observations
- Some are unusual points in the predictor space, and have influence on the fit – we call these leverage point
- Define, the leverages $h_i = H_{ii}$
- Since $var(\hat{\epsilon}_i) = \sigma^2(1 - h_i)$, a large leverage, h_i, will make $var(\hat{\epsilon}_i)$ small, i.e., the fit will be forced to be close to y_i
- The standardized residual can be computed from the $var(\hat{\epsilon}_i)$

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

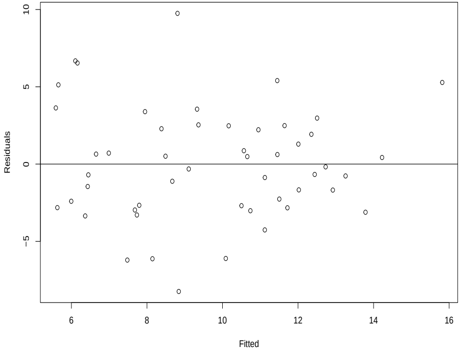
- If the model is correct, $var(r_i) = 1$ and the $corr(r_i, r_j)$ tends to be very small.

Example 2 I

We use `savings` from `faraway` package data to explore various model diagnostics

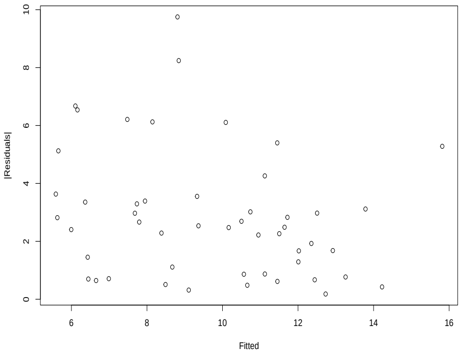
```
## Savings model
data(savings, package = "faraway")
mod2 = lm(sr ~ pop15 + pop75 + dpi + ddpi, data = savi

## First, the residuals vs. fitted plot and the absolu
## values of the residuals vs. fitted plot

### Nonlinear check
plot(fitted(mod2), residuals(mod2), xlab = "Fitted", y
abline(h = 0)
```

Example 2 II




```
### Checking nonconstant variance
plot(fitted(mod2), abs(residuals(mod2)), xlab = "Fitte
```

Example 2 III




```
### Another quick way to check for non-constant varian
summary(lm(abs(residuals(mod2)) ~ fitted(mod2)))
```

Example 2 IV

```
##
## Call:
## lm(formula = abs(residuals(mod2)) ~ fitted(mod2))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.8395 -1.6078 -0.3493  0.6625  6.7036
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    4.8398     1.1865   4.079  0.00017 *
## fitted(mod2)  -0.2035     0.1185  -1.717  0.09250 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.
##
## Residual standard error: 2.163 on 48 degrees of fre
```

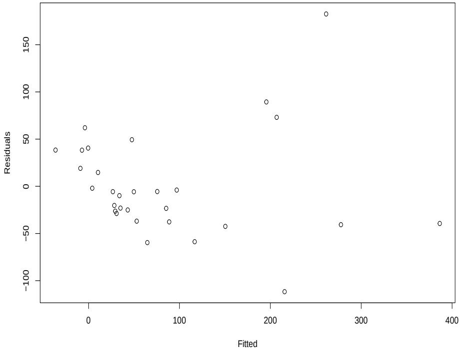
Example 2 V

```
## Multiple R-squared:  0.05784,    Adjusted R-squared
## F-statistic: 2.947 on 1 and 48 DF,  p-value: 0.0925
```

Example 3 I

Now back to `gala` data to show nonconstant variance

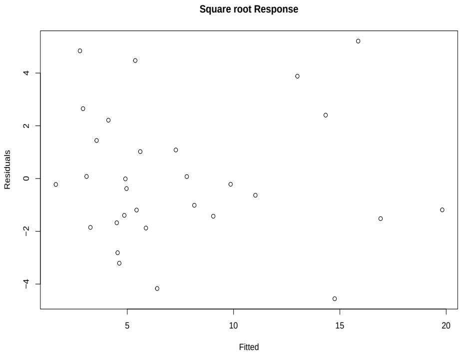
```
### Nonconstant variance
mod3 = lm(Species ~ Area + Elevation + Scruz + Nearest
    gala)
plot(fitted(mod3), residuals(mod3), xlab = "Fitted", y
```



Example 3 II

```
### There are so many approaches to deal with nonconst
### variance, one of them is square root

mod3_s = lm(sqrt(Species) ~ Area + Elevation + Scruz +
    Adjacent, gala)
plot(fitted(mod3_s), residuals(mod3_s), xlab = "Fitted
    main = "Square root Response")
```

Example 3 III

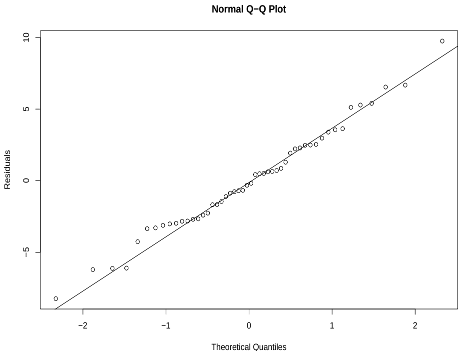


Example 4 I

We use the above model to check for the normality assumption for the residuals using Q-Q plots

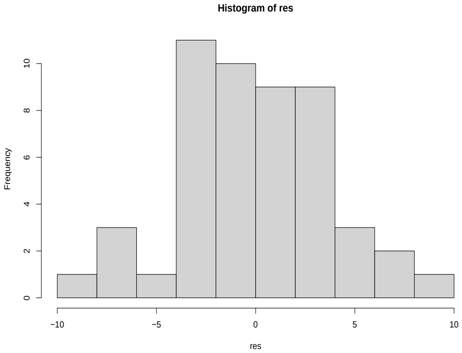
```
### Q-Q plots
res = residuals(mod2)
qqnorm(res, ylab = "Residuals")
qqline(res)
```

Example 4 II




```r
### Check for normality
hist(res)
```

Example 4 III




```r
### Statistical test

#### null hypothesis is that the residuals are normal
shapiro.test(res)
```

Example 4 IV

```
##
##  Shapiro-Wilk normality test
##
## data:  res
## W = 0.98698, p-value = 0.8524
```

Example 5 I

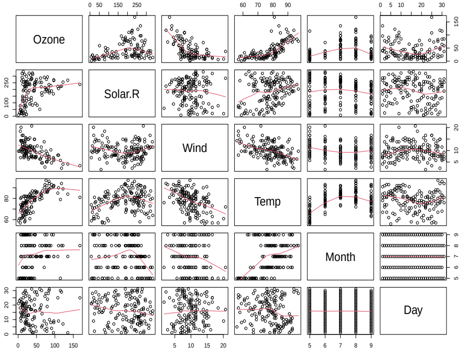
For the example, we use some data taken from an environmental study that measured four variables—ozone, radiation, temperature and wind speed—for 153 consecutive days in New York:

```r
data(airquality)

### Quickly explore the data
head(airquality)
```

```
##   Ozone Solar.R Wind Temp Month Day
## 1    41     190  7.4   67     5   1
## 2    36     118  8.0   72     5   2
## 3    12     149 12.6   74     5   3
## 4    18     313 11.5   62     5   4
## 5    NA      NA 14.3   56     5   5
## 6    28      NA 14.9   66     5   6
```

Example 5 II

```r
### Quickly explore the data
pairs(airquality, panel = panel.smooth)
```



Example 5 III

```
### Air quality model
g = lm(Ozone ~ Solar.R + Wind + Temp, airquality, na.a
summary(g)
```

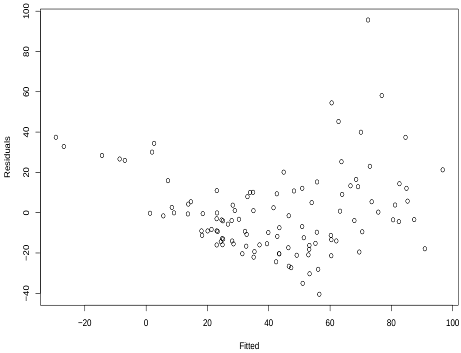
```
##
## Call:
## lm(formula = Ozone ~ Solar.R + Wind + Temp, data =
##     na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -40.485 -14.219  -3.551  10.097  95.619
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -64.34208   23.05472  -2.791  0.00623 *
```

Example 5 IV

```
## Solar.R        0.05982    0.02319   2.580  0.01124 *
## Wind          -3.33359    0.65441  -5.094 1.52e-06 *
## Temp           1.65209    0.25353   6.516 2.42e-09 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.
##
## Residual standard error: 21.18 on 107 degrees of fr
##   (42 observations deleted due to missingness)
## Multiple R-squared:  0.6059, Adjusted R-squared:  0
## F-statistic: 54.83 on 3 and 107 DF,  p-value: < 2.2
```

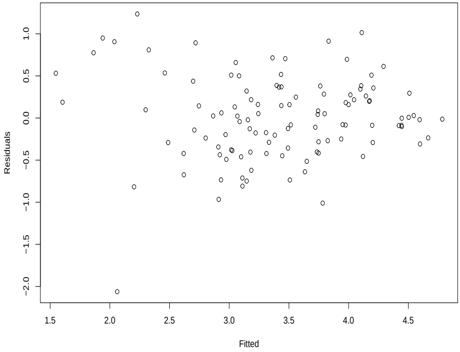
```
plot(fitted(g), residuals(g), xlab = "Fitted", ylab =
```

Example 5 V



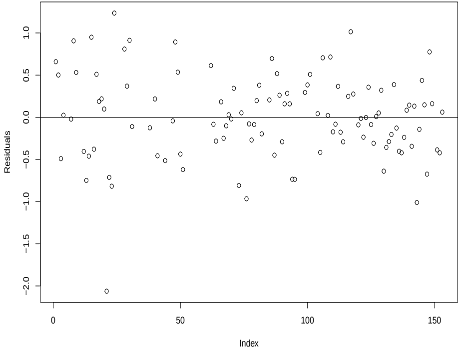

```
### We see some nonconstant variance and nonlinearity
### so we try transforming the response:
gl = lm(log(Ozone) ~ Solar.R + Wind + Temp, airquality
plot(fitted(gl), residuals(gl), xlab = "Fitted", ylab
```

Example 5 VI



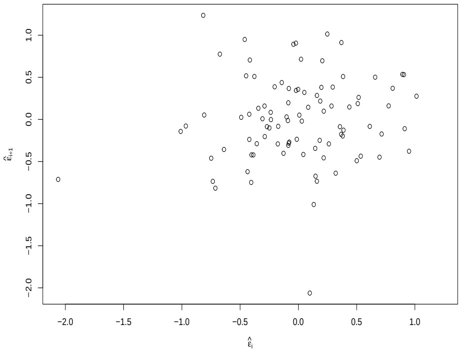

```r
plot(residuals(gl), ylab = "Residuals")
abline(h = 0)
```

Example 5 VII




```
### Unless these effects are strong, they can be diffi
### to spot.  Nothing is obviously wrong here. It is o
### better to plot successive residuals:
plot(residuals(gl)[-153], residuals(gl)[-1], xlab = ex
    ylab = expression(hat(epsilon)[i + 1]))
```

Example 5 VIII




```
### Let's check using a regression of successive
### residuals--the intercept is omitted because residu
### have mean zero:
summary(lm(residuals(gl)[-1] ~ -1 + residuals(gl)[-153
```

Example 5 IX

```
##
## Call:
## lm(formula = residuals(gl)[-1] ~ -1 + residuals(gl)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.07274 -0.28953  0.02583  0.32256  1.32594
##
## Coefficients:
##                     Estimate Std. Error t value Pr(
## residuals(gl)[-153]   0.1104     0.1053   1.048
##
## Residual standard error: 0.5078 on 91 degrees of fr
##   (60 observations deleted due to missingness)
## Multiple R-squared:  0.01193,    Adjusted R-squared
## F-statistic: 1.099 on 1 and 91 DF,  p-value: 0.2973
```

Example 5 X

```
### We can compute the Durbin-Watson statistic: Try it
```

Variable selection I

- The aim is to select the “best” subset of predictors
  - We want to explain the data in the simplest way – remove redundant predictors
- Unnecessarily predictors may be source of noise in the model
- Collinearity
- We will focus on stepwise approaches but you have a look at the criterion approaches (test for goodness of fit)

Forward selection I

- We start with a base model, and add variables one by one. A good starting point is the *null* model
- **Example:** Consider the `mtcars`

Forward selection II

```
## Foward selection

# Load necessary library
library(MASS)

# Load the dataset
data(mtcars)

# Define the full model and the null model
full_model = lm(mpg ~ ., data = mtcars)  # Model with
null_model = lm(mpg ~ 1, data = mtcars)  # Model with

# Perform forward selection using stepwise AIC
forward_model = step(null_model, scope = list(lower =
    upper = full_model), direction = "forward", trace
```

Forward selection III

```
## Start:  AIC=115.94
## mpg ~ 1
##
##        Df Sum of Sq     RSS     AIC
## + wt    1    847.73  278.32  73.217
## + cyl   1    817.71  308.33  76.494
## + disp  1    808.89  317.16  77.397
## + hp    1    678.37  447.67  88.427
## + drat  1    522.48  603.57  97.988
## + vs    1    496.53  629.52  99.335
## + am    1    405.15  720.90 103.672
## + carb  1    341.78  784.27 106.369
## + gear  1    259.75  866.30 109.552
## + qsec  1    197.39  928.66 111.776
## <none>              1126.05 115.943
##
```

Forward selection IV

```
## Step:  AIC=73.22
## mpg ~ wt
##
##        Df Sum of Sq    RSS    AIC
## + cyl   1    87.150 191.17 63.198
## + hp    1    83.274 195.05 63.840
## + qsec  1    82.858 195.46 63.908
## + vs    1    54.228 224.09 68.283
## + carb  1    44.602 233.72 69.628
## + disp  1    31.639 246.68 71.356
## <none>               278.32 73.217
## + drat  1     9.081 269.24 74.156
## + gear  1     1.137 277.19 75.086
## + am    1     0.002 278.32 75.217
##
## Step:  AIC=63.2
```

Forward selection V

```
## mpg ~ wt + cyl
##
##          Df Sum of Sq    RSS    AIC
## + hp      1   14.5514 176.62 62.665
## + carb    1   13.7724 177.40 62.805
## <none>                191.17 63.198
## + qsec    1   10.5674 180.60 63.378
## + gear    1    3.0281 188.14 64.687
## + disp    1    2.6796 188.49 64.746
## + vs      1    0.7059 190.47 65.080
## + am      1    0.1249 191.05 65.177
## + drat    1    0.0010 191.17 65.198
##
## Step:  AIC=62.66
## mpg ~ wt + cyl + hp
##
```

Forward selection VI

```
##          Df Sum of Sq    RSS    AIC
## <none>                176.62 62.665
## + am    1    6.6228 170.00 63.442
## + disp  1    6.1762 170.44 63.526
## + carb  1    2.5187 174.10 64.205
## + drat  1    2.2453 174.38 64.255
## + qsec  1    1.4010 175.22 64.410
## + gear  1    0.8558 175.76 64.509
## + vs    1    0.0599 176.56 64.654
```

```
# Display the selected model
summary(forward_model)
```

Forward selection VII

```
##
## Call:
## lm(formula = mpg ~ wt + cyl + hp, data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.9290 -1.5598 -0.5311  1.1850  5.8986
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 38.75179    1.78686  21.687  < 2e-16 **
## wt          -3.16697    0.74058  -4.276 0.000199 **
## cyl         -0.94162    0.55092  -1.709 0.098480 .
## hp          -0.01804    0.01188  -1.519 0.140015
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.
```

Forward selection VIII

```
##
## Residual standard error: 2.512 on 28 degrees of fre
## Multiple R-squared:  0.8431, Adjusted R-squared:  0
## F-statistic: 50.17 on 3 and 28 DF,  p-value: 2.184e
```

Backward selection I

- Involves starting with a large model and removing the terms one by one.

```r
# Define the full model with all predictors
full_model = lm(mpg ~ ., data = mtcars)

# Perform backward selection using AIC
best_model = step(full_model, direction = "backward",
```

```
## Start:  AIC=70.9
## mpg ~ cyl + disp + hp + drat + wt + qsec + vs + am
##
##        Df Sum of Sq    RSS    AIC
## - cyl   1    0.0799 147.57 68.915
## - vs    1    0.1601 147.66 68.932
## - carb  1    0.4067 147.90 68.986
## - gear  1    1.3531 148.85 69.190
```

Backward selection II

```
## - drat  1    1.6270 149.12 69.249
## - disp  1    3.9167 151.41 69.736
## - hp    1    6.8399 154.33 70.348
## - qsec  1    8.8641 156.36 70.765
## <none>              147.49 70.898
## - am    1   10.5467 158.04 71.108
## - wt    1   27.0144 174.51 74.280
##
## Step:  AIC=68.92
## mpg ~ disp + hp + drat + wt + qsec + vs + am + gear
##
##        Df Sum of Sq    RSS    AIC
## - vs    1    0.2685 147.84 66.973
## - carb  1    0.5201 148.09 67.028
## - gear  1    1.8211 149.40 67.308
## - drat  1    1.9826 149.56 67.342
```

Backward selection III

```
## - disp  1    3.9009 151.47 67.750
## - hp    1    7.3632 154.94 68.473
## <none>               147.57 68.915
## - qsec  1   10.0933 157.67 69.032
## - am    1   11.8359 159.41 69.384
## - wt    1   27.0280 174.60 72.297
##
## Step:  AIC=66.97
## mpg ~ disp + hp + drat + wt + qsec + am + gear + ca
##
##        Df Sum of Sq    RSS    AIC
## - carb  1    0.6855 148.53 65.121
## - gear  1    2.1437 149.99 65.434
## - drat  1    2.2139 150.06 65.449
## - disp  1    3.6467 151.49 65.753
## - hp    1    7.1060 154.95 66.475
```

Backward selection IV

```
## <none>                     147.84 66.973
## - am     1    11.5694 159.41 67.384
## - qsec   1    15.6830 163.53 68.200
## - wt     1    27.3799 175.22 70.410
##
## Step:  AIC=65.12
## mpg ~ disp + hp + drat + wt + qsec + am + gear
##
##        Df Sum of Sq    RSS    AIC
## - gear  1     1.565 150.09 63.457
## - drat  1     1.932 150.46 63.535
## <none>              148.53 65.121
## - disp  1    10.110 158.64 65.229
## - am    1    12.323 160.85 65.672
## - hp    1    14.826 163.35 66.166
## - qsec  1    26.408 174.94 68.358
```

Backward selection V

```
## - wt     1     69.127 217.66 75.350
##
## Step:  AIC=63.46
## mpg ~ disp + hp + drat + wt + qsec + am
##
##        Df Sum of Sq    RSS    AIC
## - drat  1     3.345 153.44 62.162
## - disp  1     8.545 158.64 63.229
## <none>              150.09 63.457
## - hp    1    13.285 163.38 64.171
## - am    1    20.036 170.13 65.466
## - qsec  1    25.574 175.67 66.491
## - wt    1    67.572 217.66 73.351
##
## Step:  AIC=62.16
## mpg ~ disp + hp + wt + qsec + am
```

Backward selection VI

```
##
##          Df Sum of Sq    RSS    AIC
## - disp   1     6.629 160.07 61.515
## <none>                153.44 62.162
## - hp     1    12.572 166.01 62.682
## - qsec   1    26.470 179.91 65.255
## - am     1    32.198 185.63 66.258
## - wt     1    69.043 222.48 72.051
##
## Step:  AIC=61.52
## mpg ~ hp + wt + qsec + am
##
##          Df Sum of Sq    RSS    AIC
## - hp     1     9.219 169.29 61.307
## <none>                160.07 61.515
## - qsec   1    20.225 180.29 63.323
```

Backward selection VII

```
## - am     1    25.993 186.06 64.331
## - wt     1    78.494 238.56 72.284
##
## Step:  AIC=61.31
## mpg ~ wt + qsec + am
##
##        Df Sum of Sq    RSS    AIC
## <none>              169.29 61.307
## - am    1    26.178 195.46 63.908
## - qsec  1   109.034 278.32 75.217
## - wt    1   183.347 352.63 82.790
```

```
# Display the selected model
summary(best_model)
```

Backward selection VIII

```
##
## Call:
## lm(formula = mpg ~ wt + qsec + am, data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4811 -1.5555 -0.7257  1.4110  4.6610
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.6178     6.9596   1.382 0.177915
## wt           -3.9165     0.7112  -5.507 6.95e-06 **
## qsec          1.2259     0.2887   4.247 0.000216 **
## am            2.9358     1.4109   2.081 0.046716 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.
```

Backward selection IX

```
##
## Residual standard error: 2.459 on 28 degrees of fre
## Multiple R-squared:  0.8497, Adjusted R-squared:  0
## F-statistic: 52.75 on 3 and 28 DF,  p-value: 1.21e-
```

Stepwise selection

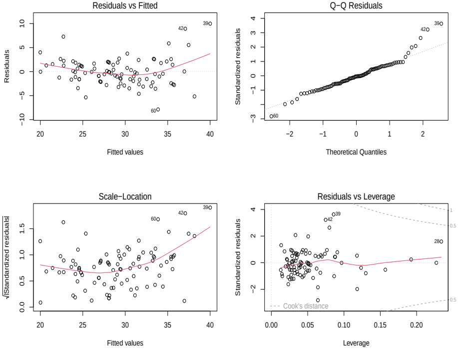
- A combination of both forward and backward selection. It uses goodness of fit such as Akaike Information Criteria (AIC) to select the model

Diagnostic plots I

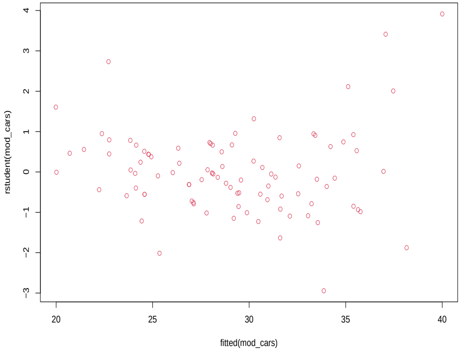
- There are number of plots we can use to assess the model:
    - **Residual plots** - Assess patterns in the residuals
    - **Q-Q plots** - Assess normality of the residuals
    - **Cook's Distance** - Identify influential points

Examples I

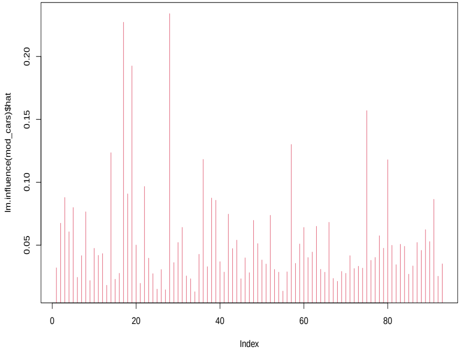
Here we use the `Cars93` from `MASS` package. Refer to the R script for the codes



Examples II



Examples III




```
## integer(0)
```

Transformations I

- We have only covered/assumed linear models
- At times, transformation of the response variable may improve model fit
  - However, we may need to consider the trade-off between prediction and inference
- One of such transformation is the Box-Cox

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

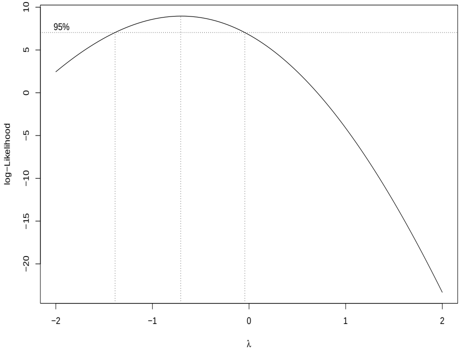
- where $y > 0$ and λ is a transformation parameter.
- In some cases, we need to add a constant to the response to ensure it is positive before applying transformation.

Transformations II

```
##
## Call:
## lm(formula = MPG.highway ~ Weight, data = Cars93)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.6501 -1.8359 -0.0774  1.8235 11.6172
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 51.6013654  1.7355498   29.73   <2e-16
## Weight      -0.0073271  0.0005548  -13.21   <2e-16
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.
##
## Residual standard error: 3.139 on 91 degrees of fre
```

Transformations III

```
## Multiple R-squared:  0.6572, Adjusted R-squared:  0
## F-statistic: 174.4 on 1 and 91 DF,  p-value: < 2.2e
```



Transformations IV

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##        Min         1Q     Median         3Q
## -0.0090961 -0.0019929 -0.0000507  0.0023903  0.0076
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  9.901e-01  1.845e-03  536.62   <2e-16
## x           -8.290e-06  5.898e-07  -14.06   <2e-16
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.
##
## Residual standard error: 0.003337 on 91 degrees of
```

Transformations V

```
## Multiple R-squared:  0.6847, Adjusted R-squared:  0
## F-statistic: 197.6 on 1 and 91 DF,  p-value: < 2.2e
```

Polynomial Regression I

- So far, we have considered models that are linear in both parameter space and predictors
- We can include higher order and interaction terms. For example:

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2^2 + \beta_3 x_1 x_2 + \epsilon$$

Factors I

- Factors are categorical variables with levels
  - May be numbers but just treated as labels
- **Example**: Gender, Cancer types, Education level, etc
- We can use factors to identify groups in the data

```
df = iris
avg_df = (df |>
    group_by(Species) |>
    summarize(sepal_mean = mean(Sepal.Length), sepal_s
print(avg_df)
```

```
## # A tibble: 3 x 3
##   Species    sepal_mean sepal_sd
##   <fct>           <dbl>    <dbl>
## 1 setosa           5.01    0.352
## 2 versicolor       5.94    0.516
```

Factors II

```
## 3 virginica           6.59    0.636
```

- We can also create factors using `factor()`.

```
#### Generate data
nsamples = 100
df = data.frame(gender = sample(c(0, 1), size = nsampl
    age = runif(nsamples, 18, 100))
head(df, 3)
```

```
##   gender      age
## 1      1 57.13880
## 2      0 18.80626
## 3      1 52.56644
```

Factors III

```r
#### Convert 0,1 in gender to factor
df = (df |>
    mutate(gender = factor(gender, levels = c(0, 1), l
        "Male"))))
head(df, 3)
```

```
##   gender      age
## 1   Male 57.13880
## 2 Female 18.80626
## 3   Male 52.56644
```

- We can use **ANOVA** as an alternative to linear models models with factor explanatory variables.

One-way Analysis of Variance I

- Consider the hypothesis to test the *null hypothesis* that three or more population means are equal
  - This could be gene expression values for cancer patients with different treatment options or cancer types. The treatment options or cancer types are the factors
- Thus, the null hypothesis becomes:

$$\mathrm{H}_0 : \mu_1 = \mu_2 = \mu_3$$

- Let the data in each group be as follows

$$y_1 = \{y_{11}, y_{21}, y_{31}, \cdots, y_{n1}\}$$
$$y_2 = \{y_{12}, y_{22}, y_{32}, \cdots, y_{n2}\}$$
$$y_3 = \{y_{13}, y_{23}, y_{33}, \cdots, y_{n3}\}$$

One-way Analysis of Variance II

- The sample means are:

$$\bar{y}_1 = \frac{1}{n_1}\sum_{i=1}^{n_1} y_{i1}$$

$$\bar{y}_2 = \frac{1}{n_2}\sum_{i=1}^{n_2} y_{i2}$$

$$\bar{y}_3 = \frac{1}{n_3}\sum_{i=1}^{n_3} y_{i3}$$

- Thus:

$$\begin{aligned}\bar{y} &= \frac{1}{n}\left(\sum_{i=1}^{n_1} y_{i1} + \sum_{i=1}^{n_2} y_{i2} + \sum_{i=1}^{n_3} y_{i3}\right)\\ &= \bar{y}_1 + \bar{y}_2 + \bar{y}_3\end{aligned}$$

One-way Analysis of Variance III

- is the overall mean
- We want to test on the equality of the means – the sum of squares

Sum pf Squares Within (SSW)

- Sum of squared deviations of the measurements to their group mean:

$$SSW = \sum_{j=1}^{g} \sum_{i=1}^{n} (y_{ij} - \bar{y}_j)^2$$

- where g is the number of groups.

One-way Analysis of Variance IV

Sum of Squares Between (SSB)

- Sum of squares of the deviations of the group mean with respect to the total mean:

$$SSB = \sum_{j=1}^{g} n_j(\bar{y}_j - \bar{y})^2$$

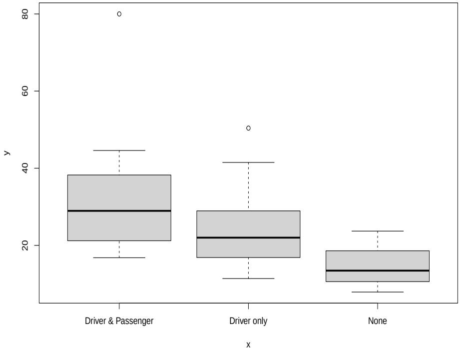
- Thus, the f-value is given by

$$f = \frac{SSB/(g-1)}{SSW/(N-g)}$$

- If the data is normally distributed then $f \sim F_{g-1,N-g}$ distribution, where, $g-1$ and $N-g$ are the degrees of freedom.
- We fail to reject the null hypothesis if $P(g-1, N-g > f) \geq \alpha$

Examples I

Example 1: Consider the following question: does the provision of airbags affect the maximum price that people are willing to pay for a car?

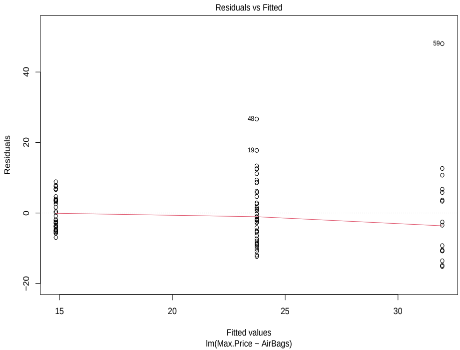


Examples II

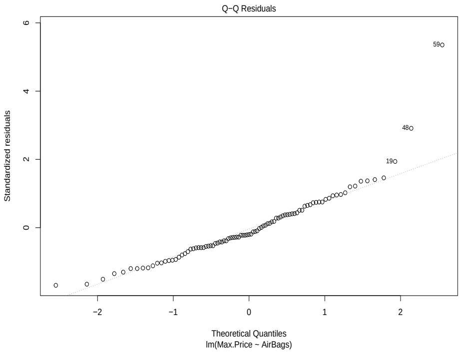
```
##
## Call:
## lm(formula = Max.Price ~ AirBags, data = Cars93)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.16  -5.34  -1.84   4.66  48.04
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>
## (Intercept)          31.962      2.317  13.794  < 2
## AirBagsDriver only   -8.223      2.714  -3.030   0.
## AirBagsNone         -17.127      2.810  -6.095 2.67
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.
##
```

Examples III

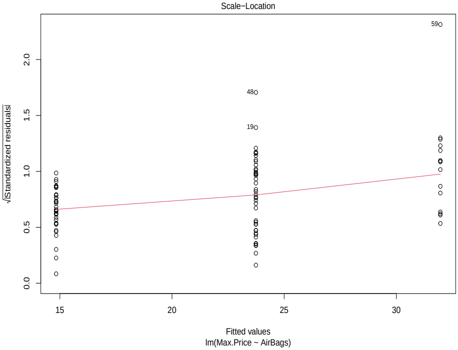
```
## Residual standard error: 9.268 on 90 degrees of fre
## Multiple R-squared:  0.3093, Adjusted R-squared:  0
## F-statistic: 20.15 on 2 and 90 DF,  p-value: 5.852e
```



Examples IV

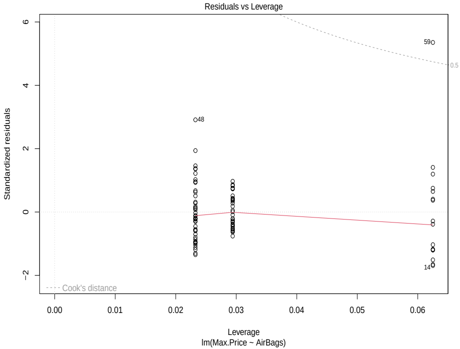

Q−Q Residuals
Standardized residuals
0
2
4
6
−2
−1
0
1
2
Theoretical Quantiles
lm(Max.Price ~ AirBags)
59
48
19

Examples V



Scale−Location
59
48
19
0.0
0.5
1.0
1.5
2.0
√|Standardized residuals|
15
20
25
30
Fitted values
lm(Max.Price ~ AirBags)

Examples VI




```
##
##  F test to compare two variances
##
## data:  MaxP0 and MaxP1
## F = 0.30309, num df = 33, denom df = 42, p-value =
## alternative hypothesis: true ratio of variances is
```

Examples VII

```
## 95 percent confidence interval:
##  0.1595059 0.5910821
## sample estimates:
## ratio of variances
##          0.3030924

##
##  F test to compare two variances
##
## data:  MaxP0 and MaxP2
## F = 0.091921, num df = 33, denom df = 15, p-value =
## alternative hypothesis: true ratio of variances is
## 95 percent confidence interval:
##  0.03504938 0.20783657
## sample estimates:
## ratio of variances
```

Examples VIII

```
##          0.09192053

##
##  F test to compare two variances
##
## data:  MaxP1 and MaxP2
## F = 0.30328, num df = 42, denom df = 15, p-value =
## alternative hypothesis: true ratio of variances is
## 95 percent confidence interval:
##  0.1177105 0.6564096
## sample estimates:
## ratio of variances
##          0.3032757
```

Example 2: Let's sample data from the normal distribution with mean 1.9 and standard deviation 0.5 corresponding to three groups of patients that do not possess any type of differences between groups.

Examples IX

```
##  [1] 1.77 1.98 1.75 1.57 1.69 2.14 2.09 2.17 1.02 1
## [1] 0.2593042
```

Two-way Analysis of Variance

- Is the extension of the one-way ANOVA to include more factors:

$$Y_{ijk} = \alpha_i + \beta_j + (\alpha\beta)_{ij} + \epsilon_{ijk}$$

- Where:
  - α_i is the group mean of the ith group
  - β_j is the group mean of the jth group
  - $(\alpha\beta)_{ij}$ is the interaction effect
  - $\epsilon_{ijk} \sim N(0, \sigma^2)$

Example 1: We may ask whether, in addition to provision of airbags, the availability of manual transmission explains differences in maximum price.

```
## Analysis of Variance Table
##
## Response: Max.Price
##                   Df Sum Sq Mean Sq F value    Pr(>F)
## AirBags            2 3462.5 1731.26 20.7783 3.934e-08
## Man.trans.avail    1  315.7  315.69  3.7889   0.05475
## Residuals         89 7415.5   83.32
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.
```

Examples II

```
## Analysis of Variance Table
##
## Response: Max.Price
##                          Df Sum Sq Mean Sq F value
## AirBags                   2 3462.5 1731.26 20.4462 5
## Man.trans.avail           1  315.7  315.69  3.7283
## AirBags:Man.trans.avail   2   48.9   24.45  0.2887
## Residuals                87 7366.6   84.67
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.
```

```
## Analysis of Variance Table
##
## Response: Max.Price
##                          Df Sum Sq Mean Sq F value
## AirBags                   2 3462.5 1731.26 20.4462 5
```

Examples III

```
## Man.trans.avail            1  315.7  315.69  3.7283
## AirBags:Man.trans.avail    2   48.9   24.45  0.2887
## Residuals                 87 7366.6   84.67
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.
```

- Here, we are interested in investigating the relationship between the response and the covariates for different factor levels.

Robust tests

- When the normality and homoscedasticity assumptions are violated, we can use alternative tests – which are robust to these assumptions.

Generalized Linear Models I

- Generalized Linear Models (GLMs) extend linear regression to allow response variables that follow different distributions from the exponential family.
- Key Features:
  - Response variable can follow **Binomial, Poisson, Normal, or Gamma distributions**.
  - A **link function** connects the expected response to a linear predictor.
  - Allows modeling of **binary outcomes, count data, and continuous positive values**.
- A GLM assumes the response variable Y follows an exponential family distribution.
- Common Exponential Family Distributions

Generalized Linear Models II

Distribution	Mean $E[Y]$	Variance $\text{Var}(Y)$	Canonical Link
Normal	μ	σ^2	Identity μ
Binomial	np	$np(1-p)$	Logit $\log \frac{p}{1-p}$
Poisson	λ	λ	Log $\log(\lambda)$
Gamma	$\alpha\beta$	$\alpha\beta^2$	Inverse $\frac{1}{\mu}$

Components of a GLM

1. **Random Component**: Specifies the **distribution** of Y from the **exponential family**.
2. **Systematic Component** (Linear Predictor):

$$\eta = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

3. **Link Function**: Relates the mean response μ to the linear predictor η.

Common Generalized Linear Models

Logistic Regression (Binary Outcomes)

Used when $Y \in \{0, 1\}$.

- **Random Component**: $Y_i \sim \text{Binomial}(n_i, p_i)$
- **Link Function**: Logit

$$\eta = \log \frac{p}{1-p}$$

Poisson Regression (Count Data)

Used for modeling count data.

- **Random Component**: $Y_i \sim \text{Poisson}(\lambda_i)$
- **Link Function**: Log

$$\eta = \log(\lambda)$$

Model Evaluation

Goodness of Fit

- **Deviance**: Measures how well the model fits the data.
- **Akaike Information Criterion (AIC)**: Lower AIC is better.

Summary

- **GLMs generalize linear regression** to non-normal response variables.
- **Common models include logistic, Poisson, and Gamma regression**.
- **MLE is used for parameter estimation**.
- **Performance is assessed using deviance and AIC**.