Multiple Regression Wisdom - Chapter 24

Alan Arnholt

Last updated: May 10, 2022 at 09:35:59 AM

Contents

1 Read in data with `read.csv()` 1
1.1 Base R Graph . . . 2
1.2 Using `ggplot2` . . . 2
1.3 Using `plotly` . . . 5
1.4 Scatterplot Matrices . . . 6

2 Basic Regression 6
2.1 Confidence Interval for β_1 . . . 10
2.2 Multiple Linear Regression . . . 14
2.3 Is There a Relationship Between the Response and Predictors? . . . 17
2.4 Variable Selection . . . 19
2.5 Diagnostic Plots . . . 24
2.6 Non-Additive Models . . . 28
2.7 Qualitative Predictors . . . 29
2.8 Moving On Now . . . 33
2.9 Matrix Scatterplots . . . 39
2.10 More Diagnostic Plots . . . 45
2.11 Non-linear Relationships . . . 48
2.12 Variance Inflation Factor (VIF) . . . 52
2.13 Exercise . . . 55

1 Read in data with `read.csv()`

```
site <- "http://statlearning.com/s/Advertising.csv"
AD <- read.csv(site)
head(AD)
```

```
  X    TV radio newspaper sales
1 1 230.1  37.8      69.2  22.1
2 2  44.5  39.3      45.1  10.4
3 3  17.2  45.9      69.3   9.3
4 4 151.5  41.3      58.5  18.5
5 5 180.8  10.8      58.4  12.9
6 6   8.7  48.9      75.0   7.2
```

```
dim(AD)
```

```
[1] 200   5
```

```r
library(DT)
datatable(AD[, -1], rownames = FALSE,
          caption = 'Table 1: This is a simple caption for the table.')
```

Show 10 entries Search:

Table 1: This is a simple caption for the table.

TV	radio	newspaper	sales
230.1	37.8	69.2	22.1
44.5	39.3	45.1	10.4
17.2	45.9	69.3	9.3
151.5	41.3	58.5	18.5
180.8	10.8	58.4	12.9
8.7	48.9	75	7.2
57.5	32.8	23.5	11.8
120.2	19.6	11.6	13.2
8.6	2.1	1	4.8
199.8	2.6	21.2	10.6

Showing 1 to 10 of 200 entries Previous 1 2 3 4 5 … 20 Next

1.1 Base R Graph

```r
plot(sales ~ TV, data = AD, col = "red", pch = 19)
mod1 <- lm(sales ~ TV, data = AD)
abline(mod1, col = "blue")
```

```r
par(mfrow = c(1, 3))
plot(sales ~ TV, data = AD, col = "red", pch = 19)
mod1 <- lm(sales ~ TV, data = AD)
abline(mod1, col = "blue")
plot(sales ~ radio, data = AD, col = "red", pch = 19)
mod2 <- lm(sales ~ radio, data = AD)
abline(mod2, col = "blue")
plot(sales ~ newspaper, data = AD, col = "red", pch = 19)
mod3 <- lm(sales ~ newspaper, data = AD)
abline(mod3, col = "blue")
```

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

Change the caption in Figure 2.

1.2 Using ggplot2

```r
library(ggplot2)
library(MASS)
p <- ggplot(data = AD, aes(x = TV, y = sales)) +
```

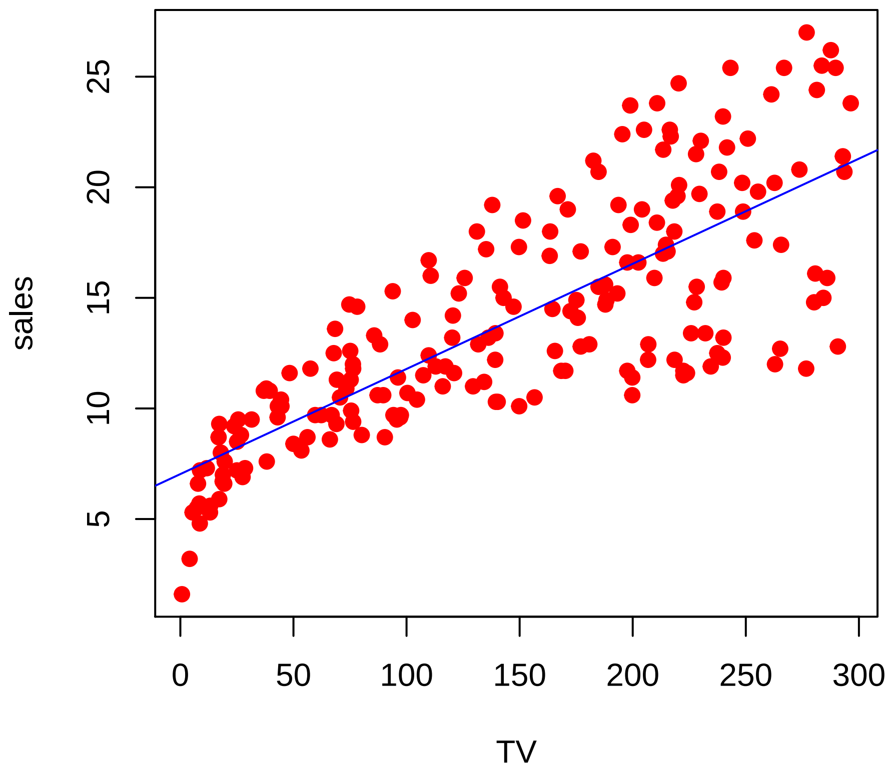


Figure 1: Base R scatterplot of ‘sales‘ versus ‘TV‘

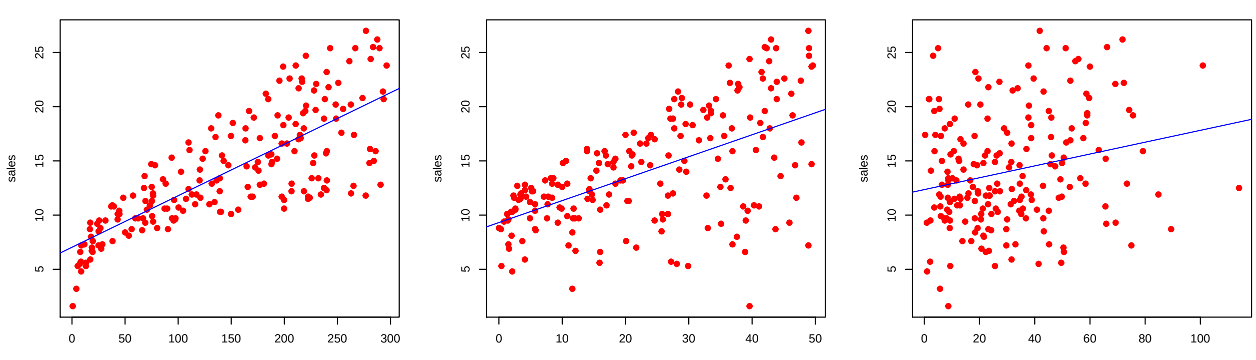
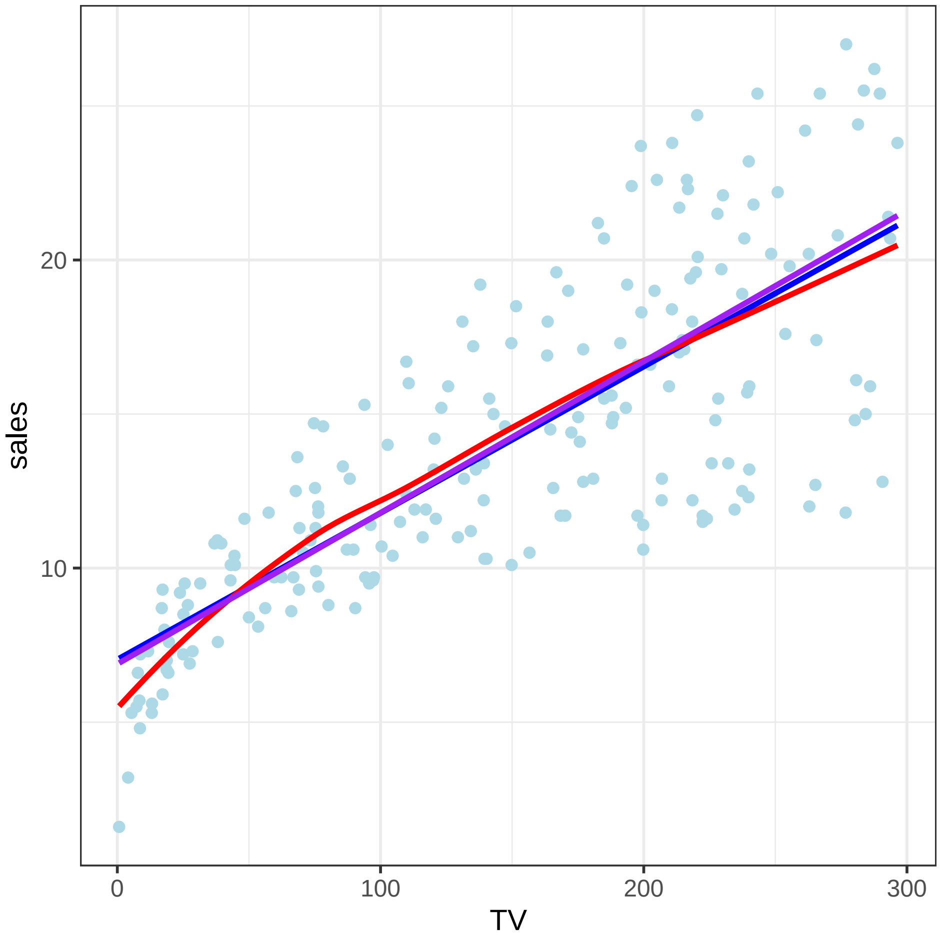


Figure 2: You should change this caption

```r
  geom_point(color = "lightblue") +
  geom_smooth(method = "lm", se = FALSE, color = "blue") +
  geom_smooth(method = "loess", color = "red", se = FALSE) +
  geom_smooth(method = "rlm", color = "purple", se = FALSE) +
  theme_bw()
p
```




```r
library(gridExtra)
p1 <- ggplot(data = AD, aes(x = TV, y = sales)) +
        geom_point(color = "lightblue") +
        geom_smooth(method = "lm", se = FALSE, color = "blue") +
        theme_bw()
p2 <- ggplot(data = AD, aes(x = radio, y = sales)) +
        geom_point(color = "lightblue") +
        geom_smooth(method = "lm", se = FALSE, color = "blue") +
        theme_bw()
p3 <- ggplot(data = AD, aes(x = newspaper, y = sales)) +
        geom_point(color = "lightblue") +
        geom_smooth(method = "lm", se = FALSE, color = "blue") +
        theme_bw()
grid.arrange(p1, p2, p3, ncol = 3)
```

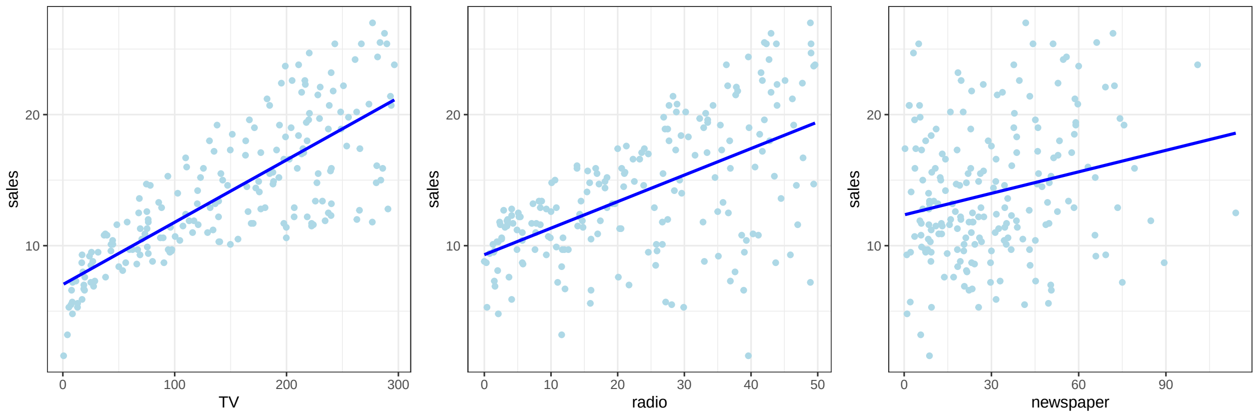
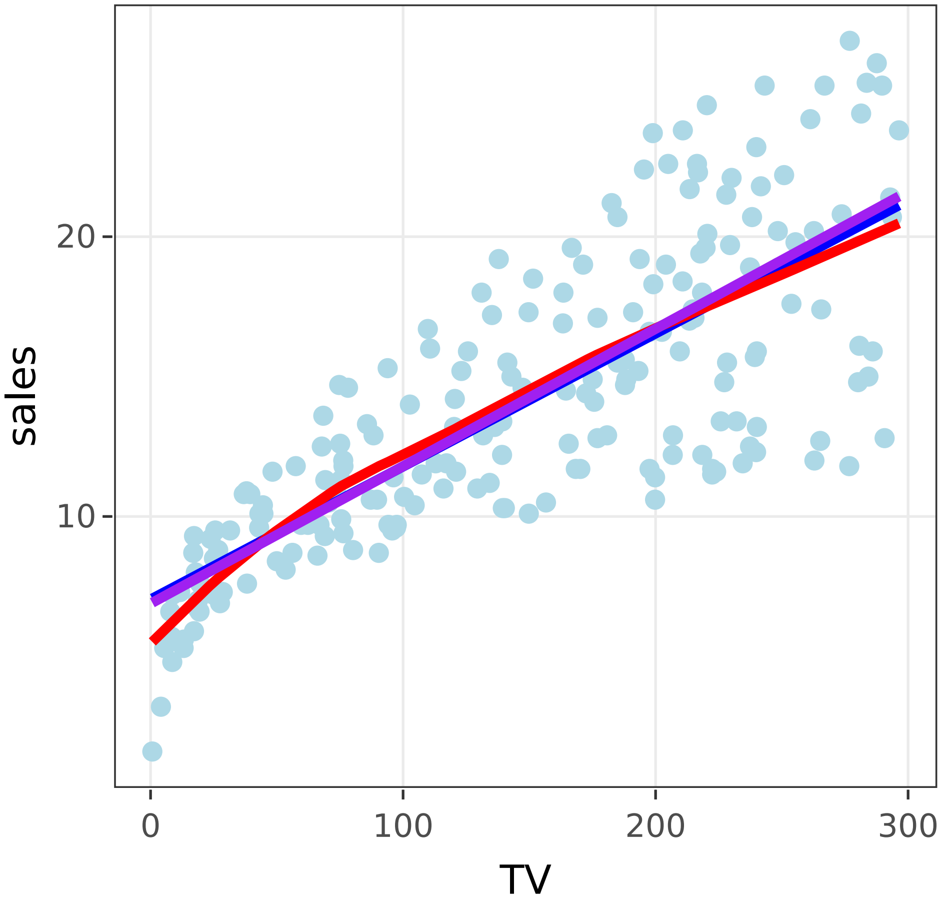


Figure 3: Using ‘grid.arrange()‘ with ‘ggplot‘

1.3 Using `plotly`

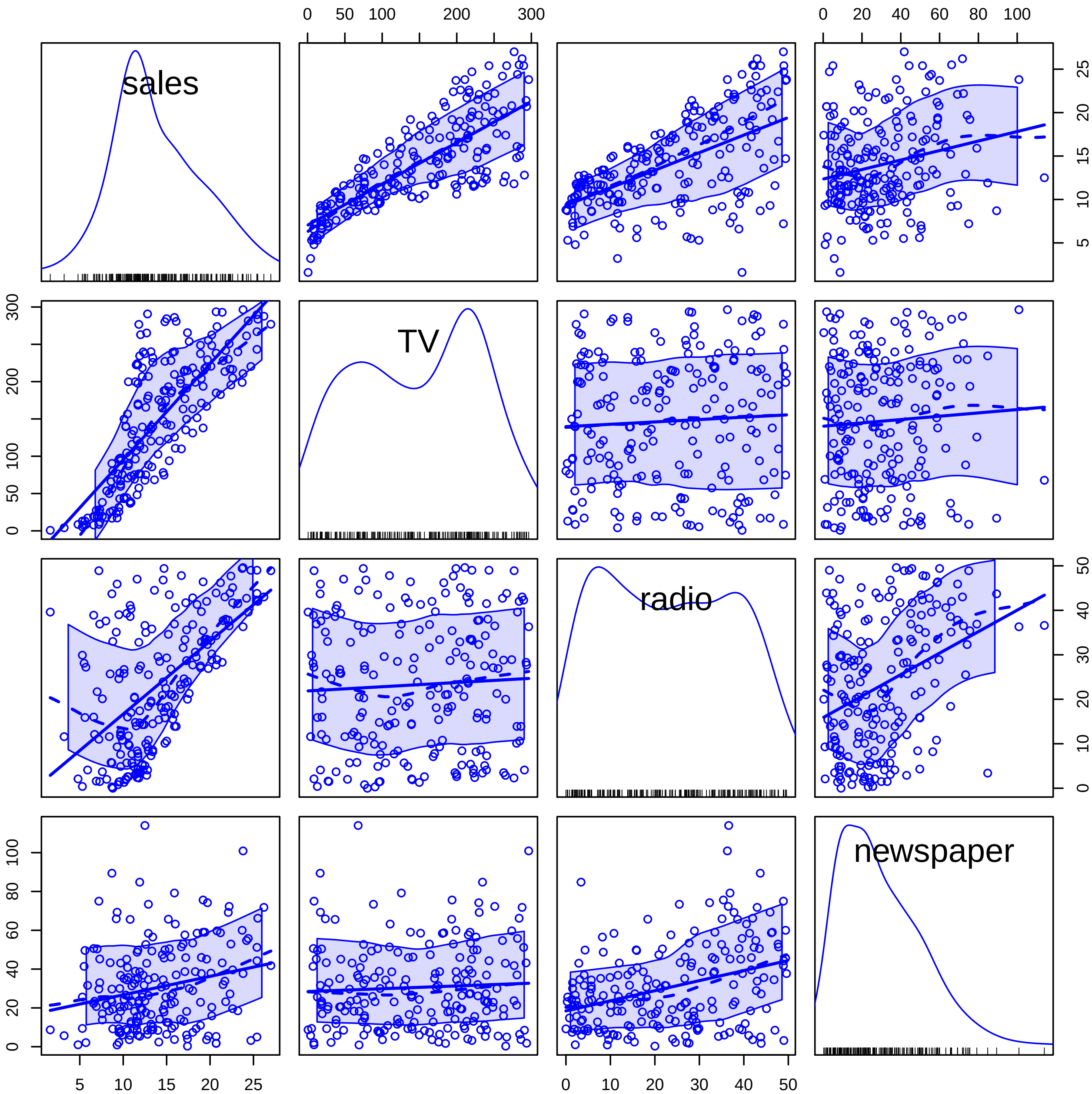
First create a plot with `ggplot2`, then pass the `ggplot2` object to `ggplotly` from the `plotly` package.

```
library(plotly)
p11 <- ggplotly(p)
p11
```



1.4 Scatterplot Matrices

```r
library(car)
scatterplotMatrix(~ sales + TV + radio + newspaper, data = AD)
```



2 Basic Regression

Recall mod1

```r
mod1 <- lm(sales ~ TV, data = AD)
summary(mod1)
```

```
Call:
lm(formula = sales ~ TV, data = AD)

Residuals:
    Min      1Q  Median      3Q     Max
-8.3860 -1.9545 -0.1913  2.0671  7.2124

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 7.032594   0.457843   15.36   <2e-16 ***
TV          0.047537   0.002691   17.67   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.259 on 198 degrees of freedom
Multiple R-squared:  0.6119,    Adjusted R-squared:  0.6099
F-statistic: 312.1 on 1 and 198 DF,  p-value: < 2.2e-16
```

$$\text{Residual} \equiv e_i = y_i - \hat{y}_i \tag{1}$$

To obtain the residuals for `mod1` use the function `resid` on a linear model object.

```
eis <- resid(mod1)
RSS <- sum(eis^2)
RSS
```

```
[1] 2102.531
```

```
RSE <- sqrt(RSS/(dim(AD)[1]-2))
RSE
```

```
[1] 3.258656
```

```
# Or
summary(mod1)$sigma
```

```
[1] 3.258656
```

```
# Or
library(broom)
NDF <- augment(mod1)
sum(NDF$.resid^2)
```

```
[1] 2102.531
```

```
RSE <- sqrt(sum(NDF$.resid^2)/df.residual(mod1))
RSE
```

```
[1] 3.258656
```

```
library(moderndive)
get_regression_table(mod1)
```

```
# A tibble: 2 x 7
  term      estimate std_error statistic p_value lower_ci upper_ci
  <chr>        <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
1 intercept    7.03      0.458      15.4       0    6.13     7.94
2 TV           0.048     0.003      17.7       0    0.042    0.053
```

```r
MDDF <- get_regression_points(mod1)
MDDF
```

```
# A tibble: 200 x 5
      ID sales    TV sales_hat residual
   <int> <dbl> <dbl>     <dbl>    <dbl>
 1     1  22.1 230.       18.0     4.13
 2     2  10.4  44.5      9.15     1.25
 3     3   9.3  17.2      7.85     1.45
 4     4  18.5 152.       14.2     4.27
 5     5  12.9 181.       15.6    -2.73
 6     6   7.2   8.7      7.45   -0.246
 7     7  11.8  57.5      9.77     2.03
 8     8  13.2 120.       12.7     0.454
 9     9   4.8   8.6      7.44   -2.64
10    10  10.6 200.       16.5    -5.93
# ... with 190 more rows
```

```r
library(dplyr)
MDDF %>%
  summarize(RSS = sum(residual^2))
```

```
# A tibble: 1 x 1
    RSS
  <dbl>
1 2102.
```

The least squares estimators of β_0 and β_1 are

$$b_0 = \hat{\beta}_0 = \bar{y} - b_1\bar{x}$$
$$b_1 = \hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

```r
y <- AD$sales
x <- AD$TV
b1 <- sum( (x - mean(x))*(y - mean(y)) ) / sum((x - mean(x))^2)
b0 <- mean(y) - b1*mean(x)
c(b0, b1)
```

```
[1] 7.03259355 0.04753664
```

```r
# Or using
coef(mod1)
```

```
(Intercept)          TV 
 7.03259355  0.04753664 
```

```r
summary(mod1)
```

```
Call:
lm(formula = sales ~ TV, data = AD)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.3860 -1.9545 -0.1913  2.0671  7.2124 
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 7.032594   0.457843   15.36   <2e-16 ***
TV          0.047537   0.002691   17.67   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.259 on 198 degrees of freedom
Multiple R-squared:  0.6119,    Adjusted R-squared:  0.6099
F-statistic: 312.1 on 1 and 198 DF,  p-value: < 2.2e-16
```

```
XTXI <- summary(mod1)$cov.unscaled
MSE <- summary(mod1)$sigma^2
var.cov.b <- MSE*XTXI
var.cov.b
```

```
             (Intercept)            TV
(Intercept)  0.209620158 -1.064495e-03
TV          -0.001064495  7.239367e-06
```

```
seb0 <- sqrt(var.cov.b[1, 1])
seb1 <- sqrt(var.cov.b[2, 2])
c(seb0, seb1)
```

```
[1] 0.457842940 0.002690607
```

```
coef(summary(mod1))
```

```
              Estimate  Std. Error  t value     Pr(>|t|)
(Intercept) 7.03259355 0.457842940 15.36028 1.40630e-35
TV          0.04753664 0.002690607 17.66763 1.46739e-42
```

```
coef(summary(mod1))[1, 2]
```

```
[1] 0.4578429
```

```
coef(summary(mod1))[2, 2]
```

```
[1] 0.002690607
```

```
tb0 <- b0/seb0
tb1 <- b1/seb1
c(tb0, tb1)
```

```
[1] 15.36028 17.66763
```

```
pvalues <- c(pt(tb0, 198, lower = FALSE)*2, pt(tb1, 198, lower = FALSE)*2)
pvalues
```

```
[1] 1.40630e-35 1.46739e-42
```

```
coef(summary(mod1))
```

```
              Estimate  Std. Error  t value     Pr(>|t|)
(Intercept) 7.03259355 0.457842940 15.36028 1.40630e-35
TV          0.04753664 0.002690607 17.66763 1.46739e-42
```

```
TSS <- sum((y - mean(y))^2)
c(RSS, TSS)
```

```
[1] 2102.531 5417.149
```

```r
R2 <- (TSS - RSS)/TSS
R2
```

```
[1] 0.6118751
```

```r
# Or
summary(mod1)$r.squared
```

```
[1] 0.6118751
```

2.1 Confidence Interval for β_1

$$\text{CI}_{1-\alpha}(\beta_1) = \left[b_1 - t_{1-\alpha/2, n-p+1} SE(b1), b_1 + t_{1-\alpha/2, n-p+1} SE(b1)\right] \tag{2}$$

Example: Use Equation (2) to construct a 90% confidence interval for β_1.

```r
alpha <- 0.10
ct <- qt(1 - alpha/2, df.residual(mod1))
ct
```

```
[1] 1.652586
```

```r
b1 + c(-1, 1)*ct*seb1
```

```
[1] 0.04309018 0.05198310
```

```r
# Or
confint(mod1, parm = "TV", level = 0.90)
```

```
          5 %      95 %
TV 0.04309018 0.0519831
```

```r
confint(mod1)
```

```
                 2.5 %     97.5 %
(Intercept) 6.12971927 7.93546783
TV          0.04223072 0.05284256
```

2.1.1 Linear Algebra

Solution of linear systems Find the solution(s) if any to the following linear equations.

$$\begin{aligned} 2x + y - z &= 8 \\ -3x - y + 2z &= -11 \\ -2x + y + 2z &= -3 \end{aligned}$$

```r
A <- matrix(c(2, -3, -2, 1, -1, 1, -1, 2, 2), nrow = 3)
b <- matrix(c(8, -11, -3), nrow = 3)
x <- solve(A)%*%b
x
```

```
     [,1]
[1,]    2
[2,]    3
[3,]   -1
```

```r
# Or
solve(A, b)
```

```
     [,1]
[1,]    2
[2,]    3
[3,]   -1
```

See wikipedia for a review of matrix multiplication rules and properties.

Consider the 2×2 matrix A.

$$A = \begin{bmatrix} 2 & 4 \\ 9 & 5 \end{bmatrix}$$

2.1.2 Linear Regression Matrix Notation

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y} \tag{3}$$

$$\sigma^2_{\hat{\beta}} = \sigma^2(\mathbf{X}'\mathbf{X})^{-1} \tag{4}$$

$$\hat{\sigma}^2_{\hat{\beta}} = MSE(\mathbf{X}'\mathbf{X})^{-1} \tag{5}$$

$$\hat{\beta} \sim \mathcal{N}(\beta, \sigma^2(\mathbf{X}'\mathbf{X})^{-1})$$

2.1.3 Estimation of the Mean Response for New Values X_h

Not only is it desirable to create confidence intervals on the parameters of the regression models, bit it is also common to estimate the mean response $(E(Y_h))$ for a particular set of $\mathbf{X}$ values.

$$\hat{Y}_h \sim \mathcal{N}(Y_h = X_h\beta, \sigma^2\mathbf{X_h}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X_h'})$$

For a vector of given values $(\mathbf{X_h})$, a $(1-\alpha)\cdot 100\%$ confidence interval for the mean response $E(Y_h)$ is

$$CI_{1-\alpha}\left[E(Y_h)\right] = \left[\hat{Y}_h - t_{1-\alpha/2;n-p-1}\cdot s_{\hat{Y}_h}, \hat{Y}_h + t_{1-\alpha/2;n-p-1}\cdot s_{\hat{Y}_h}\right]$$

The function `predict()` applied to a linear model object will compute $\hat{Y}_h$ and $s_{\hat{Y}_h}$ for a given $\mathbf{X}_h$. `R` output has $\hat{Y}_h$ labeled `fit` and $s_{\hat{Y}_h}$ labeled `se.fit`.

```r
A <- matrix(c(2, 9, 4, 5), nrow = 2)
A
```

```
     [,1] [,2]
[1,]    2    4
[2,]    9    5
```

```r
t(A)           # Transpose of A
```

```
     [,1] [,2]
[1,]    2    9
[2,]    4    5
```

```r
t(A)%*%A      # A'A
```

```
     [,1] [,2]
[1,]   85   53
[2,]   53   41
```

```r
solve(A)%*%A  # I_2
```

```
             [,1]          [,2]
[1,] 1.000000e+00 -1.110223e-16
[2,] 1.110223e-16  1.000000e+00
```

```r
zapsmall(solve(A)%*%A)  # What you expect I_2
```

```
     [,1] [,2]
[1,]    1    0
[2,]    0    1
```

```r
X <- model.matrix(mod1)
XTX <- t(X)%*%X
dim(XTX)
```

```
[1] 2 2
```

```r
XTXI <- solve(XTX)
XTXI
```

```
              (Intercept)            TV
(Intercept)  0.0197403984 -1.002458e-04
TV          -0.0001002458  6.817474e-07
```

```r
# But it is best to compute this quantity using
summary(mod1)$cov.unscaled
```

```
              (Intercept)            TV
(Intercept)  0.0197403984 -1.002458e-04
TV          -0.0001002458  6.817474e-07
```

```r
betahat <- XTXI%*%t(X)%*%y
betahat
```

```
                  [,1]
(Intercept) 7.03259355
TV          0.04753664
```

```r
coef(mod1)
```

```
(Intercept)          TV
 7.03259355  0.04753664
```

```r
XTXI <- summary(mod1)$cov.unscaled
MSE <- summary(mod1)$sigma^2
var_cov_b <- MSE*XTXI
var_cov_b
```

```
             (Intercept)            TV
(Intercept)  0.209620158 -1.064495e-03
TV          -0.001064495  7.239367e-06
```

Example Use the `GRADES` data set and model `gpa` as a function of `sat`. Compute the expected GPA (`gpa`) for an SAT score (`sat`) of 1300. Construct a 90% confidence interval for the mean GPA for students scoring

1300 on the SAT.

```
library(PASWR2)
mod.lm <- lm(gpa ~ sat, data = GRADES)
summary(mod.lm)
```

```

Call:
lm(formula = gpa ~ sat, data = GRADES)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.04954 -0.25960 -0.00655  0.26044  1.09328 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -1.1920638  0.2224502  -5.359 2.32e-07 ***
sat          0.0030943  0.0001945  15.912  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3994 on 198 degrees of freedom
Multiple R-squared:  0.5612,    Adjusted R-squared:  0.5589 
F-statistic: 253.2 on 1 and 198 DF,  p-value: < 2.2e-16
```

```
betahat <- coef(mod.lm)
betahat
```

```
(Intercept)         sat 
-1.19206381  0.00309427 
```

```
knitr::kable(tidy(mod.lm))
```

term	estimate	std.error	statistic	p.value
(Intercept)	-1.1920638	0.2224502	-5.35879	2e-07
sat	0.0030943	0.0001945	15.91171	0e+00

```
#
Xh <- matrix(c(1, 1300), nrow = 1)
Yhath <- Xh%*%betahat
Yhath
```

```
         [,1]
[1,] 2.830488
```

```
predict(mod.lm, newdata = data.frame(sat = 1300))
```

```
       1 
2.830488 
```

```
# Linear Algebra First
anova(mod.lm)
```

```
Analysis of Variance Table

Response: gpa
           Df Sum Sq Mean Sq F value    Pr(>F)    
sat         1 40.397  40.397  253.18 < 2.2e-16 ***
Residuals 198 31.592   0.160                      
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
MSE <- anova(mod.lm)[2, 3]
MSE
```

```
[1] 0.1595551
```

```
XTXI <- summary(mod.lm)$cov.unscaled
XTXI
```

```
             (Intercept)           sat
(Intercept)  0.310137964 -2.689270e-04
sat         -0.000268927  2.370131e-07
```

```
var_cov_b <- MSE*XTXI
var_cov_b
```

```
              (Intercept)           sat
(Intercept)  4.948408e-02 -4.290866e-05
sat         -4.290866e-05  3.781665e-08
```

```
s2yhath <- Xh %*% var_cov_b %*% t(Xh)
s2yhath
```

```
            [,1]
[1,] 0.001831706
```

```
syhath <- sqrt(s2yhath)
syhath
```

```
           [,1]
[1,] 0.04279843
```

```
crit_t <- qt(0.95, df.residual(mod.lm))
crit_t
```

```
[1] 1.652586
```

```
CI_EYh <- c(Yhath) + c(-1, 1)*c(crit_t*syhath)
CI_EYh
```

```
[1] 2.759760 2.901216
```

```
# Using the build in function
predict(mod.lm, newdata = data.frame(sat = 1300), interval = "conf", level = 0.90)
```

```
       fit     lwr      upr
1 2.830488 2.75976 2.901216
```

2.2 Multiple Linear Regression

```
mod2 <- lm(sales ~ TV + radio, data = AD)
summary(mod2)
```

```
Call:
lm(formula = sales ~ TV + radio, data = AD)

Residuals:
    Min      1Q  Median      3Q     Max
```

```
-8.7977 -0.8752  0.2422  1.1708  2.8328

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.92110    0.29449   9.919   <2e-16 ***
TV           0.04575    0.00139  32.909   <2e-16 ***
radio        0.18799    0.00804  23.382   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.681 on 197 degrees of freedom
Multiple R-squared:  0.8972,    Adjusted R-squared:  0.8962
F-statistic: 859.6 on 2 and 197 DF,  p-value: < 2.2e-16
```

2.2.1 Graphing the plane

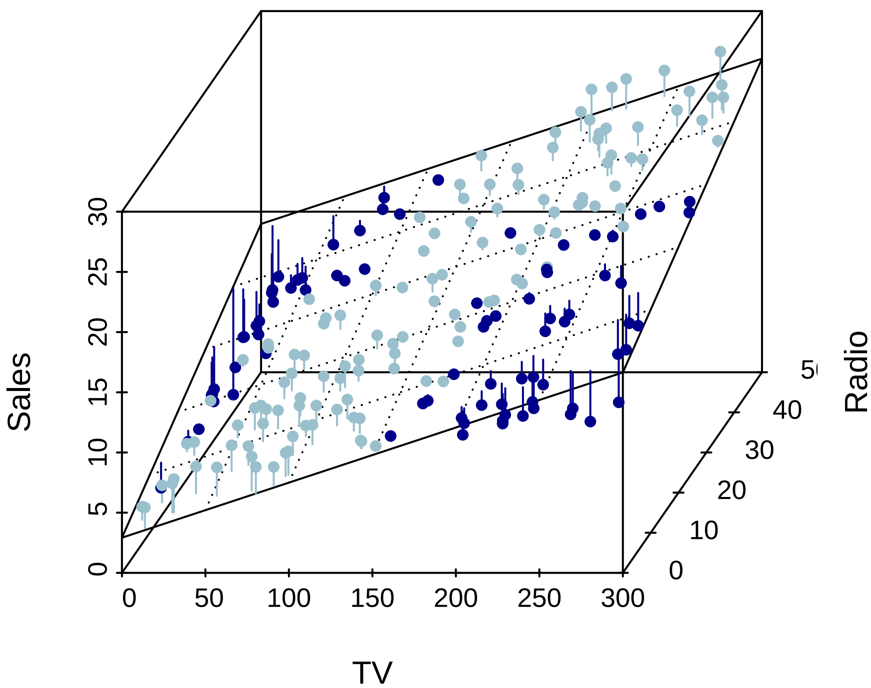


Figure 4: 3-D residuals and fitted plane

2.2.2 Using plotly

```r
library(plotly)
# draw the 3D scatterplot
p <- plot_ly(data = AD, z = ~sales, x = ~TV, y = ~radio, opacity = 0.5) %>%
  add_markers
p
```

WebGL is not supported by your browser - visit https://get.webgl.org for more info

```
x <- seq(0, 300, length = 70)
y <- seq(0, 50, length = 70)
plane <- outer(x, y, function(a, b){summary(mod2)$coef[1, 1] + summary(mod2)$coef[2, 1]*a + summary(mod2
# draw the plane
p %>%
  add_surface(x = ~x, y = ~y, z = ~plane, showscale = FALSE)
```

2.3 Is There a Relationship Between the Response and Predictors?

```r
mod3 <- lm(sales ~ TV + radio + newspaper, data = AD)
summary(mod3)
```

```

Call:
lm(formula = sales ~ TV + radio + newspaper, data = AD)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.8277 -0.8908  0.2418  1.1893  2.8292 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.938889   0.311908   9.422   <2e-16 ***
TV           0.045765   0.001395  32.809   <2e-16 ***
radio        0.188530   0.008611  21.893   <2e-16 ***
newspaper   -0.001037   0.005871  -0.177     0.86    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 1.686 on 196 degrees of freedom
Multiple R-squared:  0.8972,    Adjusted R-squared:  0.8956
F-statistic: 570.3 on 3 and 196 DF,  p-value: < 2.2e-16
```

$$H_0 : \beta_1 = \beta_2 = \beta_3 = 0$$

versus the alternative

$$H_1 : \text{at least one } \beta_j \neq 0$$

The test statistic is $F = \frac{(\text{TSS}-\text{RSS})/p}{\text{RSS}/(n-p-1)}$

```r
anova(mod3)
```

```
Analysis of Variance Table

Response: sales
           Df Sum Sq Mean Sq   F value Pr(>F)
TV          1 3314.6  3314.6 1166.7308 <2e-16 ***
radio       1 1545.6  1545.6  544.0501 <2e-16 ***
newspaper   1    0.1     0.1    0.0312 0.8599
Residuals 196  556.8     2.8
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
SSR <- sum(anova(mod3)[1:3, 2])
MSR <- SSR/3
SSE <- anova(mod3)[4, 2]
MSE <- SSE/(200-3-1)
Fobs <- MSR/MSE
Fobs
```

```
[1] 570.2707
```

```r
pvalue <- pf(Fobs, 3, 196, lower = FALSE)
pvalue
```

```
[1] 1.575227e-96
```

```r
# Or
summary(mod3)
```

```
Call:
lm(formula = sales ~ TV + radio + newspaper, data = AD)

Residuals:
    Min      1Q  Median      3Q     Max
-8.8277 -0.8908  0.2418  1.1893  2.8292

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.938889   0.311908   9.422   <2e-16 ***
TV           0.045765   0.001395  32.809   <2e-16 ***
radio        0.188530   0.008611  21.893   <2e-16 ***
newspaper   -0.001037   0.005871  -0.177     0.86
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 1.686 on 196 degrees of freedom
Multiple R-squared:  0.8972,	Adjusted R-squared:  0.8956 
F-statistic: 570.3 on 3 and 196 DF,  p-value: < 2.2e-16
```

```r
summary(mod3)$fstatistic
```

```
   value    numdf    dendf 
570.2707   3.0000 196.0000 
```

Suppose we would like to test whether $\beta_2 = \beta_3 = 0$. The reduced model with $\beta_2 = \beta_3 = 0$ is `mod1` while the full model is `mod3`.

```r
summary(mod3)
```

```
Call:
lm(formula = sales ~ TV + radio + newspaper, data = AD)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.8277 -0.8908  0.2418  1.1893  2.8292 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.938889   0.311908   9.422   <2e-16 ***
TV           0.045765   0.001395  32.809   <2e-16 ***
radio        0.188530   0.008611  21.893   <2e-16 ***
newspaper   -0.001037   0.005871  -0.177     0.86    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.686 on 196 degrees of freedom
Multiple R-squared:  0.8972,	Adjusted R-squared:  0.8956 
F-statistic: 570.3 on 3 and 196 DF,  p-value: < 2.2e-16
```

```r
anova(mod1, mod3)
```

```
Analysis of Variance Table

Model 1: sales ~ TV
Model 2: sales ~ TV + radio + newspaper
  Res.Df     RSS Df Sum of Sq      F    Pr(>F)    
1    198 2102.53                                  
2    196  556.83  2    1545.7 272.04 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

2.4 Variable Selection

- Forward selection

```r
mod.fs <- lm(sales ~ 1, data = AD)
SCOPE <- (~ TV + radio + newspaper)
add1(mod.fs, scope = SCOPE, test = "F")
```

```
Single term additions

Model:
```

```
sales ~ 1
          Df Sum of Sq    RSS    AIC F value    Pr(>F)
<none>                 5417.1 661.80
TV         1    3314.6 2102.5 474.52 312.145 < 2.2e-16 ***
radio      1    1798.7 3618.5 583.10  98.422 < 2.2e-16 ***
newspaper  1     282.3 5134.8 653.10  10.887  0.001148 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
mod.fs <- update(mod.fs, .~. + TV)
add1(mod.fs, scope = SCOPE, test = "F")
```

```
Single term additions

Model:
sales ~ TV
          Df Sum of Sq     RSS    AIC F value    Pr(>F)
<none>                 2102.53 474.52
radio      1   1545.62  556.91 210.82  546.74 < 2.2e-16 ***
newspaper  1    183.97 1918.56 458.20   18.89 2.217e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
mod.fs <- update(mod.fs, .~. + radio)
add1(mod.fs, scope = SCOPE, test = "F")
```

```
Single term additions

Model:
sales ~ TV + radio
          Df Sum of Sq    RSS    AIC F value Pr(>F)
<none>                 556.91 210.82
newspaper  1  0.088717 556.83 212.79  0.0312 0.8599
```

```r
summary(mod.fs)
```

```
Call:
lm(formula = sales ~ TV + radio, data = AD)

Residuals:
    Min      1Q  Median      3Q     Max
-8.7977 -0.8752  0.2422  1.1708  2.8328

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.92110    0.29449   9.919   <2e-16 ***
TV           0.04575    0.00139  32.909   <2e-16 ***
radio        0.18799    0.00804  23.382   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.681 on 197 degrees of freedom
Multiple R-squared:  0.8972,	Adjusted R-squared:  0.8962
F-statistic: 859.6 on 2 and 197 DF,  p-value: < 2.2e-16
```

- Using `stepAIC`

```r
stepAIC(lm(sales ~ 1, data = AD), scope = (~TV + radio + newspaper), direction = "forward", test = "F")
```

```
Start:  AIC=661.8
sales ~ 1

            Df Sum of Sq    RSS    AIC F Value     Pr(F)    
+ TV         1    3314.6 2102.5 474.52 312.145 < 2.2e-16 ***
+ radio      1    1798.7 3618.5 583.10  98.422 < 2.2e-16 ***
+ newspaper  1     282.3 5134.8 653.10  10.887  0.001148 ** 
<none>                   5417.1 661.80                      
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=474.52
sales ~ TV

            Df Sum of Sq     RSS    AIC F Value     Pr(F)    
+ radio      1   1545.62  556.91 210.82  546.74 < 2.2e-16 ***
+ newspaper  1    183.97 1918.56 458.20   18.89 2.217e-05 ***
<none>                   2102.53 474.52                      
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=210.82
sales ~ TV + radio

            Df Sum of Sq    RSS    AIC  F Value  Pr(F)
<none>                   556.91 210.82                
+ newspaper  1  0.088717 556.83 212.79 0.031228 0.8599

Call:
lm(formula = sales ~ TV + radio, data = AD)

Coefficients:
(Intercept)           TV        radio  
    2.92110      0.04575      0.18799  
```

```r
# Or
null <- lm(sales ~ 1, data = AD)
full <- lm(sales ~ ., data = AD)
stepAIC(null, scope = list(lower = null, upper = full), direction = "forward", test = "F")
```

```
Start:  AIC=661.8
sales ~ 1

            Df Sum of Sq    RSS    AIC F Value     Pr(F)    
+ TV         1    3314.6 2102.5 474.52 312.145 < 2.2e-16 ***
+ radio      1    1798.7 3618.5 583.10  98.422 < 2.2e-16 ***
+ newspaper  1     282.3 5134.8 653.10  10.887  0.001148 ** 
<none>                   5417.1 661.80                      
+ X          1      14.4 5402.7 663.27   0.529  0.467917    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=474.52
```

```
sales ~ TV

            Df Sum of Sq     RSS    AIC F Value     Pr(F)
+ radio      1   1545.62  556.91 210.82  546.74 < 2.2e-16 ***
+ newspaper  1    183.97 1918.56 458.20   18.89 2.217e-05 ***
+ X          1     23.23 2079.30 474.29    2.20    0.1395
<none>                   2102.53 474.52
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=210.82
sales ~ TV + radio

            Df Sum of Sq    RSS    AIC  F Value  Pr(F)
<none>                   556.91 210.82
+ X          1  0.181080 556.73 212.75 0.063750 0.8009
+ newspaper  1  0.088717 556.83 212.79 0.031228 0.8599
```

```
Call:
lm(formula = sales ~ TV + radio, data = AD)

Coefficients:
(Intercept)           TV        radio
    2.92110      0.04575      0.18799
```

- Backward elimination

```r
mod.be <- lm(sales ~ TV + radio + newspaper, data = AD)
drop1(mod.be, test = "F")
```

```
Single term deletions

Model:
sales ~ TV + radio + newspaper
          Df Sum of Sq    RSS    AIC   F value Pr(>F)
<none>                  556.8 212.79
TV         1   3058.01 3614.8 584.90 1076.4058 <2e-16 ***
radio      1   1361.74 1918.6 458.20  479.3252 <2e-16 ***
newspaper  1      0.09  556.9 210.82    0.0312 0.8599
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
mod.be <- update(mod.be, .~. - newspaper)
drop1(mod.be, test = "F")
```

```
Single term deletions

Model:
sales ~ TV + radio
       Df Sum of Sq    RSS    AIC F value    Pr(>F)
<none>                556.9 210.82
TV      1    3061.6 3618.5 583.10 1082.98 < 2.2e-16 ***
radio   1    1545.6 2102.5 474.52  546.74 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(mod.be)
```

```
Call:
lm(formula = sales ~ TV + radio, data = AD)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.7977 -0.8752  0.2422  1.1708  2.8328 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.92110    0.29449   9.919   <2e-16 ***
TV           0.04575    0.00139  32.909   <2e-16 ***
radio        0.18799    0.00804  23.382   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.681 on 197 degrees of freedom
Multiple R-squared:  0.8972,    Adjusted R-squared:  0.8962 
F-statistic: 859.6 on 2 and 197 DF,  p-value: < 2.2e-16
```

- Using stepAIC

```r
stepAIC(lm(sales ~ TV + radio + newspaper, data = AD), scope = (~TV + radio + newspaper), direction = "
```

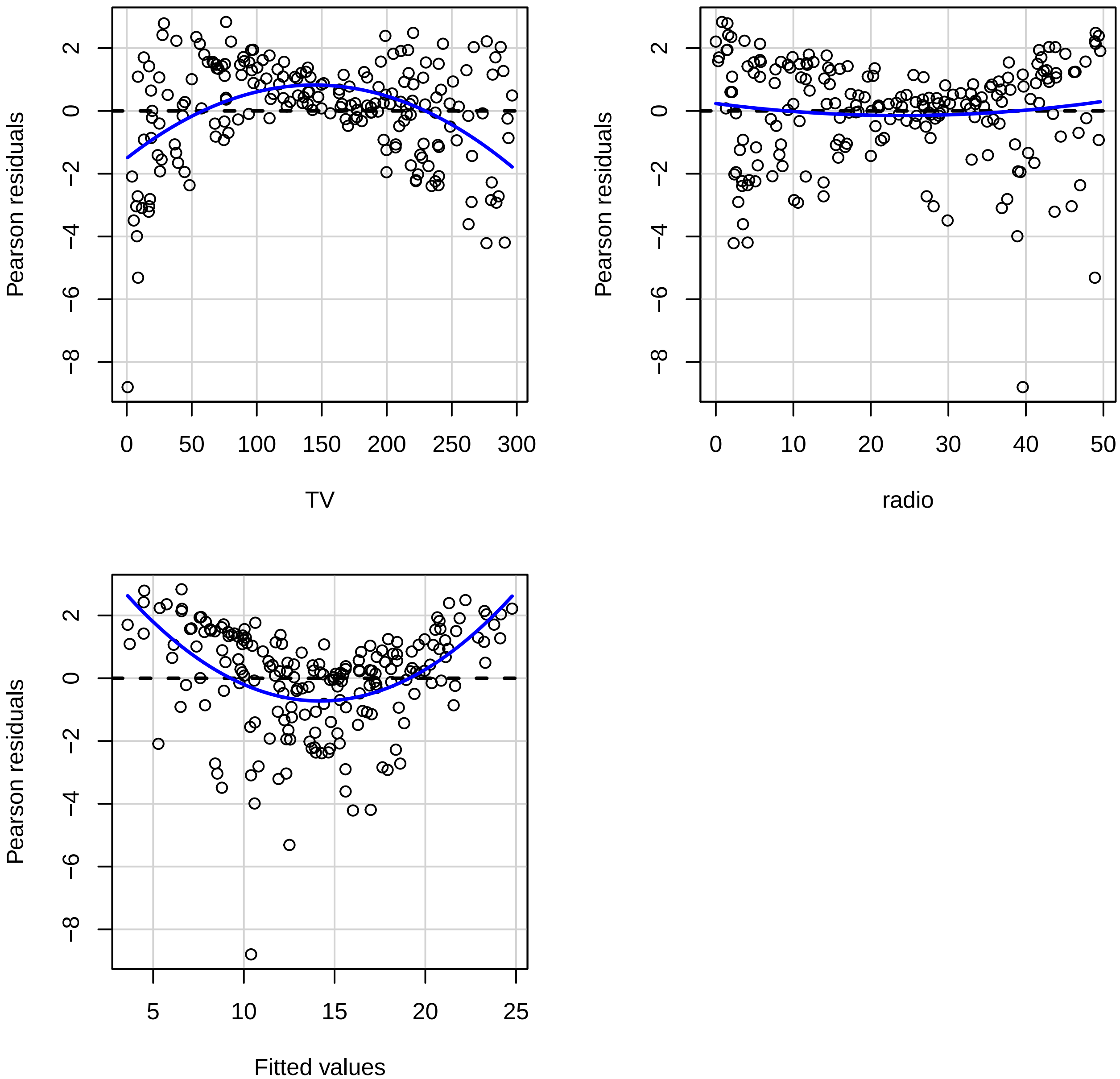
```
Start:  AIC=212.79
sales ~ TV + radio + newspaper

            Df Sum of Sq    RSS    AIC F Value  Pr(F)    
- newspaper  1      0.09  556.9 210.82    0.03 0.8599    
<none>                    556.8 212.79                   
- radio      1   1361.74 1918.6 458.20  479.33 <2e-16 ***
- TV         1   3058.01 3614.8 584.90 1076.41 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=210.82
sales ~ TV + radio

        Df Sum of Sq    RSS    AIC F Value     Pr(F)    
<none>                556.9 210.82                      
- radio  1    1545.6 2102.5 474.52  546.74 < 2.2e-16 ***
- TV     1    3061.6 3618.5 583.10 1082.98 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Call:
lm(formula = sales ~ TV + radio, data = AD)

Coefficients:
(Intercept)           TV        radio  
    2.92110      0.04575      0.18799  
```

```r
# Or
stepAIC(full, scope = list(lower = null, upper = full), direction = "backward", test = "F")
```

```
Start:  AIC=214.71
sales ~ X + TV + radio + newspaper

            Df Sum of Sq    RSS    AIC F Value  Pr(F)
- newspaper  1      0.13  556.7 212.75    0.04 0.8342
- X          1      0.22  556.8 212.79    0.08 0.7827
<none>                    556.6 214.71
- radio      1   1354.48 1911.1 459.42  474.52 <2e-16 ***
- TV         1   3056.91 3613.5 586.82 1070.95 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=212.75
sales ~ X + TV + radio

        Df Sum of Sq    RSS    AIC F Value  Pr(F)
- X      1      0.18  556.9 210.82    0.06 0.8009
<none>                556.7 212.75
- radio  1   1522.57 2079.3 474.29  536.03 <2e-16 ***
- TV     1   3060.94 3617.7 585.05 1077.61 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=210.82
sales ~ TV + radio

        Df Sum of Sq    RSS    AIC F Value     Pr(F)
<none>                556.9 210.82
- radio  1    1545.6 2102.5 474.52  546.74 < 2.2e-16 ***
- TV     1    3061.6 3618.5 583.10 1082.98 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Call:
lm(formula = sales ~ TV + radio, data = AD)

Coefficients:
(Intercept)           TV        radio
    2.92110      0.04575      0.18799
```

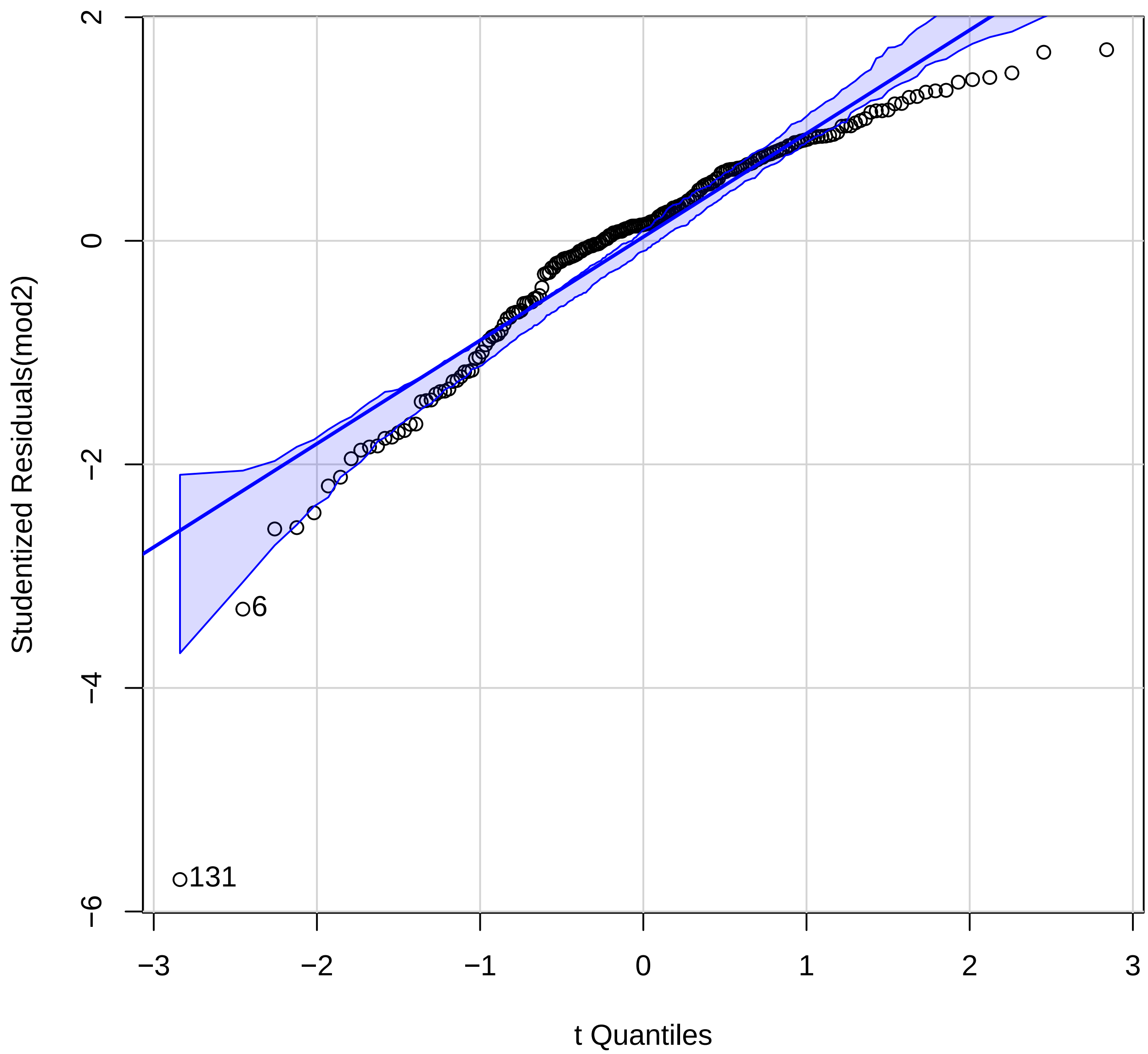
2.5 Diagnostic Plots

```
residualPlots(mod2)
```



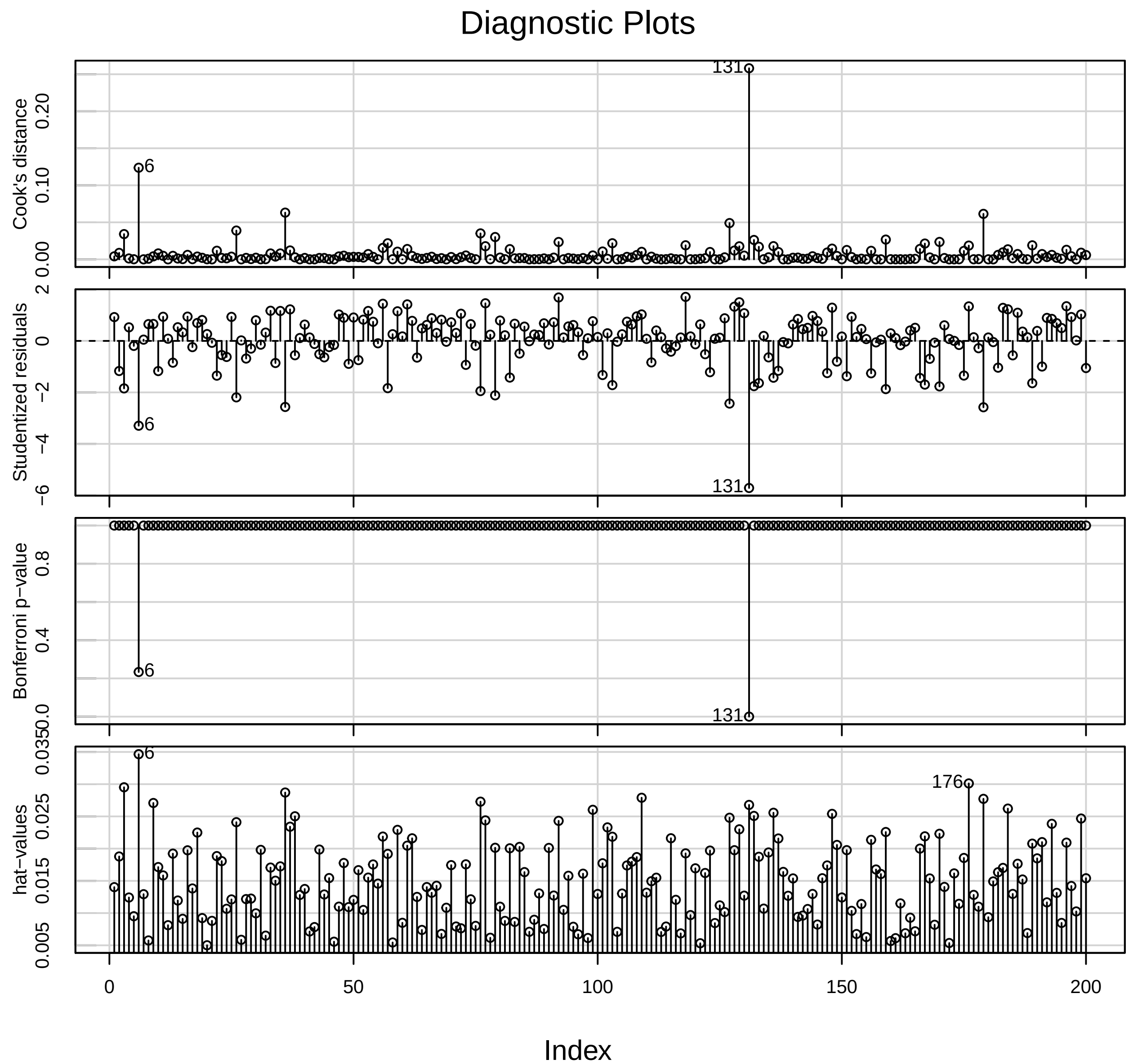

```
              Test stat Pr(>|Test stat|)
TV              -6.7745        1.423e-10 ***
radio            1.0543           0.2931
Tukey test       7.6351        2.256e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
qqPlot(mod2)
```




```
[1]   6 131
```

```
influenceIndexPlot(mod2)
```



We use a *confidence interval* to quantify the uncertainty surrounding the *average* `sales` over a large number of cities. For example, given that \$100,000 is spent on `TV` advertising and \$20,000 is spent on `Radio` advertising in each city, the 95% confidence interval is [10.9852544, 11.5276775]. We interpret this to mean that 95% of intervals of this form will contain the true value of `Sales`.

```
predict(mod.be, newdata = data.frame(TV = 100, radio = 20), interval = "conf")
```

```
       fit      lwr      upr
1 11.25647 10.98525 11.52768
```

On the other hand, a *prediction interval* can be used to quantify the uncertainty surrounding `sales` for a *particular* city. Given that \$100,000 is spent on `TV` advertising and \$20,000 is spent on `radio` advertising in **a particular** city, the 95% prediction interval is [7.9296161, 14.5833158]. We interpret this to mean that 95% of intervals of this form will contain the true value of `Sales` for this city.

```
predict(mod.be, newdata = data.frame(TV = 100, radio = 20), interval = "pred")
```

```
       fit      lwr      upr
```

```
1 11.25647 7.929616 14.58332
```

Note that both the intervals are centered at 11.256466, but that the prediction interval is substantially wider than the confidence interval, reflecting the increased uncertainty about `Sales` for a given city in comparison to the average `sales` over many locations.

2.6 Non-Additive Models

```r
nam1 <- lm(sales ~ TV*radio, data = AD)
# Same as
nam2 <- lm(sales ~ TV + radio + TV:radio, data = AD)
summary(nam1)
```

```

Call:
lm(formula = sales ~ TV * radio, data = AD)

Residuals:
    Min      1Q  Median      3Q     Max 
-6.3366 -0.4028  0.1831  0.5948  1.5246 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 6.750e+00  2.479e-01  27.233   <2e-16 ***
TV          1.910e-02  1.504e-03  12.699   <2e-16 ***
radio       2.886e-02  8.905e-03   3.241   0.0014 ** 
TV:radio    1.086e-03  5.242e-05  20.727   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9435 on 196 degrees of freedom
Multiple R-squared:  0.9678,	Adjusted R-squared:  0.9673 
F-statistic:  1963 on 3 and 196 DF,  p-value: < 2.2e-16
```

```r
summary(nam2)
```

```

Call:
lm(formula = sales ~ TV + radio + TV:radio, data = AD)

Residuals:
    Min      1Q  Median      3Q     Max 
-6.3366 -0.4028  0.1831  0.5948  1.5246 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 6.750e+00  2.479e-01  27.233   <2e-16 ***
TV          1.910e-02  1.504e-03  12.699   <2e-16 ***
radio       2.886e-02  8.905e-03   3.241   0.0014 ** 
TV:radio    1.086e-03  5.242e-05  20.727   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9435 on 196 degrees of freedom
Multiple R-squared:  0.9678,	Adjusted R-squared:  0.9673 
F-statistic:  1963 on 3 and 196 DF,  p-value: < 2.2e-16
```

Hierarchical Principle: If an interaction term is included in a model, one should also include the main effects, even if the *p-values* associated with their coefficients are not significant.

2.7 Qualitative Predictors

In the `Credit` data frame there are four qualitative features/variables `Gender`, `Student`, `Married`, and `Ethnicity`.

```
Credit <- read.csv("http://statlearning.com/s/Credit.csv")
datatable(Credit[, -1], rownames = FALSE)
```

Show 10 entries Search:

Limit	Rating	Cards	Age	Education	Own	Student	Married	Region	Balance
3606	283	2	34	11	No	No	Yes	South	333
6645	483	3	82	15	Yes	Yes	Yes	West	903
7075	514	4	71	11	No	No	No	West	580
9504	681	3	36	11	Yes	No	No	West	964
4897	357	2	68	16	No	No	Yes	South	331
8047	569	4	77	10	No	No	No	South	1151
3388	259	2	37	12	Yes	No	No	East	203
7114	512	2	87	9	No	No	No	West	872
3300	266	5	66	13	Yes	No	No	South	279
6819	491	3	41	19	Yes	Yes	Yes	East	1350

Showing 1 to 10 of 400 entries Previous 1 2 3 4 5 … 40 Next

```
modP <- lm(Balance ~ Income*Student, data = Credit)
summary(modP)
```

```

Call:
lm(formula = Balance ~ Income * Student, data = Credit)

Residuals:
    Min      1Q  Median      3Q     Max 
-773.39 -325.70  -41.13  321.65  814.04 

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)    
(Intercept)       200.6232    33.6984   5.953 5.79e-09 ***
Income              6.2182     0.5921  10.502  < 2e-16 ***
StudentYes        476.6758   104.3512   4.568 6.59e-06 ***
Income:StudentYes  -1.9992     1.7313  -1.155    0.249    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 391.6 on 396 degrees of freedom
Multiple R-squared:  0.2799,	Adjusted R-squared:  0.2744 
F-statistic:  51.3 on 3 and 396 DF,  p-value: < 2.2e-16
```

Fitted Model: $\widehat{\text{Balance}} = 200.6231529 + 6.2181687 \cdot \text{Income} + 476.6758432 \cdot \text{Student} - 1.9991509 \cdot \text{Income} \times \text{Student}$

2.7.1 Predictors with Only Two Levels

Suppose we wish to investigate differences in credit card balance between males and females, ignoring the other variables for the moment.

```
library(ISLR)
data(Credit)
modS <- lm(Balance ~ Gender, data = Credit)
summary(modS)
```

```

Call:
lm(formula = Balance ~ Gender, data = Credit)

Residuals:
    Min      1Q  Median      3Q     Max 
-529.54 -455.35  -60.17  334.71 1489.20 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)    509.80      33.13  15.389   <2e-16 ***
GenderFemale    19.73      46.05   0.429    0.669    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 460.2 on 398 degrees of freedom
Multiple R-squared:  0.0004611,	Adjusted R-squared:  -0.00205 
F-statistic: 0.1836 on 1 and 398 DF,  p-value: 0.6685
```

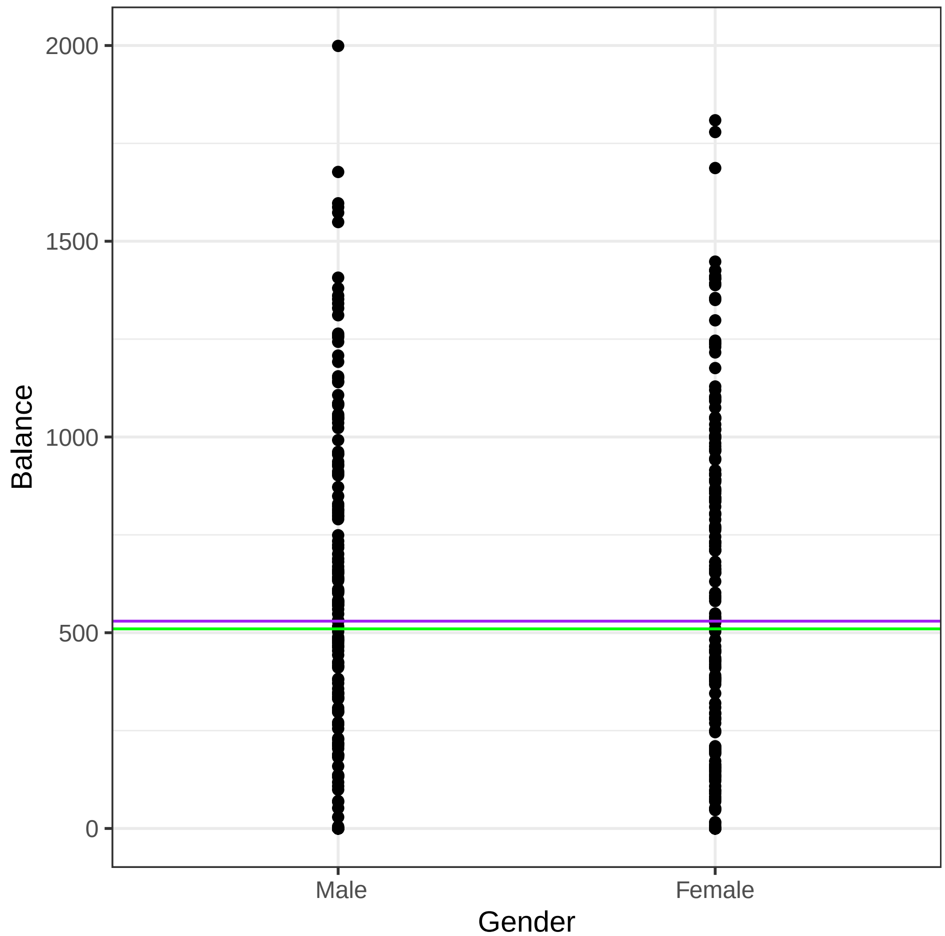
```
coef(modS)
```

```
 (Intercept) GenderFemale 
   509.80311     19.73312 
```

```
tapply(Credit$Balance, Credit$Gender, mean)
```

```
    Male   Female 
509.8031 529.5362 
```

```
library(ggplot2)
ggplot(data = Credit, aes(x = Gender, y = Balance)) +
  geom_point() +
  theme_bw() +
  geom_hline(yintercept = coef(modS)[1] + coef(modS)[2], color = "purple") +
  geom_hline(yintercept = coef(modS)[1], color = "green")
```



Do females have a higher ratio of `Balance` to `Income` (credit utilization)? Here is an article from the Washington Post with numbers that mirror some of the results in the `Credit` data set.

```
Credit$Utilization <- Credit$Balance / (Credit$Income*100)
tapply(Credit$Utilization, Credit$Gender, mean)
```

```
     Male    Female
0.1487092 0.1535206
```

```
# Tidyverse approach
Credit %>%
  mutate(Ratio = Balance / (Income*100) ) %>%
  group_by(Gender) %>%
  summarize(mean(Ratio))
```

```
# A tibble: 2 x 2
  Gender   `mean(Ratio)`
  <fct>            <dbl>
1 " Male"          0.149
2 "Female"         0.154
```

```
modU <- lm(Utilization ~ Gender, data = Credit)
summary(modU)
```

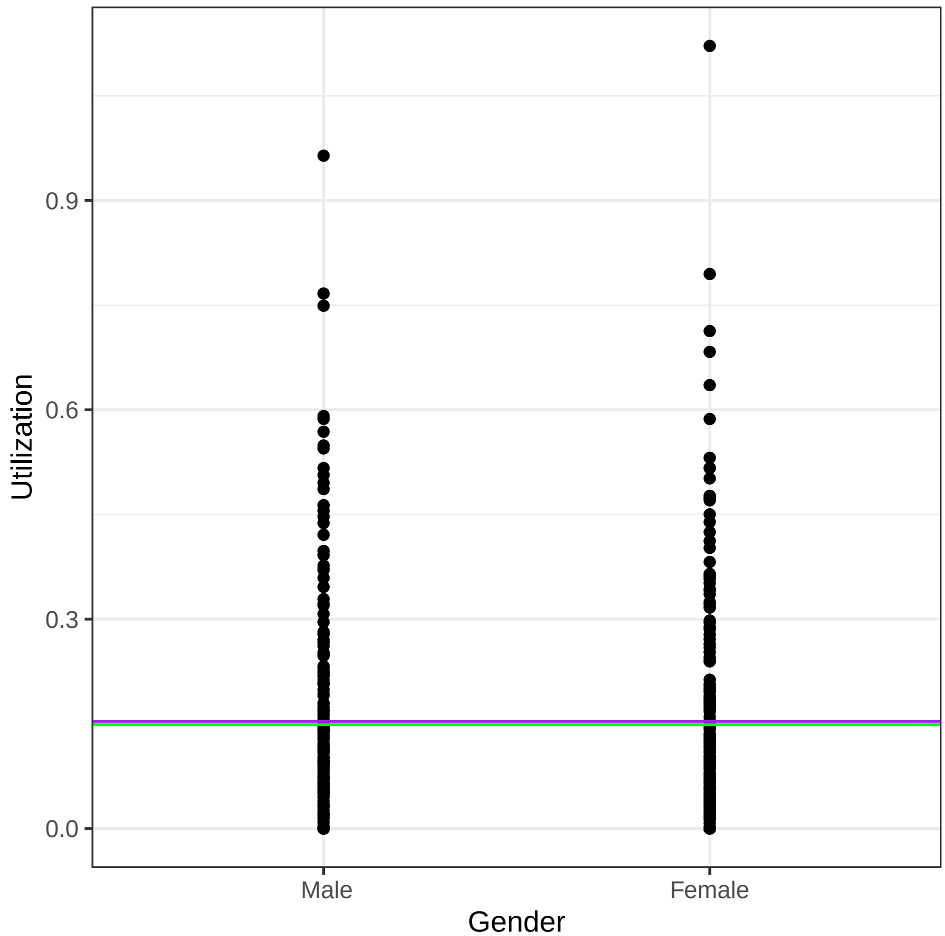
```
Call:
lm(formula = Utilization ~ Gender, data = Credit)
```

```
Residuals:
     Min       1Q   Median       3Q      Max 
-0.15352 -0.13494 -0.05202  0.06069  0.96804 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.148709   0.012388  12.004   <2e-16 ***
GenderFemale 0.004811   0.017221   0.279     0.78    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1721 on 398 degrees of freedom
Multiple R-squared:  0.0001961, Adjusted R-squared:  -0.002316 
F-statistic: 0.07806 on 1 and 398 DF,  p-value: 0.7801
```

```
coef(modU)
```

```
 (Intercept) GenderFemale 
 0.148709165  0.004811408 
```

```
ggplot(data = Credit, aes(x = Gender, y = Utilization)) +
  geom_point() +
  theme_bw() +
  geom_hline(yintercept = coef(modU)[1] + coef(modU)[2], color = "purple") +
  geom_hline(yintercept = coef(modU)[1], color = "green")
```



2.8 Moving On Now

```
modS1 <- lm(Balance ~ Limit + Student, data = Credit)
summary(modS1)
```

```
Call:
lm(formula = Balance ~ Limit + Student, data = Credit)

Residuals:
    Min      1Q  Median      3Q     Max
-637.77 -116.90    6.04  130.92  434.24

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -3.347e+02  2.307e+01  -14.51   <2e-16 ***
Limit        1.720e-01  4.331e-03   39.70   <2e-16 ***
StudentYes   4.044e+02  3.328e+01   12.15   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 199.7 on 397 degrees of freedom
Multiple R-squared:  0.8123,    Adjusted R-squared:  0.8114
F-statistic: 859.2 on 2 and 397 DF,  p-value: < 2.2e-16
```

```r
coef(modS1)
```

```
 (Intercept)        Limit   StudentYes 
-334.7299372    0.1719538  404.4036438 
```

```r
# Interaction --- Non-additive Model
modS2 <- lm(Balance ~ Limit*Student, data = Credit)
summary(modS2)
```

```

Call:
lm(formula = Balance ~ Limit * Student, data = Credit)

Residuals:
    Min      1Q  Median      3Q     Max 
-705.84 -116.90    6.91  133.97  435.92 

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)    
(Intercept)      -3.262e+02  2.392e+01 -13.636  < 2e-16 ***
Limit             1.702e-01  4.533e-03  37.538  < 2e-16 ***
StudentYes        3.091e+02  7.878e+01   3.924 0.000103 ***
Limit:StudentYes  2.028e-02  1.520e-02   1.334 0.183010    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 199.5 on 396 degrees of freedom
Multiple R-squared:  0.8132,	Adjusted R-squared:  0.8118 
F-statistic: 574.5 on 3 and 396 DF,  p-value: < 2.2e-16
```

2.8.1 What does this look like?

Several points:

- Is the interaction significant?
- Which model is `ggplot2` graphing below?
- Is this the correct model?

```r
ggplot(data = Credit, aes(x = Limit, y = Balance, color = Student)) +
  geom_point() +
  stat_smooth(method = "lm") +
  theme_bw()
```

2.8.2 Correct Graph

```r
S2M <- lm(Balance ~ Limit + Student, data = Credit)
#
ggplot(data = Credit, aes(x = Limit, y = Balance, color = Student)) +
  geom_point() +
  theme_bw() +
  geom_abline(intercept = coef(S2M)[1], slope = coef(S2M)[2], color = "red") +
  geom_abline(intercept = coef(S2M)[1] + coef(S2M)[3], slope = coef(S2M)[2], color = "blue") +
  scale_color_manual(values = c("red", "blue"))
```

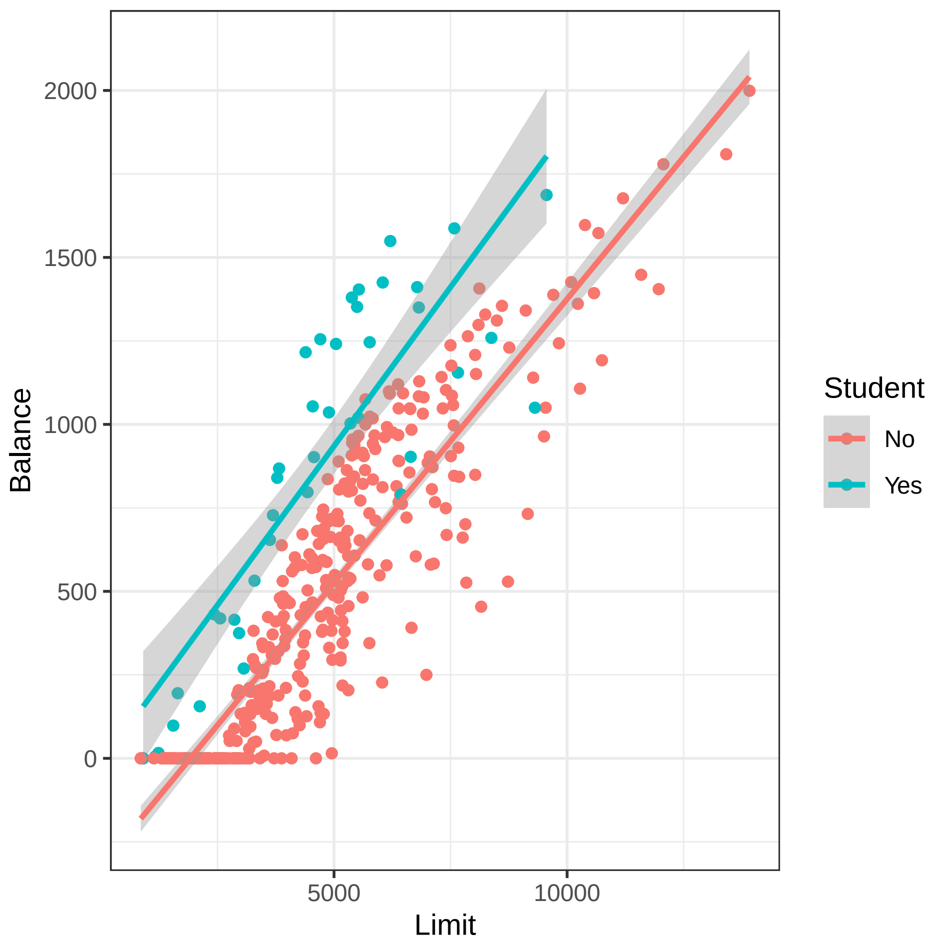


Figure 5: Balance versus Limit

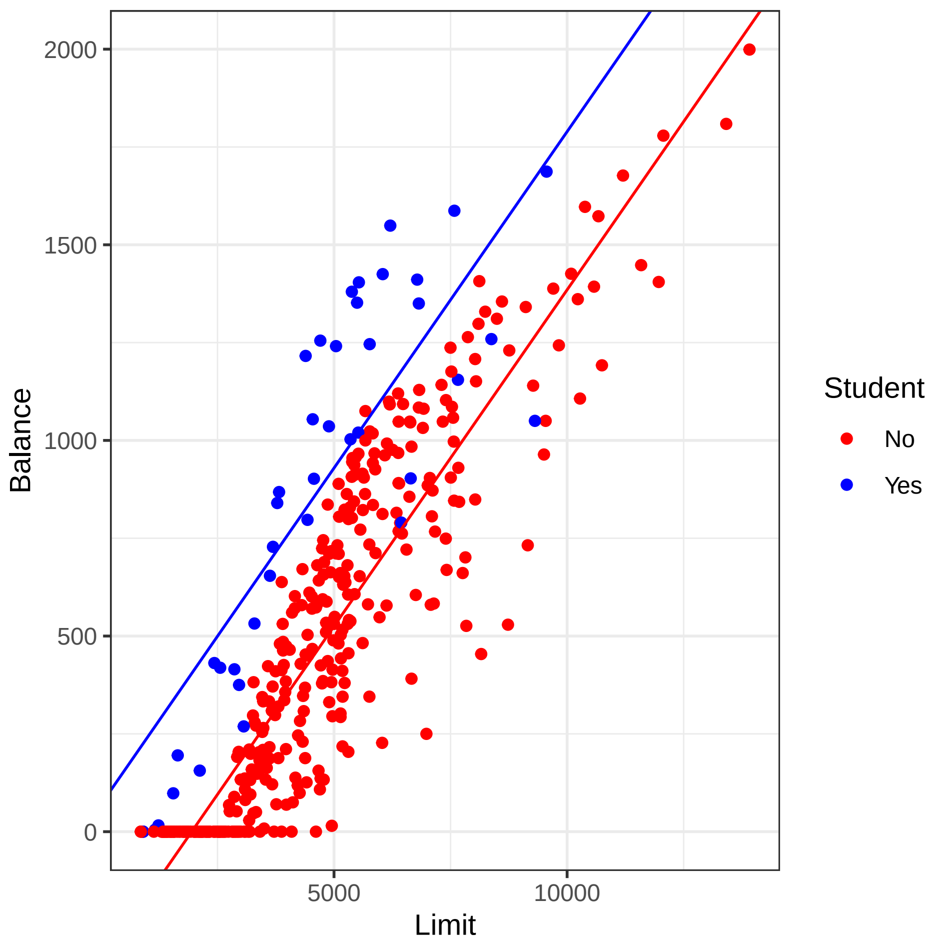


Figure 6: Figure this one out!

2.8.3 Qualitative predictors with More than Two Levels

```r
modQ3 <- lm(Balance ~ Limit + Ethnicity, data = Credit)
summary(modQ3)
```

```

Call:
lm(formula = Balance ~ Limit + Ethnicity, data = Credit)

Residuals:
    Min      1Q  Median      3Q     Max 
-677.39 -145.75   -8.75  139.56  776.46 

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)    
(Intercept)        -3.078e+02  3.417e+01  -9.007   <2e-16 ***
Limit               1.718e-01  5.079e-03  33.831   <2e-16 ***
EthnicityAsian      2.835e+01  3.304e+01   0.858    0.391    
EthnicityCaucasian  1.381e+01  2.878e+01   0.480    0.632    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 234 on 396 degrees of freedom
Multiple R-squared:  0.743,	Adjusted R-squared:  0.7411 
F-statistic: 381.6 on 3 and 396 DF,  p-value: < 2.2e-16
```

```r
coef(modQ3)
```

```
       (Intercept)              Limit     EthnicityAsian EthnicityCaucasian 
      -307.7574777          0.1718203         28.3533975         13.8089629 
```

```r
modRM <- lm(Balance ~ Limit, data = Credit)
anova(modRM, modQ3)
```

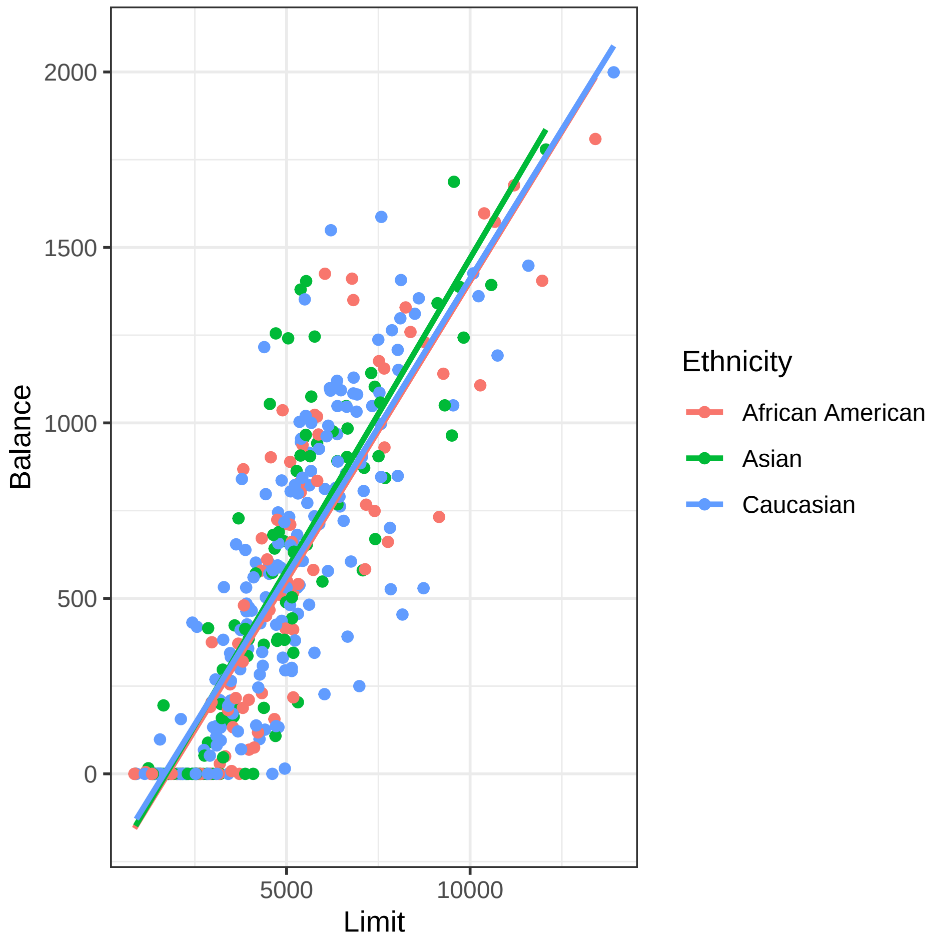
```
Analysis of Variance Table

Model 1: Balance ~ Limit
Model 2: Balance ~ Limit + Ethnicity
  Res.Df      RSS Df Sum of Sq      F Pr(>F)
1    398 21715657                           
2    396 21675307  2     40350 0.3686 0.6919
```

What follows fits three separate regression lines based on `Ethnicity`.

```r
AfAmer <- lm(Balance ~ Limit, data = subset(Credit, Ethnicity == "African American"))
AsAmer <- lm(Balance ~ Limit, data = subset(Credit, Ethnicity == "Asian"))
CaAmer <- lm(Balance ~ Limit, data = subset(Credit, Ethnicity == "Caucasian"))
rbind(coef(AfAmer), coef(AsAmer), coef(CaAmer))
```

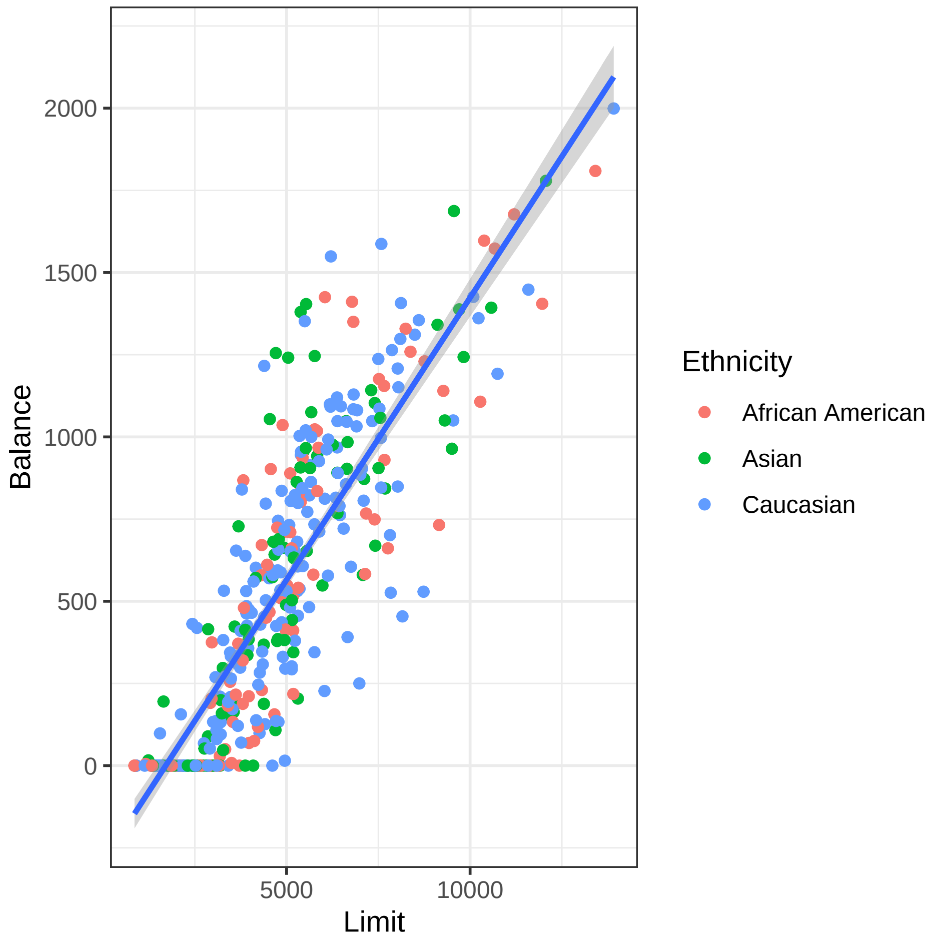
```
     (Intercept)     Limit
[1,]   -301.2245 0.1704820
[2,]   -305.4270 0.1774679
[3,]   -282.4442 0.1693873
```

```r
ggplot(data = Credit, aes(x = Limit, y = Balance, color = Ethnicity)) +
  geom_point() +
  theme_bw() +
  stat_smooth(method = "lm", se = FALSE)
```



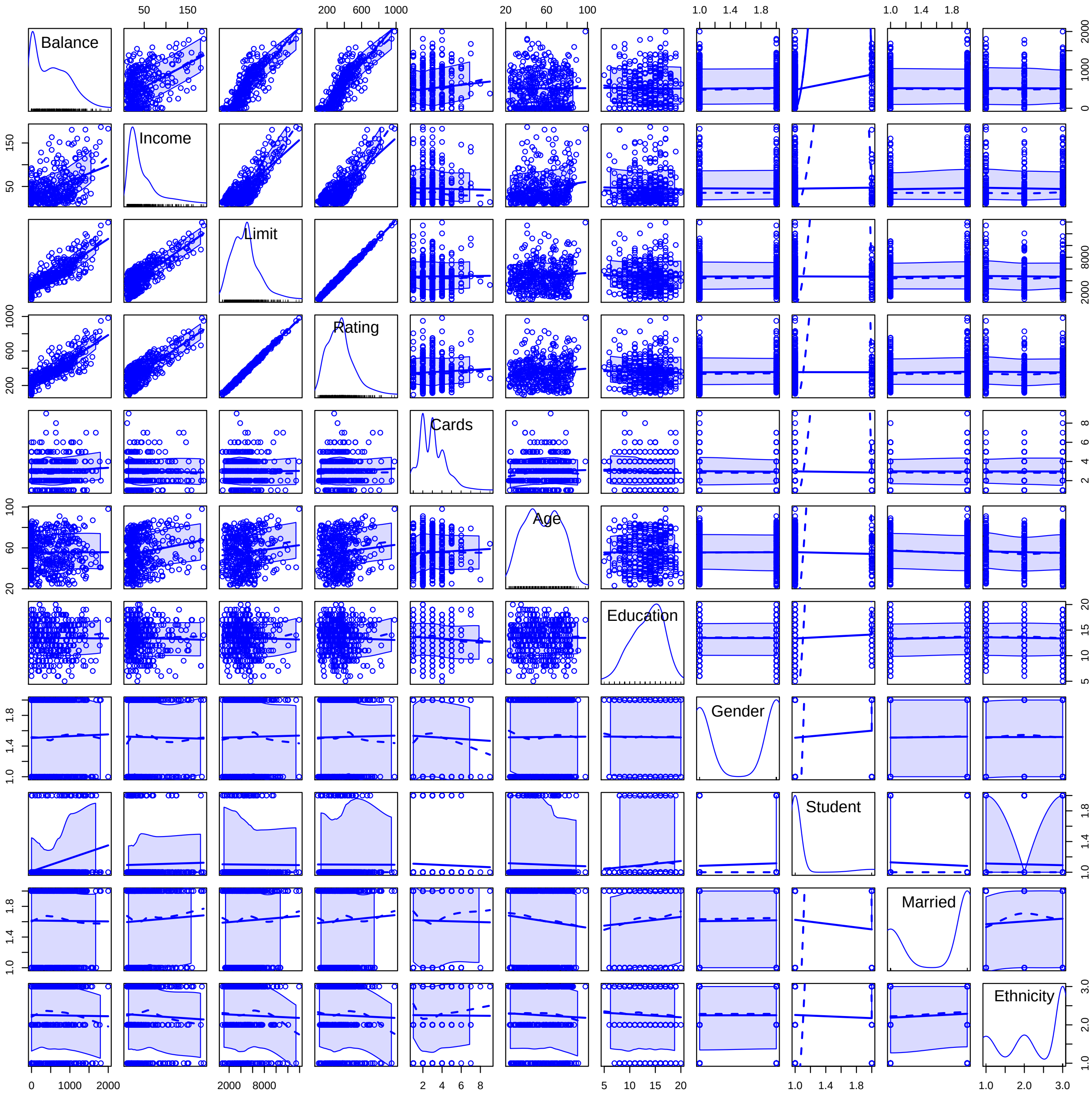
Note: `Ethnicity` is not significant, so we really should have just one line.

```
ggplot(data = Credit, aes(x = Limit, y = Balance)) +
  geom_point(aes(color = Ethnicity)) +
  theme_bw() +
  stat_smooth(method = "lm")
```



2.9 Matrix Scatterplots

```
scatterplotMatrix(~ Balance + Income + Limit + Rating + Cards + Age + Education + Gender + Student + Ma
```




```
null <- lm(Balance ~ 1, data = Credit)
full <- lm(Balance ~ ., data = Credit)
modC <- stepAIC(full, scope = list(lower = null, upper = full), direction = "backward", test = "F")
```

```
Start:  AIC=3682.12
Balance ~ ID + Income + Limit + Rating + Cards + Age + Education +
    Gender + Student + Married + Ethnicity + Utilization

              Df Sum of Sq     RSS    AIC F Value     Pr(F)
- Ethnicity    2     11141 3721961 3679.3    0.58 0.5607055
- Education    1      2980 3713800 3680.4    0.31 0.5780258
- ID           1      6003 3716824 3680.8    0.62 0.4298721
- Gender       1      7246 3718066 3680.9    0.75 0.3858391
- Married      1      8652 3719472 3681.1    0.90 0.3433763
```

```
<none>                      3710820 3682.1
- Age          1     36685 3747505 3684.1    3.82 0.0514884 .
- Rating       1     51096 3761916 3685.6    5.32 0.0216718 *
- Utilization  1     67189 3778009 3687.3    6.99 0.0085356 **
- Cards        1    130397 3841217 3693.9   13.56 0.0002635 ***
- Limit        1    286876 3997696 3709.9   29.84 8.421e-08 ***
- Income       1   3051935 6762756 3920.2  317.46 < 2.2e-16 ***
- Student      1   4655076 8365896 4005.3  484.22 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=3679.32
Balance ~ ID + Income + Limit + Rating + Cards + Age + Education +
    Gender + Student + Married + Utilization

              Df Sum of Sq     RSS    AIC F Value     Pr(F)
- Education    1      3018 3724978 3677.6    0.31 0.5752095
- ID           1      6005 3727965 3678.0    0.63 0.4293237
- Married      1      6550 3728511 3678.0    0.68 0.4091349
- Gender       1      6838 3728799 3678.1    0.71 0.3990231
<none>                      3721961 3679.3
- Age          1     39096 3761056 3681.5    4.08 0.0441954 *
- Rating       1     48423 3770384 3682.5    5.05 0.0252167 *
- Utilization  1     70020 3791981 3684.8    7.30 0.0072007 **
- Cards        1    133187 3855148 3691.4   13.88 0.0002233 ***
- Limit        1    293994 4015955 3707.7   30.65 5.709e-08 ***
- Income       1   3042199 6764160 3916.3  317.14 < 2.2e-16 ***
- Student      1   4672671 8394632 4002.7  487.11 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=3677.64
Balance ~ ID + Income + Limit + Rating + Cards + Age + Gender +
    Student + Married + Utilization

              Df Sum of Sq     RSS    AIC F Value     Pr(F)
- ID           1      6006 3730984 3676.3    0.63 0.4288762
- Gender       1      6717 3731695 3676.4    0.70 0.4028206
- Married      1      7188 3732166 3676.4    0.75 0.3868054
<none>                      3724978 3677.6
- Age          1     39443 3764421 3679.9    4.12 0.0430840 *
- Rating       1     50628 3775606 3681.0    5.29 0.0220135 *
- Utilization  1     71717 3796695 3683.3    7.49 0.0064909 **
- Cards        1    132836 3857815 3689.7   13.87 0.0002246 ***
- Limit        1    291061 4016040 3705.7   30.40 6.431e-08 ***
- Income       1   3040558 6765536 3914.4  317.53 < 2.2e-16 ***
- Student      1   4693948 8418927 4001.8  490.19 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=3676.29
Balance ~ Income + Limit + Rating + Cards + Age + Gender + Student +
    Married + Utilization
```

```
              Df Sum of Sq     RSS    AIC F Value     Pr(F)    
- Married      1      6896 3737880 3675.0    0.72 0.3963978    
- Gender       1      8373 3739357 3675.2    0.88 0.3500974    
<none>                     3730984 3676.3                      
- Age          1     37726 3768710 3678.3    3.94 0.0477529 *  
- Rating       1     50282 3781266 3679.6    5.26 0.0224028 *  
- Utilization  1     74587 3805572 3682.2    7.80 0.0054920 ** 
- Cards        1    130839 3861823 3688.1   13.68 0.0002483 ***
- Limit        1    291132 4022117 3704.3   30.43  6.31e-08 ***
- Income       1   3035245 6766229 3912.4  317.27 < 2.2e-16 ***
- Student      1   4689629 8420613 3999.9  490.21 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=3675.03
Balance ~ Income + Limit + Rating + Cards + Age + Gender + Student + 
    Utilization

              Df Sum of Sq     RSS    AIC F Value     Pr(F)    
- Gender       1      8668 3746548 3674.0    0.91 0.3415625    
<none>                     3737880 3675.0                      
- Age          1     35395 3773275 3676.8    3.70 0.0550578 .  
- Rating       1     47158 3785038 3678.0    4.93 0.0269210 *  
- Utilization  1     72879 3810759 3680.7    7.62 0.0060328 ** 
- Cards        1    135372 3873252 3687.3   14.16 0.0001936 ***
- Limit        1    303600 4041480 3704.3   31.76 3.347e-08 ***
- Income       1   3056864 6794744 3912.1  319.76 < 2.2e-16 ***
- Student      1   4770749 8508629 4002.1  499.04 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=3673.95
Balance ~ Income + Limit + Rating + Cards + Age + Student + Utilization

              Df Sum of Sq     RSS    AIC F Value     Pr(F)    
<none>                     3746548 3674.0                      
- Age          1     35671 3782220 3675.7    3.73 0.0540906 .  
- Rating       1     46902 3793451 3676.9    4.91 0.0273163 *  
- Utilization  1     75071 3821620 3679.9    7.85 0.0053206 ** 
- Cards        1    136510 3883058 3686.3   14.28 0.0001817 ***
- Limit        1    303278 4049826 3703.1   31.73 3.383e-08 ***
- Income       1   3048196 6794744 3910.1  318.93 < 2.2e-16 ***
- Student      1   4765085 8511634 4000.2  498.57 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
modC
```

```
Call:
lm(formula = Balance ~ Income + Limit + Rating + Cards + Age + 
    Student + Utilization, data = Credit)

Coefficients:
(Intercept)       Income        Limit       Rating        Cards          Age  
  -487.7563      -6.9234       0.1823       1.0649      16.3703      -0.5606  
```

```
 StudentYes  Utilization
   403.9969     145.4632
```

```r
modD <- stepAIC(null, scope = list(lower = null, upper = full), direction = "forward", test = "F")
```

```
Start:  AIC=4905.56
Balance ~ 1

              Df Sum of Sq      RSS    AIC F Value     Pr(F)
+ Rating       1  62904790 21435122 4359.6 1167.99 < 2.2e-16 ***
+ Limit        1  62624255 21715657 4364.8 1147.76 < 2.2e-16 ***
+ Utilization  1  27382381 56957530 4750.5  191.34 < 2.2e-16 ***
+ Income       1  18131167 66208745 4810.7  108.99 < 2.2e-16 ***
+ Student      1   5658372 78681540 4879.8   28.62 1.488e-07 ***
+ Cards        1    630416 83709496 4904.6    3.00   0.08418 .
<none>                     84339912 4905.6
+ Gender       1     38892 84301020 4907.4    0.18   0.66852
+ Education    1      5481 84334431 4907.5    0.03   0.87231
+ ID           1      3101 84336810 4907.5    0.01   0.90377
+ Married      1      2715 84337197 4907.5    0.01   0.90994
+ Age          1       284 84339628 4907.6    0.00   0.97081
+ Ethnicity    2     18454 84321458 4909.5    0.04   0.95749
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=4359.63
Balance ~ Rating

              Df Sum of Sq      RSS    AIC F Value     Pr(F)
+ Utilization  1  11779424  9655698 4042.6  484.32 < 2.2e-16 ***
+ Income       1  10902581 10532541 4077.4  410.95 < 2.2e-16 ***
+ Student      1   5735163 15699959 4237.1  145.02 < 2.2e-16 ***
+ Age          1    649110 20786012 4349.3   12.40 0.0004798 ***
+ Cards        1    138580 21296542 4359.0    2.58 0.1087889
+ Married      1    118209 21316913 4359.4    2.20 0.1386707
<none>                     21435122 4359.6
+ Education    1     27243 21407879 4361.1    0.51 0.4776403
+ Gender       1     16065 21419057 4361.3    0.30 0.5855899
+ ID           1     14092 21421030 4361.4    0.26 0.6096002
+ Limit        1      7960 21427162 4361.5    0.15 0.7011619
+ Ethnicity    2     51100 21384022 4362.7    0.47 0.6233922
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=4042.64
Balance ~ Rating + Utilization

            Df Sum of Sq     RSS    AIC F Value     Pr(F)
+ Student    1   2671767 6983931 3915.1 151.493 < 2.2e-16 ***
+ Income     1   1025771 8629927 3999.7  47.069  2.65e-11 ***
+ Married    1     95060 9560638 4040.7   3.937   0.04791 *
+ Age        1     50502 9605197 4042.5   2.082   0.14983
<none>                   9655698 4042.6
+ Limit      1     42855 9612843 4042.9   1.765   0.18472
+ Education  1     28909 9626789 4043.4   1.189   0.27616
```

```
+ ID         1     12426 9643273 4044.1   0.510   0.47545
+ Gender     1      7187 9648511 4044.3   0.295   0.58735
+ Cards      1      3371 9652327 4044.5   0.138   0.71017
+ Ethnicity  2     13259 9642439 4046.1   0.272   0.76231
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=3915.06
Balance ~ Rating + Utilization + Student

            Df Sum of Sq     RSS    AIC F Value   Pr(F)
+ Income     1   2893712 4090219 3703.1 279.451 < 2e-16 ***
+ Limit      1     77766 6906165 3912.6   4.448 0.03557 *
+ Age        1     58618 6925313 3913.7   3.343 0.06823 .
<none>                   6983931 3915.1
+ Married    1     33686 6950245 3915.1   1.914 0.16725
+ Education  1      2344 6981587 3916.9   0.133 0.71591
+ ID         1      1986 6981946 3916.9   0.112 0.73768
+ Cards      1      1302 6982630 3917.0   0.074 0.78627
+ Gender     1         9 6983922 3917.1   0.001 0.98212
+ Ethnicity  2      1715 6982217 3919.0   0.048 0.95278
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=3703.06
Balance ~ Rating + Utilization + Student + Income

            Df Sum of Sq     RSS    AIC F Value     Pr(F)
+ Limit      1    178086 3912133 3687.3 17.9354 2.847e-05 ***
+ Age        1     34096 4056122 3701.7  3.3120   0.06953 .
<none>                   4090219 3703.1
+ Married    1     15941 4074278 3703.5  1.5416   0.21512
+ Gender     1      8880 4081339 3704.2  0.8572   0.35508
+ ID         1      5005 4085214 3704.6  0.4827   0.48760
+ Cards      1      4628 4085591 3704.6  0.4463   0.50447
+ Education  1       445 4089774 3705.0  0.0428   0.83613
+ Ethnicity  2     16108 4074111 3705.5  0.7769   0.46054
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=3687.25
Balance ~ Rating + Utilization + Student + Income + Limit

            Df Sum of Sq     RSS    AIC F Value     Pr(F)
+ Cards      1    129913 3782220 3675.7 13.4989 0.0002718 ***
+ Age        1     29075 3883058 3686.3  2.9427 0.0870572 .
<none>                   3912133 3687.3
+ Gender     1     10045 3902089 3688.2  1.0116 0.3151296
+ Married    1      8872 3903262 3688.3  0.8932 0.3451820
+ ID         1      3818 3908316 3688.9  0.3839 0.5358946
+ Education  1      3501 3908633 3688.9  0.3520 0.5533444
+ Ethnicity  2     12590 3899543 3690.0  0.6328 0.5316436
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Step:  AIC=3675.74
Balance ~ Rating + Utilization + Student + Income + Limit + Cards

            Df Sum of Sq     RSS    AIC F Value   Pr(F)
+ Age        1     35671 3746548 3674.0  3.7323 0.05409 .
<none>                   3782220 3675.7
+ Gender     1      8945 3773275 3676.8  0.9293 0.33564
+ ID         1      5574 3776646 3677.2  0.5785 0.44735
+ Married    1      4801 3777419 3677.2  0.4982 0.48069
+ Education  1      3733 3778487 3677.3  0.3873 0.53408
+ Ethnicity  2     10981 3771239 3678.6  0.5693 0.56641
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Step:  AIC=3673.95
Balance ~ Rating + Utilization + Student + Income + Limit + Cards +
    Age

            Df Sum of Sq     RSS    AIC F Value  Pr(F)
<none>                   3746548 3674.0
+ Gender     1    8668.5 3737880 3675.0 0.90676 0.3416
+ ID         1    7358.8 3739190 3675.2 0.76949 0.3809
+ Married    1    7191.5 3739357 3675.2 0.75197 0.3864
+ Education  1    3505.2 3743043 3675.6 0.36616 0.5455
+ Ethnicity  2    8615.0 3737933 3677.0 0.44943 0.6383
```

```r
modD
```

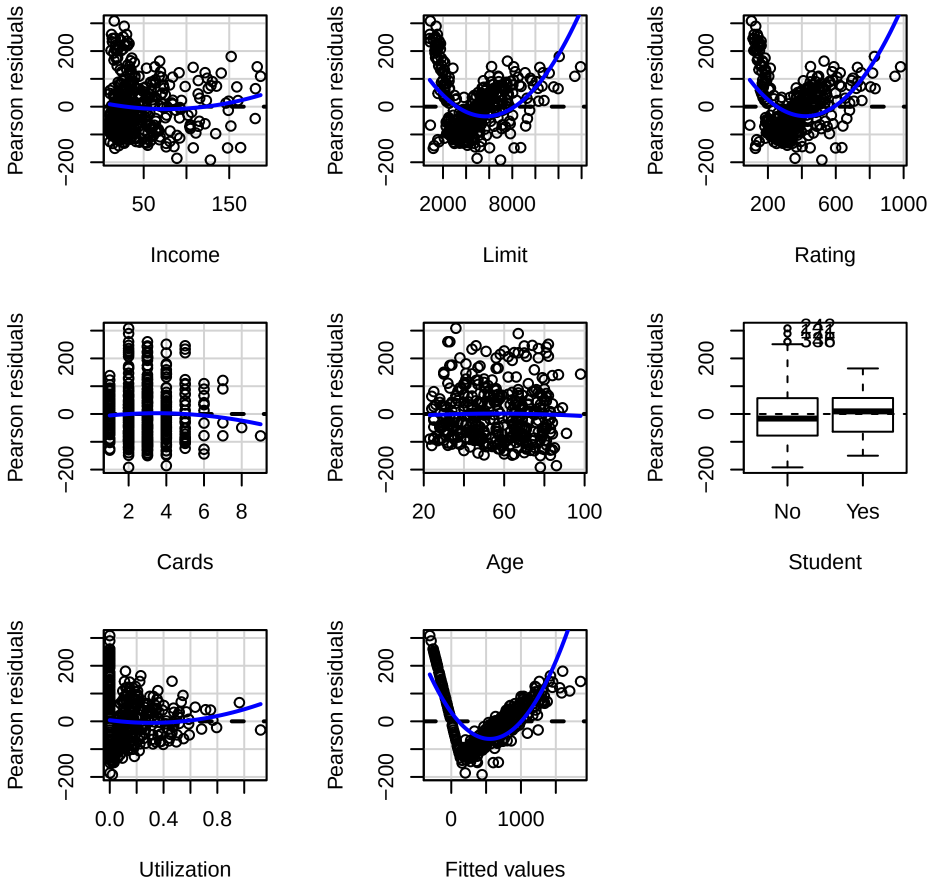
```
Call:
lm(formula = Balance ~ Rating + Utilization + Student + Income +
    Limit + Cards + Age, data = Credit)

Coefficients:
(Intercept)       Rating  Utilization   StudentYes       Income        Limit
  -487.7563       1.0649     145.4632     403.9969      -6.9234       0.1823
      Cards          Age
    16.3703      -0.5606
```

```r
# Predict
predict(modC, newdata = data.frame(Income = 80, Limit = 5000, Cards = 3, Age = 52, Student = "No", Rati
```

```
       fit      lwr      upr
1 756.2699 309.4089 1203.131
```

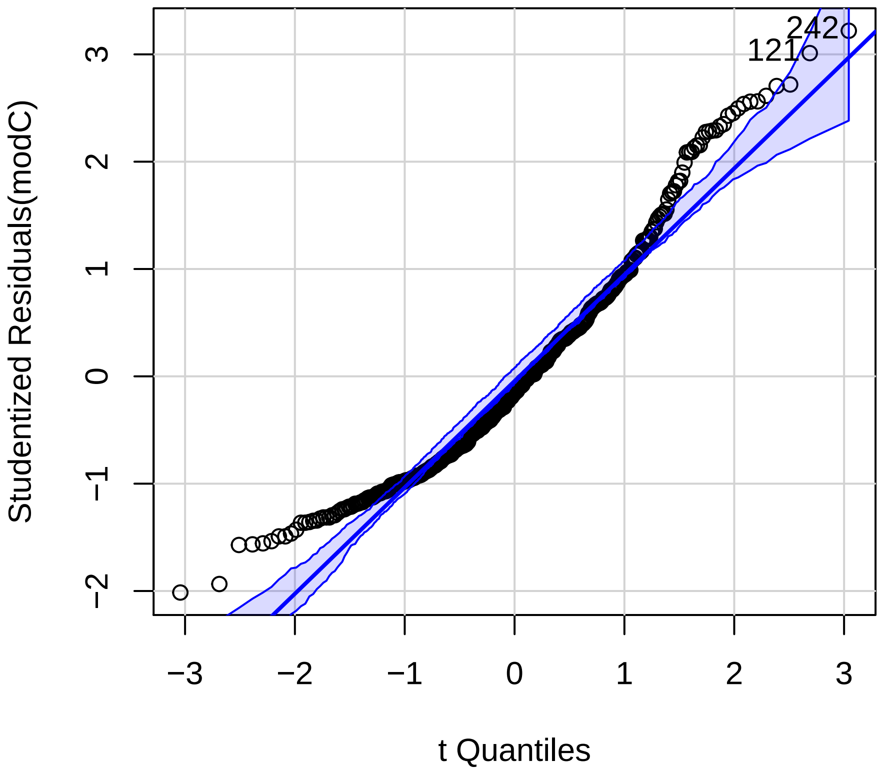
2.10 More Diagnostic Plots

```r
residualPlots(modC)
```



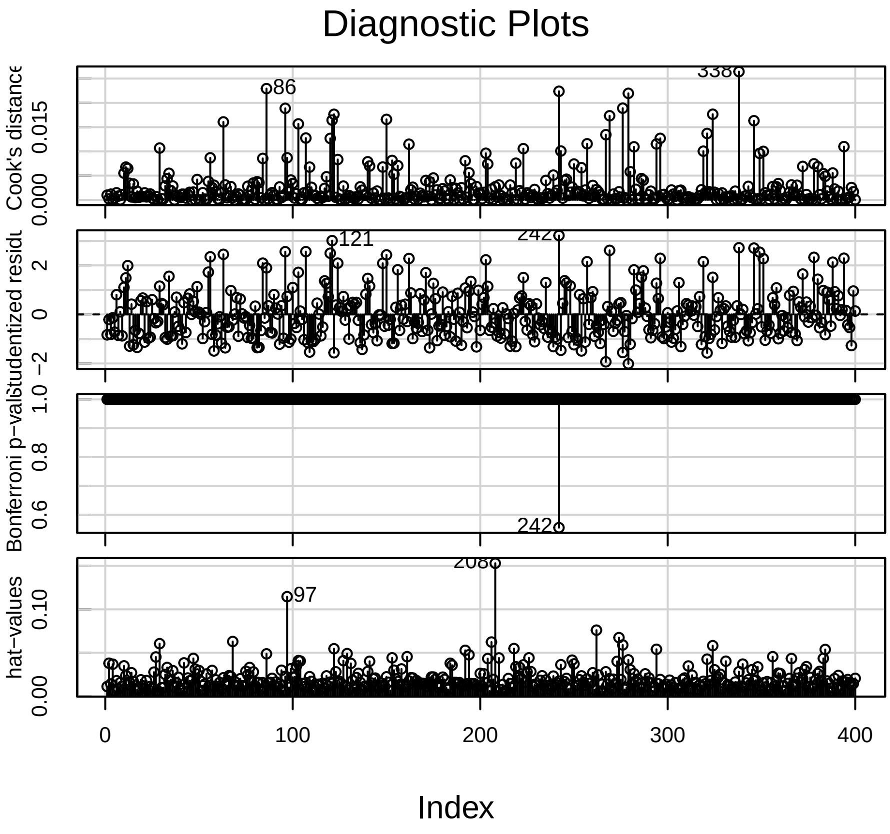

```
             Test stat Pr(>|Test stat|)
Income          1.5431           0.1236
Limit          12.2615           <2e-16 ***
Rating         11.7689           <2e-16 ***
Cards          -0.7709           0.4412
Age            -0.2795           0.7800
Student
Utilization     1.4250           0.1550
Tukey test     25.9407           <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
qqPlot(modC)
```




```
[1] 121 242
```

```
influenceIndexPlot(modC)
```



2.11 Non-linear Relationships

```
library(ISLR)
car1 <- lm(mpg ~ horsepower, data = Auto)
car2 <- lm(mpg ~ poly(horsepower, 2), data = Auto)
car5 <- lm(mpg ~ poly(horsepower, 5), data = Auto)
xs <- seq(min(Auto$horsepower), max(Auto$horsepower), length = 500)
y1 <- predict(car1, newdata = data.frame(horsepower = xs))
y2 <- predict(car2, newdata = data.frame(horsepower = xs))
y5 <- predict(car5, newdata = data.frame(horsepower = xs))
DF <- data.frame(x = xs, y1 = y1, y2 = y2, y5 = y5)
ggplot(data = Auto, aes(x = horsepower, y = mpg)) +
  geom_point() +
  theme_bw() +
  geom_line(data = DF, aes(x = x, y = y1), color = "red") +
  geom_line(data = DF, aes(x = x, y = y2), color = "blue") +
  geom_line(data = DF, aes(x = x, y = y5), color = "green")
```

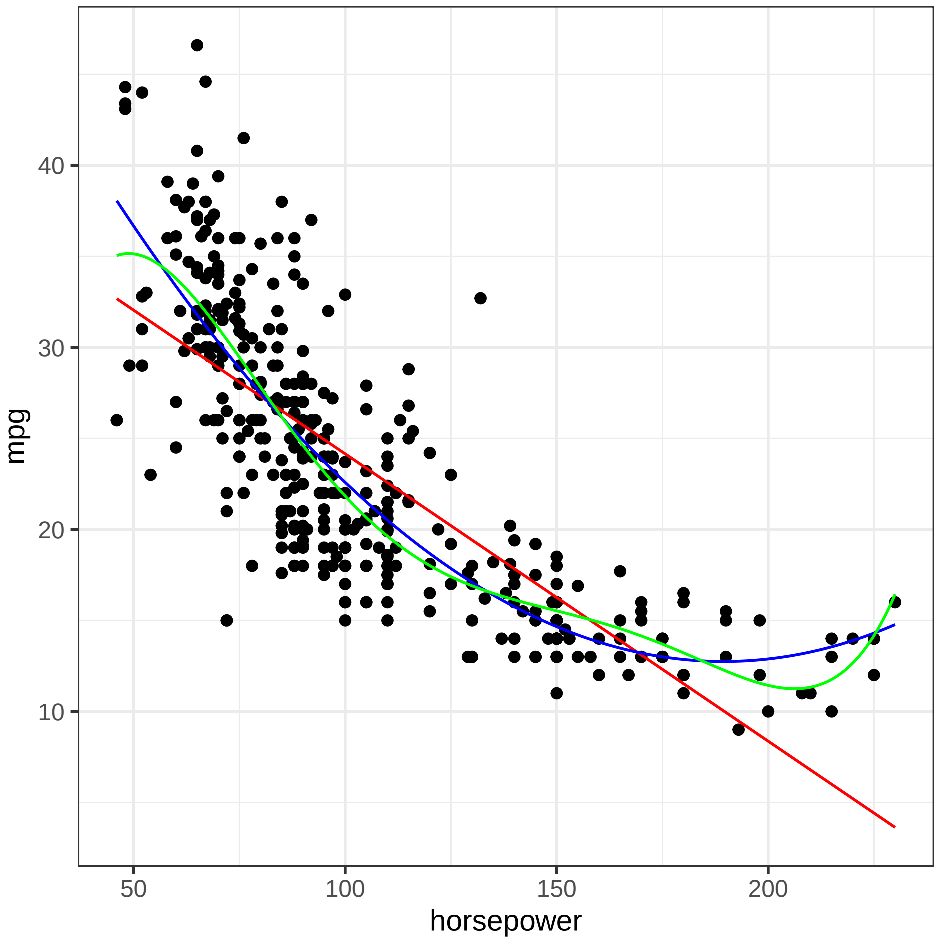
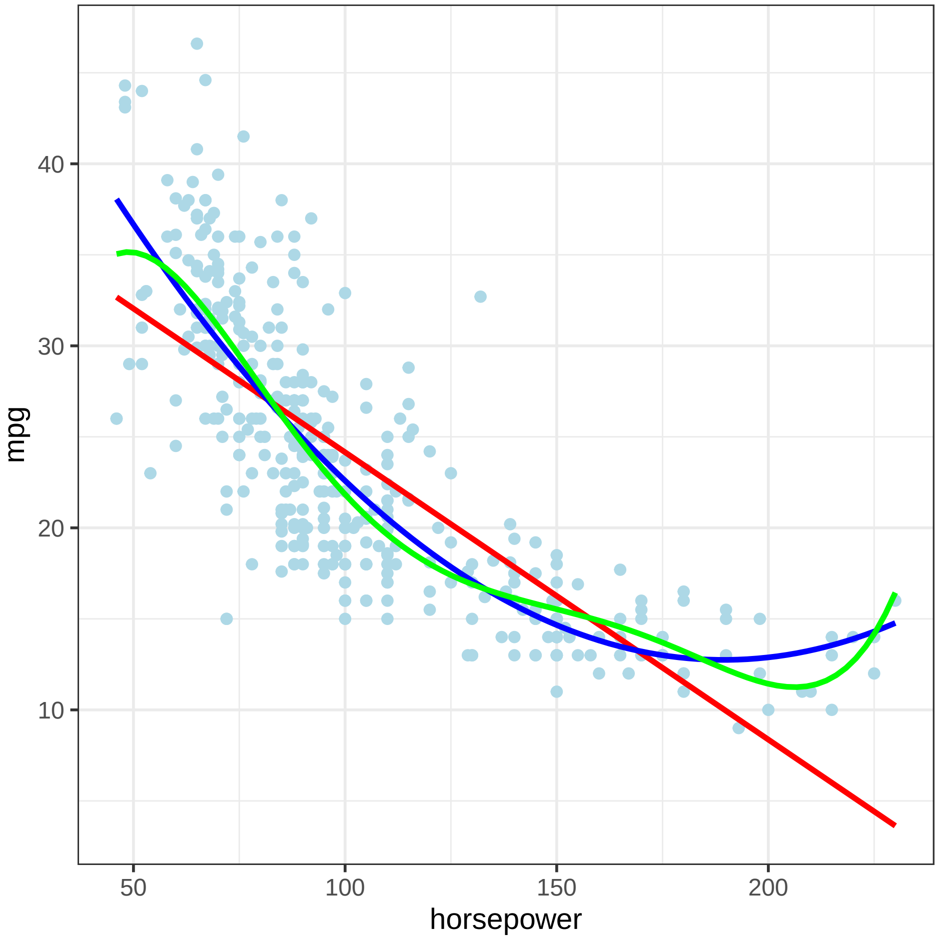


Figure 7: Showing non-linear relationships

```
ggplot(data = Auto, aes(x = horsepower, y = mpg)) +
  geom_point(color = "lightblue") +
  theme_bw() +
```

```
  stat_smooth(method = "lm", data = Auto, color = "red", se = FALSE) +
  stat_smooth(method = "lm", formula = y ~ poly(x, 2), data = Auto, color = "blue", se = FALSE) +
  stat_smooth(method = "lm", formula = y ~ poly(x, 5), data = Auto, color = "green", se = FALSE)
```

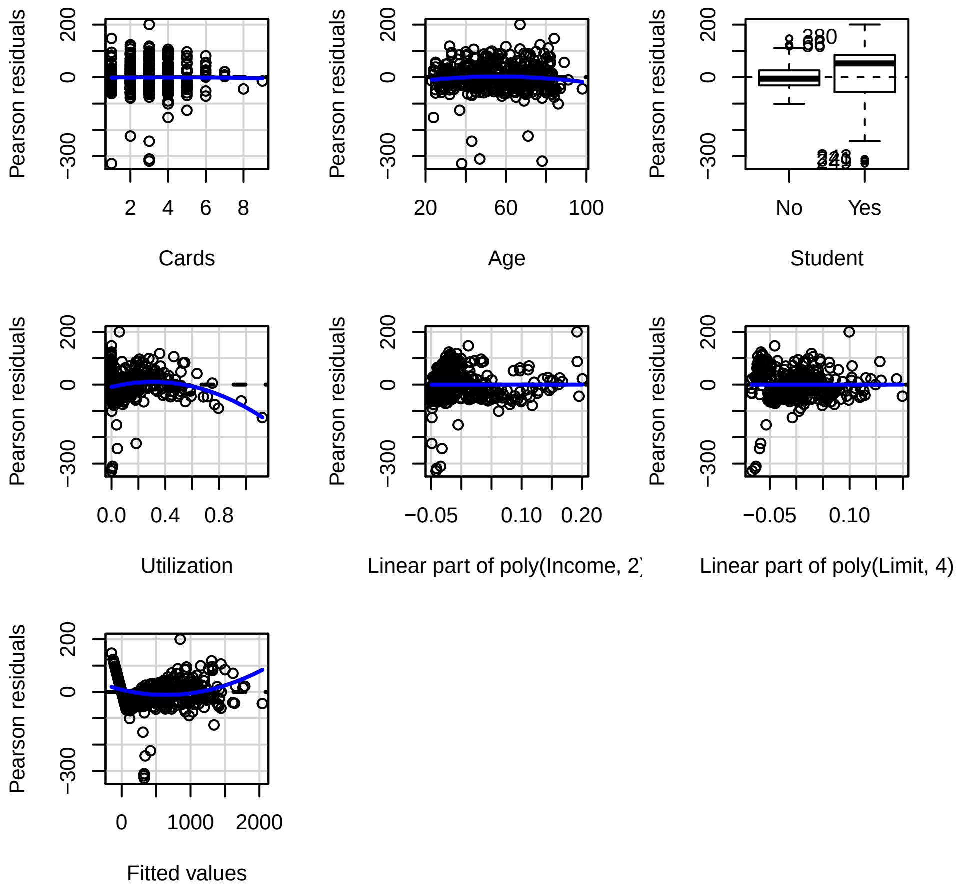



```
newC <- update(modC, .~. - Limit - Income - Rating + poly(Income, 2) + poly(Limit, 4))
summary(newC)
```

```

Call:
lm(formula = Balance ~ Cards + Age + Student + Utilization +
    poly(Income, 2) + poly(Limit, 4), data = Credit)

Residuals:
    Min      1Q  Median      3Q     Max
-327.92  -31.35   -3.54   30.91  200.35

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)        408.5875    12.2594  33.328  < 2e-16 ***
Cards               17.2245     2.1497   8.013 1.32e-14 ***
Age                 -0.7256     0.1699  -4.270 2.46e-05 ***
StudentYes         369.4275    10.7980  34.213  < 2e-16 ***
Utilization        422.8552    36.6956  11.523  < 2e-16 ***
poly(Income, 2)1 -5263.1585   167.9209 -31.343  < 2e-16 ***
poly(Income, 2)2  -896.3437    95.2993  -9.406  < 2e-16 ***
```

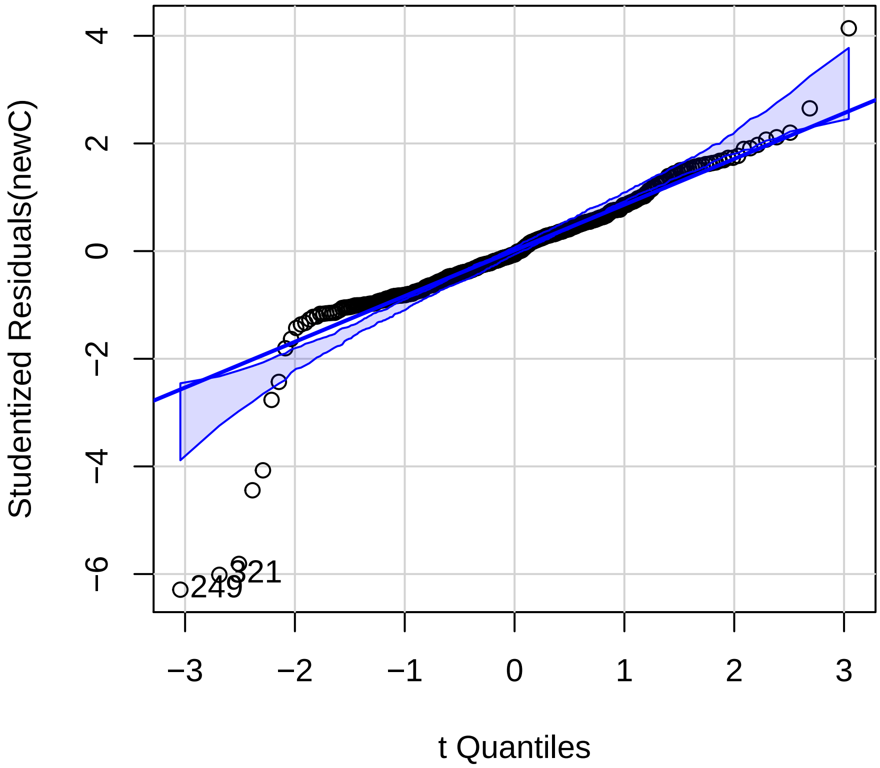
```
poly(Limit, 4)1  11775.8484   165.8389  71.008  < 2e-16 ***
poly(Limit, 4)2   1920.4673    97.7898  19.639  < 2e-16 ***
poly(Limit, 4)3   -814.2430    61.0972 -13.327  < 2e-16 ***
poly(Limit, 4)4    393.7068    59.2827   6.641 1.05e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 57.15 on 389 degrees of freedom
Multiple R-squared:  0.9849,    Adjusted R-squared:  0.9845
F-statistic:  2544 on 10 and 389 DF,  p-value: < 2.2e-16
```

```
residualPlots(newC)
```



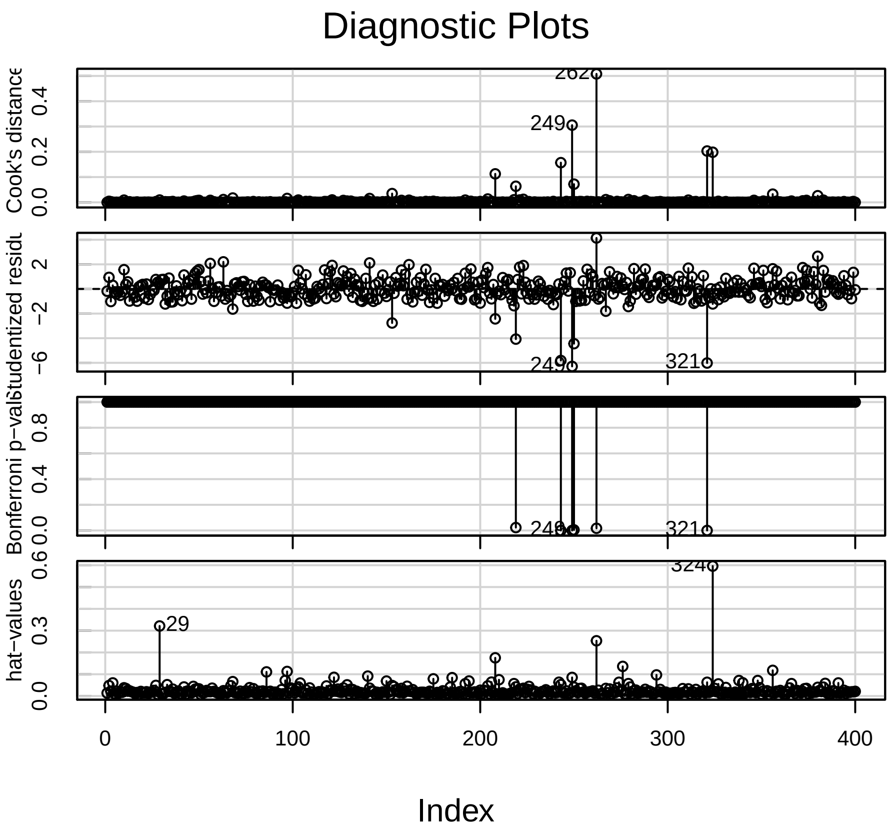

```
                Test stat Pr(>|Test stat|)
Cards             -0.1284           0.8979
Age               -1.2123           0.2261
Student
Utilization       -5.4273        1.010e-07 ***
poly(Income, 2)
poly(Limit, 4)
Tukey test         8.1970        2.465e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
qqPlot(newC)
```




```
[1] 249 321
```

```
influenceIndexPlot(newC)
```



2.12 Variance Inflation Factor (VIF)

The VIF is the ratio of the variance of $\hat{\beta}_j$ when fitting the full model divided by the variance of $\hat{\beta}_j$ if it is fit on its own. The smallest possible value for VIF is 1, which indicates the complete absence of collinearity. The VIF for each variable can be computed using the formula:

$$VIF(\hat{\beta}_j) = \frac{1}{1 - R^2_{X_j|X_{-j}}}$$

where $R^2_{X_j|X_{-j}}$ is the R^2 from a regression of X_j onto all of the other predictors. If $R^2_{X_j|X_{-j}}$ is close to one, then collinearity is present, and so the VIF will be large.

Compute the VIF for each $\hat{\beta}_j$ of `modC`

```
modC
```

```

Call:
lm(formula = Balance ~ Income + Limit + Rating + Cards + Age +
    Student + Utilization, data = Credit)

Coefficients:
(Intercept)       Income        Limit       Rating        Cards          Age
  -487.7563      -6.9234       0.1823       1.0649      16.3703      -0.5606
 StudentYes  Utilization
   403.9969     145.4632
```

```
R2inc <- summary(lm(Income ~ Limit + Rating + Cards + Age + Student + Utilization, data = Credit))$r.squ
R2inc
```

```
[1] 0.8716908
```

```
VIFinc <- 1/(1 - R2inc)
VIFinc
```

```
[1] 7.793671
```

```
R2lim <- summary(lm(Limit ~ Income + Rating + Cards + Age + Student + Utilization, data = Credit))$r.squ
R2lim
```

```
[1] 0.9957067
```

```
VIFlim <- 1/(1 - R2lim)
VIFlim
```

```
[1] 232.9193
```

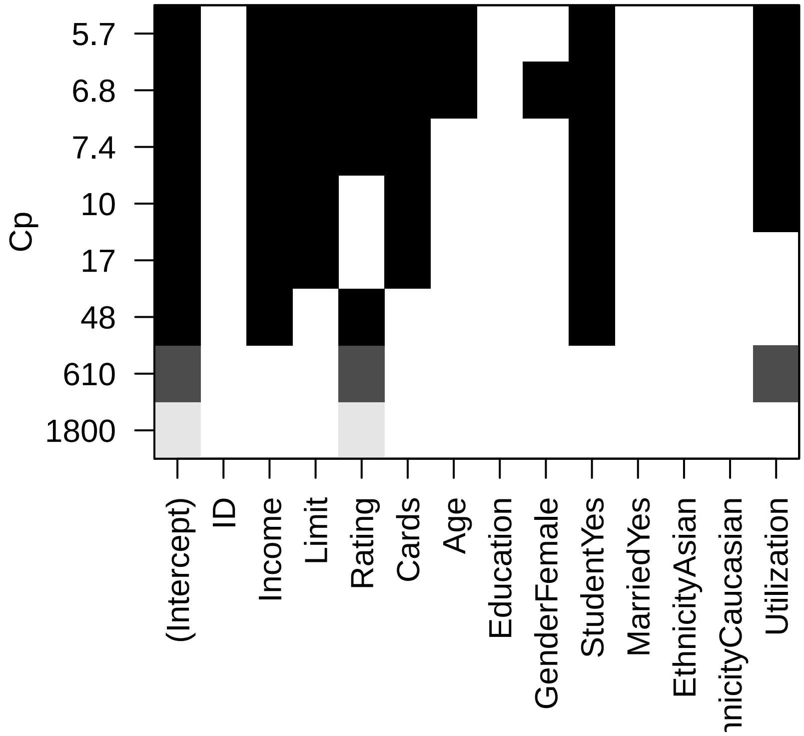
This is tedious is there a function to do this? Yes!

```
car::vif(modC)
```

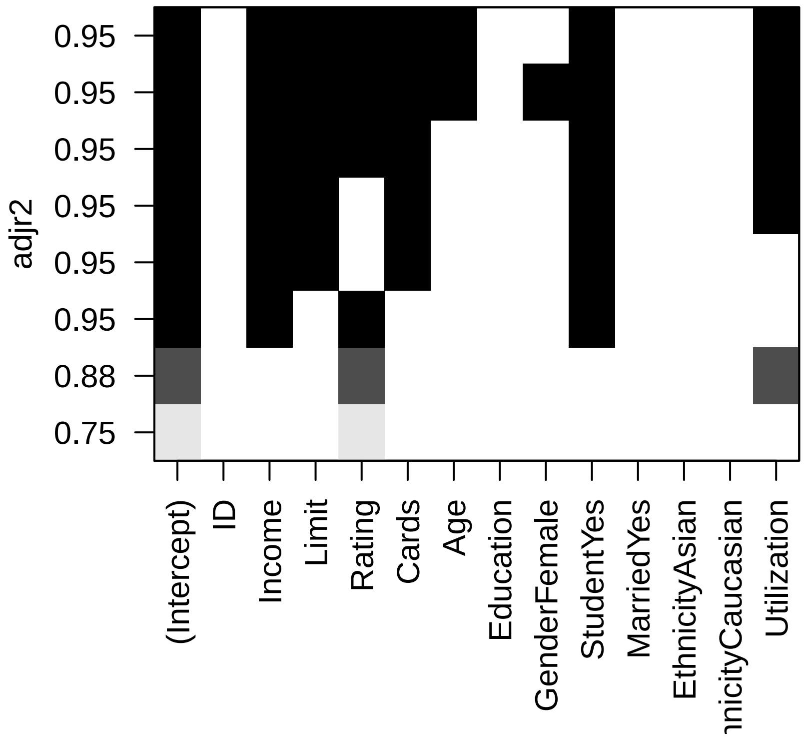
```
     Income       Limit      Rating       Cards         Age     Student
   7.793671  232.919318  230.957276    1.472901    1.046060    1.233070
Utilization
   3.323397
```

2.12.1 Building Models with `leaps`

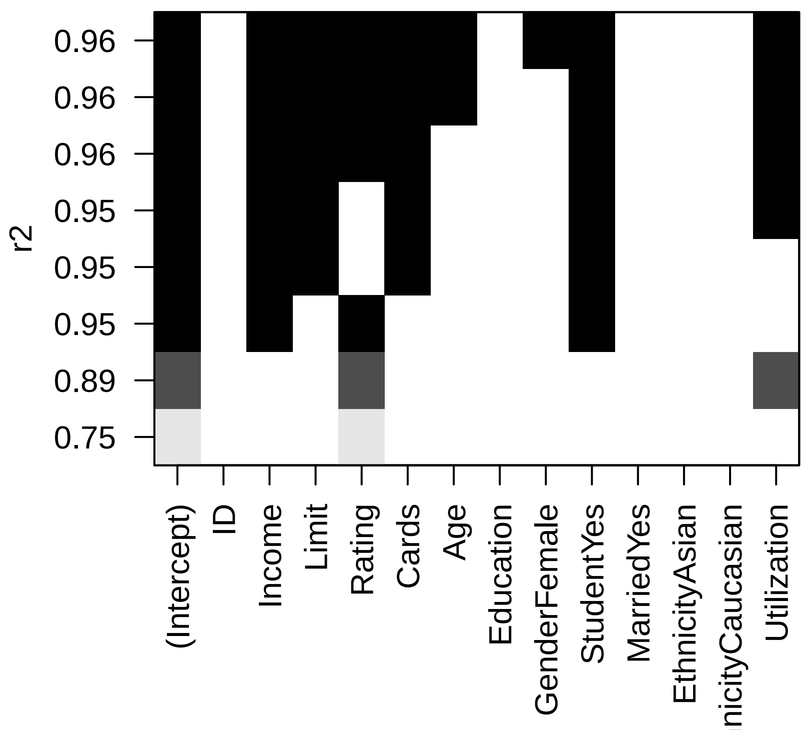
```
library(leaps)
bestsubsetmodel <- regsubsets(Balance ~ ., data = Credit)
plot(bestsubsetmodel, scale = "Cp")
```



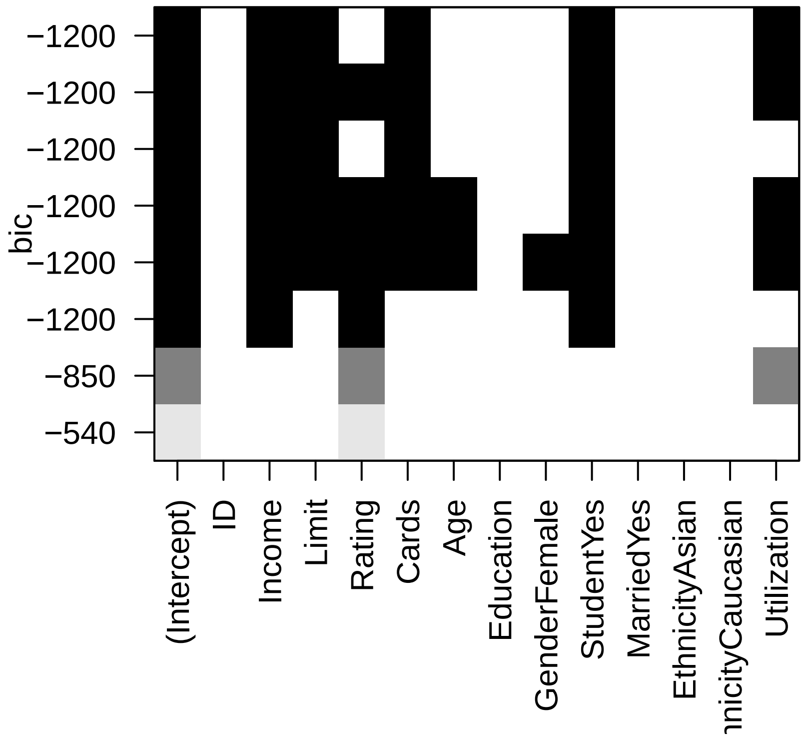
```r
plot(bestsubsetmodel, scale = "adjr2")
```




```r
plot(bestsubsetmodel, scale = "r2")
```



```r
plot(bestsubsetmodel, scale = "bic")
```




```r
coef(bestsubsetmodel, 7)
```

```
 (Intercept)       Income        Limit       Rating        Cards          Age
-487.7563318   -6.9233687    0.1822902    1.0649244   16.3703375   -0.5606190
  StudentYes  Utilization
 403.9969037  145.4632091
```

2.13 Exercise

- Create a model that predicts an individuals credit rating (`Rating`).
- Create another model that predicts rating with `Limit`, `Cards`, `Married`, `Student`, and `Education` as features.
- Use your model to predict the `Rating` for an individual that has a credit card limit of $6,000, has 4 credit cards, is married, is not a student, and has an undergraduate degree (`Education` = 16).
- Use your model to predict the `Rating` for an individual that has a credit card limit of $12,000, has 2 credit cards, is married, is not a student, and has an eighth grade education (`Education` = 8).