Rice Example: Multiple Regression and the car() package

Multiple regression extends the simple linear regression model to include multiple predictor variables. In this example, we consider the yield (response) versus height and tillers (predictors) for n = 8 varieties of rice.

We go beyond the basic model fitting using lm() to illustrate several topics:

1. Predicted values, confidence intervals and prediction intervals using the predict() function.
2. Additional hypothesis tests using lht() from the car package. Comparing a reduced versus full model using anova().
3. When there are two or more predictors, anova() is different from Anova() from the car package. We are generally interested in the Anova results! Anova() gives the unique (or marginal) ANOVA table (which does not depend on the order the predictors are listed). anova() gives the sequential ANOVA table (which depends on the order the predictors are listed).

```
library(scatterplot3d)
library(car)
Rice <- read.csv("~/Dropbox/STAT512/Lectures/MultReg1/MR1_Rice.csv")
Rice
```

```
##   yield    ht tillers
## 1 5.755 110.5    14.5
## 2 5.939 105.4    16.0
## 3 6.010 118.1    14.6
## 4 6.545 104.5    18.2
## 5 6.730  93.6    15.4
## 6 6.750  84.1    17.6
## 7 6.899  77.8    17.9
## 8 7.862  75.6    19.4
```

Pairwise correlations and plots

The cor function is handy for computing pairwise correlations. But in order to get a formal test of correlation, we need to use cor.test().

```
cor(Rice)
```

data.frame

```
##              yield         ht    tillers
## yield    1.0000000 -0.8687070  0.8349761
## ht      -0.8687070  1.0000000 -0.7762814
## tillers  0.8349761 -0.7762814  1.0000000
```

```
with(cor.test(yield, ht), data=Rice)
```

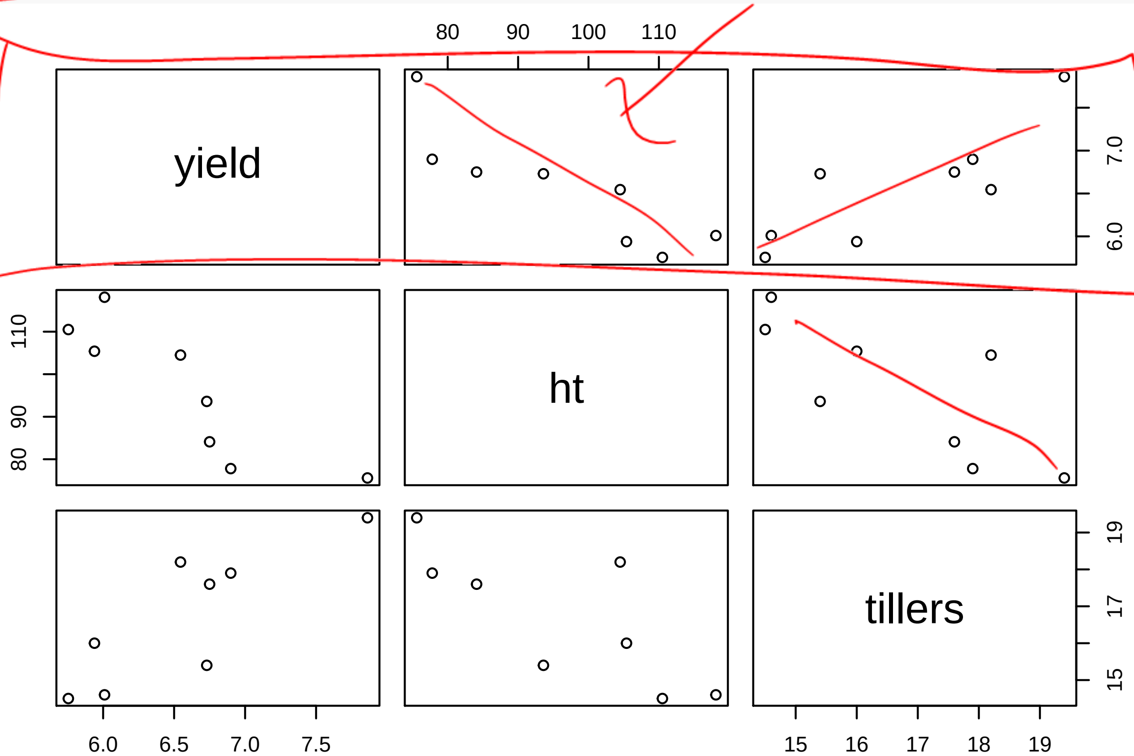
```
##
##  Pearson's product-moment correlation
##
## data:  yield and ht
## t = -4.2959, df = 6, p-value = 0.005116
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.9759487 -0.4229363
```

$H_0: \rho = 0$

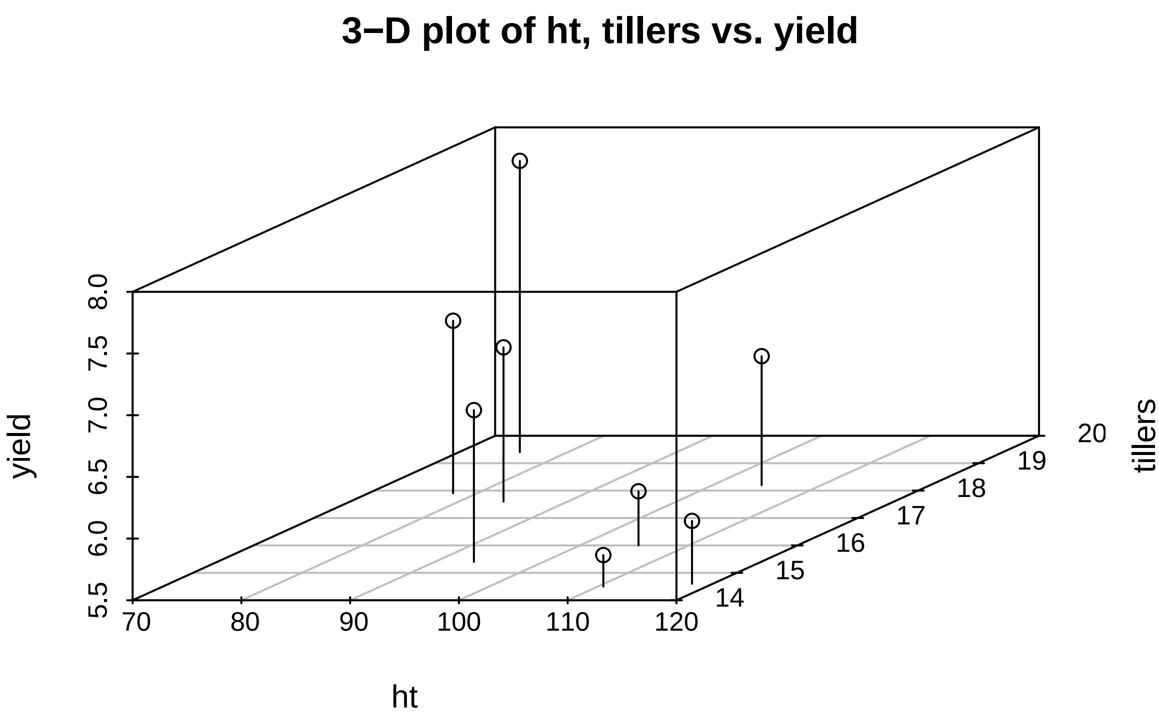
```
## sample estimates:
##       cor
## -0.868707
```

data.frame

```r
pairs(Rice)
```




```r
#3-D Graph
#The "h" option means: vertical lines to the horizontal plane.
with(scatterplot3d(ht, tillers, yield, type = "h",
                   main = "3-D plot of ht, tillers vs. yield"), data = Rice)
```



Simple Linear Regression

Models 1 and 2 are the simple linear regressions (including just one predictor each).

```
Model1 <- lm(yield ~ ht, data = Rice)
summary(Model1)
```

```
##
## Call:
## lm(formula = yield ~ ht, data = Rice)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.34626 -0.27605 -0.09448  0.27023  0.53495
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 10.137455   0.842265  12.036    2e-05 ***
## ht          -0.037175   0.008653  -4.296  0.00512 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3624 on 6 degrees of freedom
## Multiple R-squared:  0.7547, Adjusted R-squared:  0.7138
## F-statistic: 18.46 on 1 and 6 DF,  p-value: 0.005116
```

β_1

$H_0: \beta_1 = 0$

75% of yield variability explained by ht

```
Model2 <- lm(yield ~ tillers, data = Rice)
summary(Model2)
```

```
##
## Call:
## lm(formula = yield ~ tillers, data = Rice)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.4820 -0.1935 -0.0628  0.1912  0.5724
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.37548    1.40249   0.981  0.36460
## tillers      0.31053    0.08355   3.717  0.00989 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4026 on 6 degrees of freedom
## Multiple R-squared:  0.6972, Adjusted R-squared:  0.6467
## F-statistic: 13.81 on 1 and 6 DF,  p-value: 0.009891
```

$H_0: \beta_{Tillers} = 0$

70% var of yield explained by tiller

Multiple Regression

or

Model3 is the multiple regression model, including both ht and tillers.

```
Model3 <- lm(yield ~ ht + tillers, data = Rice)
summary(Model3)
```

```
##
## Call:
## lm(formula = yield ~ ht + tillers, data = Rice)
##
## Residuals:
##        1        2        3        4        5        6        7        8
## -0.13596 -0.29855  0.28449 -0.04461  0.30241 -0.23388 -0.27959  0.40569
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.33560    2.94293   2.153   0.0839 .
## ht          -0.02375    0.01290  -1.842   0.1249
## tillers      0.15031    0.11207   1.341   0.2375
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3404 on 5 degrees of freedom
## Multiple R-squared:  0.8196, Adjusted R-squared:  0.7474
## F-statistic: 11.36 on 2 and 5 DF,  p-value: 0.01383
```

$\hat{\beta}$ vector

$H_0: \beta_1 = 0$

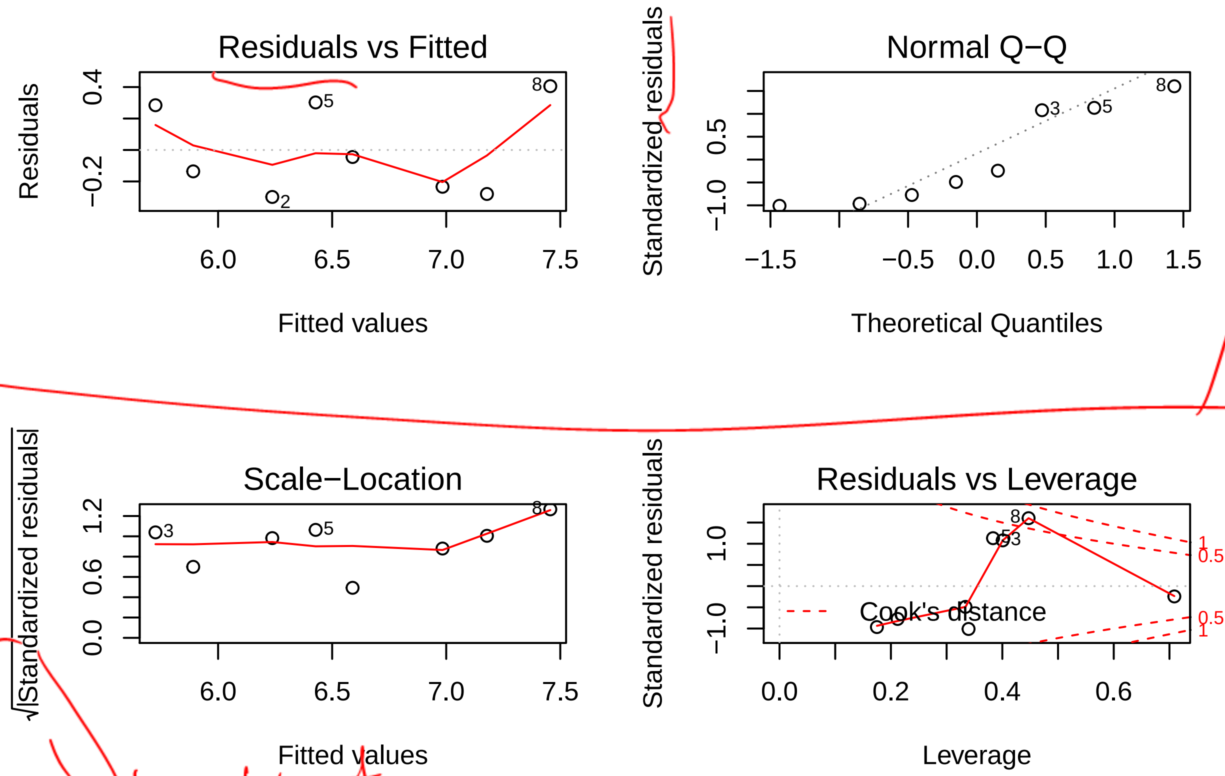
$H_0: \beta_2 = 0$

$\hat{\sigma} = \sqrt{MSResid}$

82% explained by model

significance model

```
par(mfrow = c(2, 2))
plot(Model3)
```



Full model

lm object

Confidence Intervals and Prediction Intervals

lb   up bound

```r
confint(Model3)
```

```
##                   2.5 %       97.5 %
## (Intercept) -1.22944837 13.900641358
## ht          -0.05689702  0.009400813
## tillers     -0.13777084  0.438396122
```

← 95% CI for β_1

```r
NewData <- data.frame(ht = 80, tillers = 17)
predict(Model3, NewData, interval = "confidence")
```

made up

```
##        fit      lwr      upr
## 1 6.991063 6.425802 7.556324
```

```r
predict(Model3, NewData, interval = "prediction")
```

exact spelling for predictors

```
##        fit      lwr      upr
## 1 6.991063 5.949278 8.032848
```

← wider

Additional Hypothesis Testing

We illustrate the use of using lht() from the car package.

```r
#Test1: B2 = 0.1
c1 <- c(0, 0, 1)
lht(Model3, c1, rhs = c(0.1))
```

```
## Linear hypothesis test
```

```
##
## Hypothesis:
## tillers = 0.1
##
## Model 1: restricted model
## Model 2: yield ~ ht + tillers
##
##   Res.Df     RSS Df Sum of Sq      F Pr(>F)
## 1      6 0.60281
## 2      5 0.57946  1  0.023358 0.2015 0.6723
```

– FTR $H_0: \beta_2 = .1$

```
#Test2: B1 = B2 = 0
c2 <- matrix(c( 0, 1, 0,
                0, 0, 1), nrow=2, byrow=TRUE)
lht(Model3, c2, rhs = c(0, 0))
```

```
## Linear hypothesis test
##
## Hypothesis:
## ht = 0
## tillers = 0
##
## Model 1: restricted model
## Model 2: yield ~ ht + tillers
##
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1      7 3.2115
## 2      5 0.5795  2    2.6321 11.356 0.01383 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Reject H_0

```
#Null Model contains no predictors (not usually of interest!)
Model0 <- lm(yield ~ 1, data = Rice)
anova(Model0, Model3)
```

```
## Analysis of Variance Table
##
## Model 1: yield ~ 1
## Model 2: yield ~ ht + tillers
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1      7 3.2115
## 2      5 0.5795  2    2.6321 11.356 0.01383 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#Test3: B1=B2 or B1-B2 = 0
c3 <- c(0, 1, -1)
lht(Model3, c3, rhs=c(0))
```

$H_0: \beta_1 = \beta_2$

$\beta_1 - \beta_2 = 0$

why?

```
## Linear hypothesis test
##
## Hypothesis:
## ht - tillers = 0
##
## Model 1: restricted model
## Model 2: yield ~ ht + tillers
```

```
##
##   Res.Df     RSS Df Sum of Sq      F Pr(>F)
## 1      6 0.91442
## 2      5 0.57946  1   0.33497 2.8904 0.1499
```

FTR H_0

```
#Test4
c4 <- c(1, 80, 17)
lht(Model3, c4, rhs=c(7))
```

use predict()

interval = "confidence"

```
## Linear hypothesis test
##
## Hypothesis:
## (Intercept)  + 80 ht  + 17 tillers = 7
##
## Model 1: restricted model
## Model 2: yield ~ ht + tillers
##
##   Res.Df     RSS Df  Sum of Sq      F Pr(>F)
## 1      6 0.57965
## 2      5 0.57946  1 0.00019142 0.0017 0.9692
```

anova vs Anova

When there is just a single predictor, there is no difference between anova() and Anova(). But when there are now two predictors, there is a difference between anova() and Anova() from the car package. In general, we will be using Anova(). anova() can be used to compare a reduced vs full model.

```
anova(Model1)
```

always Type I

```
## Analysis of Variance Table
##
## Response: yield
##           Df  Sum Sq Mean Sq F value   Pr(>F)
## ht         1 2.42357 2.42357  18.455 0.005116 **
## Residuals  6 0.78794 0.13132
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Anova(Model1, type = 3)
```

```
## Anova Table (Type III tests)
##
## Response: yield
##              Sum Sq Df F value    Pr(>F)
## (Intercept) 19.0239  1 144.864 1.996e-05 ***
## ht           2.4236  1  18.455  0.005116 **
## Residuals    0.7879  6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(Model3)
```

```
## Analysis of Variance Table
##
## Response: yield
##           Df  Sum Sq Mean Sq F value   Pr(>F)
```

```
## ht          1 2.42357 2.42357 20.9125 0.005985 **
## tillers     1 0.20848 0.20848  1.7989 0.237538
## Residuals  5 0.57946 0.11589
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Anova(Model3, type = 3)
```

```
## Anova Table (Type III tests)
##
## Response: yield
##              Sum Sq Df F value  Pr(>F)
## (Intercept) 0.53711  1  4.6346 0.08395 .
## ht          0.39304  1  3.3914 0.12489
## tillers     0.20848  1  1.7989 0.23754
## Residuals   0.57946  5
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(Model2, Model3)
```

```
## Analysis of Variance Table
##
## Model 1: yield ~ tillers
## Model 2: yield ~ ht + tillers
##   Res.Df     RSS Df Sum of Sq      F Pr(>F)
## 1      6 0.97249
## 2      5 0.57946  1   0.39304 3.3914 0.1249
```