

Homework 6

Yunting Chiu

2021-03-20

1. (6.3) A student stated: “Adding predictor variables to a regression model can never reduce R^2 , so we should include all available predictor variables in the model.” Comment.
 - Not really. If the model have a bigger R^2 value, which means the fitter is better. However, better fitting does not necessarily imply the fitted model is a useful one.
2. (6.4) Why is it not meaningful to attach a sign to the coefficient of multiple correlation R , although we do so for the coefficient of simple correlation r_{12} ?
 - The range of simple correlation coefficient is -1 to 1. The multiple correlation shows how any variable can be predicted by using a set of other variables. There will be several independent variables, each of which will have a different effect. The multiple correlation coefficient will be more complicated than the simple one because the range is not limited to -1 to 1.
3. (6.27) In a small-scale regression study, the following data were obtained,

Y	X1	X2
42.0	7.0	33.0
33.0	4.0	41.0
75.0	16.0	7.0
28.0	3.0	49.0
91.0	21.0	5.0
55.0	8.0	31.0

Assume the standard multiple regression model with independent normal error terms. Compute \mathbf{b} , \mathbf{e} , \mathbf{H} , SSErr , R^2 , s_b^2 , \hat{Y} for $X_1 = 10$, $X_2 = 30$. You can do the computations using software or by hand, although it would be lengthy to do them by hand.

- The first column is `scalar` (1, 1, 1, 1, 1, 1)

```
scalar <- c(1, 1, 1, 1, 1, 1)
Y <- c(42, 33, 75, 28, 91, 55)
X1 <- c(7, 4, 16, 3, 21, 8)
X2 <- c(33, 41, 7, 49, 5, 31)

# namely X is the matrix
X <- matrix(c(scalar, X1, X2), 6, 3, byrow = FALSE)
X
```

```
##      [,1] [,2] [,3]
## [1,]    1    7   33
## [2,]    1    4   41
## [3,]    1   16    7
## [4,]    1    3   49
## [5,]    1   21    5
```

```
## [6,]    1    8   31
```

b: slope

$$\hat{\beta} = (X^T X)^{-1} (X^T Y)$$

```
library(matlib)
b = inv(t(X) %*% X) %*% t(X) %*% Y
b
```

```
##           [,1]
## [1,] 33.9321020
## [2,]  2.7847707
## [3,] -0.2643979
```

H: Hat

- H is called the hat-matrix (because it transforms y to \hat{y})

$$H = X(X^T X)^{-1} X^T$$

```
H = X %*% inv(t(X) %*% X) %*% t(X)
H
```

```
##           [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## [1,]  0.23143639  0.25168006  0.21178834  0.1488734 -0.05475455  0.21099418
## [2,]  0.25168006  0.31240977  0.09437951  0.2662835 -0.14787196  0.22314063
## [3,]  0.21178834  0.09437951  0.70442097 -0.3191731  0.10446756  0.20412257
## [4,]  0.14887339  0.26628346 -0.31917314  0.6142637  0.14143589  0.14834214
## [5,] -0.05475455 -0.14787196  0.10446756  0.1414359  0.94040059  0.01632796
## [6,]  0.21099418  0.22314063  0.20412257  0.1483421  0.01632796  0.19708945
```

e: residuals

$$e = Y - \hat{Y} = Y - H * Y$$

```
e <- Y - H %*% Y
e
```

```
##           [,1]
## [1,] -2.70036663
## [2,] -1.23087135
## [3,] -1.63764825
## [4,] -1.33091751
## [5,] -0.09029763
## [6,]  6.98606687
```

SSErr

$$SSE = e^T * e$$

- **SSErr** is error sum of squares.

```
SSE <- t(e) %*% e
SSE
```

```
##           [,1]
## [1,] 62.07354
```

```
# SSErr <- sum(e^2)
# SSErr
```

R^2

$$R^2 = \frac{SS_{Regression}}{SSTotal} = 1 - \frac{SSE_{Error}}{SSTotal}$$

```
SST <- sum((Y - mean(Y))^2) # total sum of squares
SST
```

```
## [1] 3072
```

```
R2 <- 1 - SSE/SST # R^2
R2
```

```
##           [,1]
## [1,] 0.9797938
```

s_b^2

$$MSE = \frac{SSE}{n-2}$$

```
MSE <- SSE/ (6-2)
MSE
```

```
##           [,1]
## [1,] 15.51839
```

$$S^2\{b\} = MSE(X^T * X)^{-1}$$

```
s2b <- MSE[1,1] * inv(t(X) %*% X)
s2b
```

```
##           [,1]      [,2]      [,3]
## [1,] 536.60338 -25.6191888 -10.1962033
## [2,] -25.61919   1.2462499   0.4830506
## [3,] -10.19620   0.4830506   0.1968509
```

$E\{Y|X_1 = 10, X_2 = 30\}$

- The equation below is an unbiased estimator of estimated of Y_n .

$$\hat{Y}_h = X_h^T * b$$

```
Xone <- c(1, 10, 30)
yhat <- t(Xone) %*% b
yhat
```

```
##           [,1]
## [1,] 53.84787
```

- Reference:
 - <https://stats.stackexchange.com/questions/352130/convert-the-beta-coefficient-from-matrix-to-scalar-notation-in-ols-regression>
 - <https://www.stat.cmu.edu/~cshalizi/mreg/15/lectures/13/lecture-13.pdf>
 - <http://fs2.american.edu/baron/www/Handouts/review%20-%20regression.pdf>

4. (Computer project, #6.5—#6.8) Dataset “Brand preference” is available on our Blackboard, on <http://statweb.lsu.edu/EXSTWeb/StatLab/DataSets/NKNWData/CH06PR05.txt>, and here:

```
brand <- read.table("./data/CH06PR05.txt")
brand %>%
  rename(brand = "V1", moisture = "V2", sweetness = "V3") -> brand
brand
```

```
##      brand moisture sweetness
## 1      64         4          2
## 2      73         4          4
## 3      61         4          2
## 4      76         4          4
## 5      72         6          2
## 6      80         6          4
## 7      71         6          2
## 8      83         6          4
## 9      83         8          2
## 10     89         8          4
## 11     86         8          2
## 12     93         8          4
## 13     88        10          2
## 14     95        10          4
## 15     94        10          2
## 16    100        10          4
```

```
# Y, X1, X2
```

It was collected to study the relation between degree of brand liking (Y) and moisture content (X1) and sweetness (X2) of the product. (a) Fit a regression model to these data and state the estimated regression function. Interpret b_1 .

$$\hat{Y} = 37.6500 + 4.4250X_1 + 4.3750X_2$$

- The slope b_1 is 4.425, meaning that the mean response degree of brand liking is increase by 4.425 with 1 unit increase of moisture when sweetness is held constant.

```
reg <- lm(brand ~ moisture + sweetness, data = brand)
summary(reg)
```

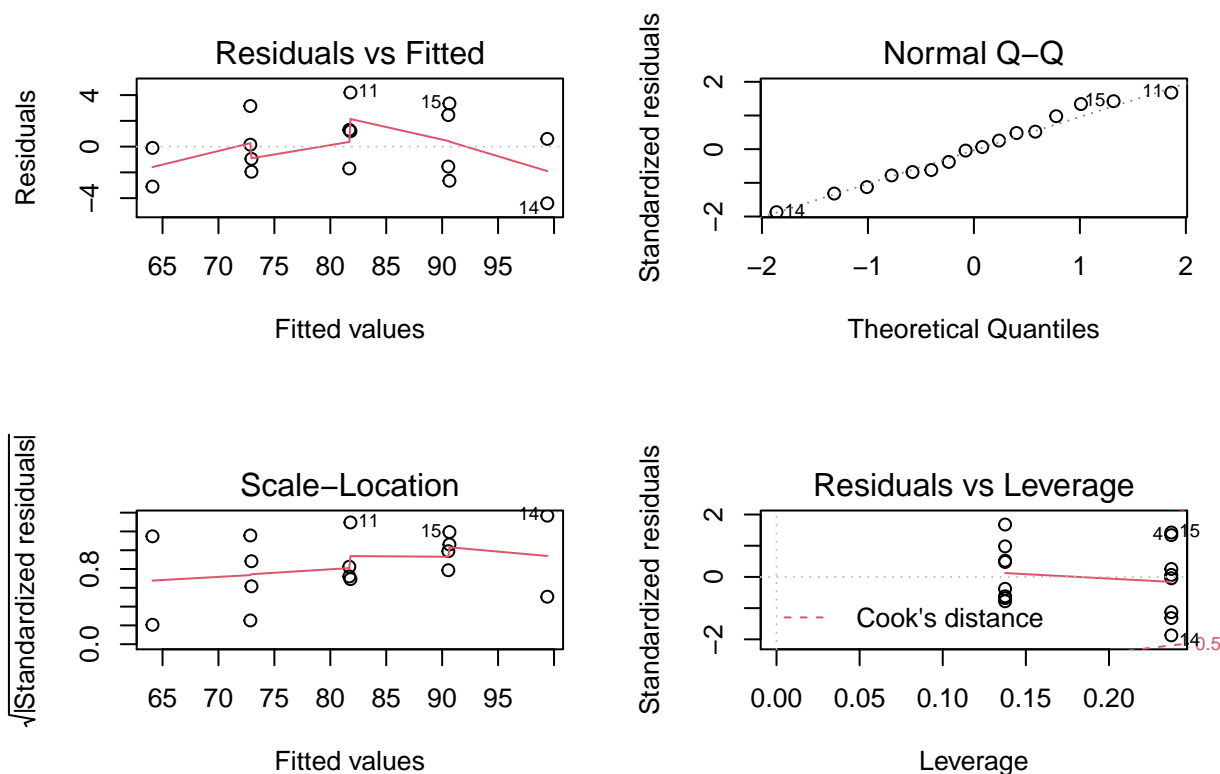
```
##
## Call:
## lm(formula = brand ~ moisture + sweetness, data = brand)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -4.400 -1.762  0.025  1.587  4.200
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   37.6500     2.9961  12.566 1.20e-08 ***
## moisture       4.4250     0.3011  14.695 1.78e-09 ***
## sweetness     4.3750     0.6733   6.498 2.01e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.693 on 13 degrees of freedom
## Multiple R-squared:  0.9521, Adjusted R-squared:  0.9447
```

```
## F-statistic: 129.1 on 2 and 13 DF, p-value: 2.658e-09
```

(b) Obtain residual plots. What information do they provide? Plot residuals against fitted values, against each predictor, and against the product of predictors.

- We need to make residual plots to see if the data points meet the linear assumptions or not.
- Residuals vs Fitted plot: a strong pattern indicates non-linearity in the data
- Normal QQ plot: Some outliers can be found in the right upper area. It is necessary to conduct further testing.
- Scale-Location: The residuals seem symmetric and the red line is approximately horizontal. However, the average magnitude of the standardized residuals is not changing much as a function of the fitted values.
- Residual vs Leverage: There are some potential outliers can be found in the right side. Reference: <https://boostedml.com/2019/03/linear-regression-plots-scale-location-plot.html>

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



(c) Test homoscedasticity.

- With a high p-value 0.42877, there is no evidence of non-constant variance.

```
library(car)
```

```
## Loading required package: carData
```

```
##
```

```
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
```

```
##
```

```
## recode
```

```
## The following object is masked from 'package:purrr':
```

```
##
##      some
ncvTest(reg)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.6261627, Df = 1, p = 0.42877
```

(d) Conduct a formal lack of fit test.

- We conclude that the p-value is 0.3843, we fail to reject the H_0 , meaning that there is no evidence of lack of fit. Thus, there should be no significant difference in error between estimates from the reduced (reg) model and the full model.
- Reference: <https://stats.stackexchange.com/questions/339331/difference-between-full-model-and-reduced-model-in-one-way-anova>

$$FullModel : Y_{ij} = \mu_j + e_{ij}$$

$$ReducedModel : Y_{ij} = \mu + e_{ij}$$

```
full <- lm(brand ~ as.factor(moisture) + as.factor(sweetness), data = brand)
anova(reg, full)
```

```
## Analysis of Variance Table
##
## Model 1: brand ~ moisture + sweetness
## Model 2: brand ~ as.factor(moisture) + as.factor(sweetness)
##   Res.Df  RSS Df Sum of Sq    F Pr(>F)
## 1      13 94.30
## 2      11 79.25  2     15.05 1.0445 0.3843
```

(e) Test whether the proposed linear regression model is significant. What do the results of the ANOVA F-test imply about the slopes?

- Let we set $H_0: \beta_1 = \beta_2 = 0$ vs $H_a: \beta_1 = \beta_2 \neq 0$.
- We found both p-values (β_1 and β_2) are smaller than significant level, implying that we are in favor of H_a .

```
anova(reg)
```

```
## Analysis of Variance Table
##
## Response: brand
##           Df Sum Sq Mean Sq F value    Pr(>F)
## moisture   1 1566.45  1566.45  215.947 1.778e-09 ***
## sweetness   1  306.25   306.25   42.219 2.011e-05 ***
## Residuals  13   94.30     7.25
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(reg)
```

```
##
## Call:
## lm(formula = brand ~ moisture + sweetness, data = brand)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -4.400 -1.762  0.025  1.587  4.200
```

```
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  37.6500     2.9961  12.566 1.20e-08 ***
## moisture     4.4250     0.3011  14.695 1.78e-09 ***
## sweetness    4.3750     0.6733   6.498 2.01e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.693 on 13 degrees of freedom
## Multiple R-squared:  0.9521, Adjusted R-squared:  0.9447
## F-statistic: 129.1 on 2 and 13 DF,  p-value: 2.658e-09
```

(f) Estimate both slopes simultaneously using the Bonferroni procedure with at least a 99 percent confidence level.

- The Bonferroni test is a statistical test used to reduce the instance of a false positive (type 1 error)
- To perform the Bonferroni correction, we need to divide the α level by the number of tests that are being executed.
- We need to test 2 slopes, and the significance level is 0.01. So, $0.01/2 = 0.005$. Throughout the Bonferroni procedure, the adjusted *alpha* level is 0.005 with two slopes.
- Reference: <https://toptipbio.com/bonferroni-correction-method/>

```
confint(reg, level = (1-0.005))
```

```
##           0.25 %    99.75 %
## (Intercept) 27.545738 47.754262
## moisture     3.409483  5.440517
## sweetness    2.104236  6.645764
```

(g) Report R2 and adjusted R2. How are they interpreted here?

- It means that 95 % of the variation in the **brand** is explained by the **moisture** and **sweetness** independent variables.

```
summary(reg)$r.square
```

```
## [1] 0.952059
```

- It means that 94 % of the variation in the **brand** is explained by model addition that are significant of **moisture** and **sweetness**.

```
summary(reg)$adj.r.square
```

```
## [1] 0.9446834
```

- Reference: <https://discuss.analyticsvidhya.com/t/difference-between-r-square-and-adjusted-r-square/264/2>

(h) Calculate the squared correlation coefficient between Y_i and \hat{Y} . Compare with part (g).

- The squared correlation coefficient between Y_i and \hat{Y} is R^2 , which is 0.952059.

(i) Obtain a 99% confidence interval for E(Y) when $X_1 = 5$ and $X_2 = 4$. Interpret it.

- We have 99 % confidence to conclude that the **mean** of degree of **brand liking** is between 73.88111 to 80.66889 when the **moisture content** is 5 and the **sweetness** of the product is 4.

```
predict(reg, tibble(moisture = 5, sweetness = 4), interval = "confidence", level = 0.99)
```

```
##           fit          lwr          upr
## 1 77.275 73.88111 80.66889
```

(j) Obtain a 99% prediction interval for a new observation Y when $X_1 = 5$ and $X_2 = 4$. Interpret it.

- We have 99 % confidence to conclude that **a next new observation** of degree of brand liking is between 73.88111 to 80.66889 when the moisture content is 5 and the sweetness of the product is 4.

```
predict(reg, tibble(moisture = 5, sweetness = 4), interval = "prediction", level = 0.99)
```

```
##      fit      lwr      upr
## 1 77.275 68.48077 86.06923
```