Anthony Cunningham

Due 2/17/2017

STAT 3200

**Homework 3**

#1. > library(car)

> attach(Robey)

> plot(tfr ~ contraceptors)

> fit = lm(tfr ~ contraceptors, data=Robey)

> abline(fit)



\* I would say that it is sensible to assume that fertility rate has a linear relationship with % of women that use contraception just by looking at the scatter plot. Once a regression line is superimposed, it becomes easier to see that tfr and contraceptors may have a negatively linear relationship. Most observations don’t deviate too far from the regression line, while there is a nice balance of observations both above and below the regression line, implying normally-distributed errors. Yes, the simple linear regression model fit looks appropriate.

#2. Simple Linear Regression Model: Y = b0 + b1X + e

\*e ~ N(0,o2) Y = total fertility rate (children/woman)

X = Percent of contraceptors among married women of childbearing age

Assumptions: -linear relationship between Y and X (ie E(Y|X) lie on a straight line)

-constant variance of errors (similar spread around the line for each x)

-normally-distributed errors

-independent errors

#3. > summary(fit) Model: Y = 6.875 – 0.05841X

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 6.875085 0.156860 43.83 <2e-16 \*\*\*

contraceptors -0.058416 0.003584 -16.30 <2e-16 \*\*\*

Residual standard error: 0.5745 on 48 degrees of freedom

**Multiple R-squared: 0.847**, Adjusted R-squared: 0.8438

F-statistic: 265.7 on 1 and 48 DF, p-value: < 2.2e-16

\* Method used to obtain model fit was the Least Squares Method, which minimizes the sum of the squared residuals.

> mean(contraceptors)

[1] 37.44 \* xsamplemean = 37.44

Ysamplemean = 6.875 – 0.05841(37.44) = 4.688

\*Point(37.44, 4.688) is on the regression line since I used the regression model to calculate Ysamplemean, which is a formulaic representation of the regression line.

#4. \*b0 = 6.875 = the amount of children per woman if 0% of married, child-bearing women in a given country do not use contraception

b1 = -0.05841 = the change in the amount of children per woman in a given country if 1% more married, child-bearing women use contraception

SE = 0.5745 = the sample standard deviation sigma-hat, or, the average deviation from the fitted regression line in terms of children born per woman at a given contraceptor rate (x)

R2 = 0.847 = the percentage of total variation in total fertility rate that is explained by the predictor variable, contraceptor rate

#5. > cor(tfr, contraceptors)

[1] -0.9203109

\* In simple linear regression, the correlation coefficient *r* is simply the square root of the coefficient of determination *R2*.

#6. > anova(fit)

Analysis of Variance Table

Response: tfr

Df Sum Sq Mean Sq F value Pr(>F)

contraceptors 1 87.672 87.672 265.67 < 2.2e-16 \*\*\*

Residuals 48 15.840 0.330

\* R2 = RegSS/TSS = 87.672/(87.672 + 15.840) = 87.672/103.512 = 0.847 = R2

\* Yes, the calculated R2 from the table is the exact same as the R2 in R’s summary output.

#7. H0: B1 = 0

Ha: B1 ≠ 0

Test Statistic: t = (b1 – 0)/(SE(b1 hat)) = -0.058416/0.003584 = -16.3 = t

Degrees of Freedom = n – 2 = 50 - 2 = 48 = df

> 2\*pt(16.3, (50-2), lower.tail=FALSE)

[1] 3.367736e-21

\* 2-sided p-value = 3.368\*10-21

\* all values verified by R summary output table

Conclusion: Reject H0: B1 = 0 since sample data is statistically significant at 0.05 level

(ie. p-value < 0.05)

#8. > confint(fit)

2.5 % 97.5 %

(Intercept) 6.55969710 7.19047384

contraceptors -0.06562173 -0.05120976

\* In this case, we once again reject H0 since 0 is not within the range of plausible values of the 95% confidence interval of b1.

#9. Model: Y = 6.875 – 0.05841X Y(x = 50)= 6.875 – 0.05841(50) = 6.875 – 2.9205 = 3.9545 = Y(x = 50)

#10. > predict(fit, newdata=data.frame(contraceptors=c(40, 70)), se.fit=TRUE)

$fit \* x = 40 is closer to the sample mean x-bar = 37.44 than

1 2 x = 70. The standard error at x = 40 is therefore smaller

4.538456 2.785983 than at x = 70 since the term (xi - xsamplemean)2 lies in the

numerator of the standard error formula, meaning that the

$se.fit farther away xi from the sample mean, the bigger the

1 2 standard error will be.

0.08175771 0.14218783

#11. > predict(fit, newdata=data.frame(contraceptors=50), interval="confidence", level = 0.95)

fit lwr upr \*E(Y|x=50) = 3.954

3.954298 3.767553 4.141043 95% CI(Y|x=50): [3.768, 4.141]

> predict(fit, newdata=data.frame(contraceptors=50), interval="prediction", level = 0.95)

fit lwr upr \*E(Y|x=50) = 3.954

3.954298 2.784265 5.124332 95% PI(Ynew|x=50): [2.784, 5.124]

\* The prediction interval is wider than the confidence interval since, in the standard error formulas, the only difference is that there is a 1 that is added in the prediction interval’s standard error formula, whereas the confidence interval’s standard error formula does not have that added 1. This makes the prediction interval wider than the confidence interval holding all else constant.

#12. > par(mfrow=c(2,1))

> plot(fit$fitted.values, fit$residuals)

> abline(h=0)

> qqnorm(fit$residuals)

> qqline(fit$residuals)



\* I think that it is safe to assume constant variance and normality in this sample. The data points in the residual plot are split pretty evenly below and above 0, indicating that the constant variance assumption has not been violated. Most of the data points on the QQ plot are right near the QQ normal line, indicating that the normality assumption has not been violated.

#13. > sRobey = Robey[with(Robey, order(contraceptors)),]

> newdata1=sRobey[23:28,]

> newfit1 = lm(tfr ~ contraceptors, newdata1)

> summary(newfit1)

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 8.10188 3.19594 2.535 0.0643 .

Contraceptors -0.07859 0.07737 -1.016 0.3672

Residual standard error: 0.6511 on 4 degrees of freedom

Multiple R-squared: 0.2051, Adjusted R-squared: 0.006333

F-statistic: 1.032 on 1 and 4 DF, p-value: 0.3672

> plot(tfr ~ contraceptors, data=newdata1, xlim=c(0,80), ylim=c(1.5,7.5))

> abline(newfit1)



#14. > up3obs = sRobey[1:3, ]

> low3obs = sRobey[48:50, ]

> newdata2 = rbind(up3obs, low3obs)

> newfit2 = lm(tfr ~ contraceptors, newdata2)

> summary(newfit2)

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 7.437687 0.146636 50.72 9.04e-07 \*\*\*

Contraceptors -0.071118 0.002771 -25.67 1.37e-05 \*\*\*

Residual standard error: 0.2378 on 4 degrees of freedom

Multiple R-squared: 0.994, Adjusted R-squared: 0.9925

F-statistic: 658.8 on 1 and 4 DF, p-value: 1.369e-05

> plot(tfr ~ contraceptors, data=newdata2)

> abline(newfit2)



#15.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Est. intercept | Est. slope | SE (slope) | SE(intercept) | p-value (slope) |
| Middle 6 obs. | 8.102 | -0.0786 | 0.0774 | 3.196 | 3.67 \* 10-1 |
| Outer 6 obs. | 7.438 | -0.0711 | 0.00277 | 0.147 | 1.37 \* 10-5 |

As this analysis shows, it is helpful to have a great spread among one’s predictor variables. If one’s x-values are clustered near one another, then we can only infer upon the range of x-values in the sample, which isn’t very helpful nor insightful. In this example, the middle 6 observations only cover from x = [30, 45] approximately. It is very difficult to sense a trend if the data is clustered near one another and only on a relatively small range of x-values. What about the values when x <= 20 or x >=60? The lack of dispersion among the x-values disallows us to answer this. This can be seen in the respective standard errors for both the slope and y-intercepts. It is no coincidence that the standard errors for both slope and y-intercept are smaller in the subset with more dispersed x-values, which means smaller variability and greater accuracy when inferring about a sample. With the term (xi – xsamplemean)2 found in the denominator in the formula for standard error (for both slope and y-intercept), it is easily shown that the larger the distance from the sample mean, the smaller the standard errors become. Smaller standard errors mean that we can compute narrower (and, therefore, more focused) confidence and prediction intervals for Y at a given x, which makes our inference more precise.