Regression Analysis - STAT 482 - Exam 2: Due Dec 14, 2021

Alex Towell (atowell@siue.edu)

Problem 1

A beer distributor is interested in the amount of time to service its retail outlets. Two factors are thought to influence the delivery time (y) in minutes: the number of cases delivered (x_1) and the distance traveled (x_2) in miles. A random sample of delivery time data has been collected. The data is available on Blackboard as a csv file.

P1: (a)

Provide an interpretation of a regression coefficient in a multiple regression model.

P1: (b)

Compute b_1, b_2, the estimated the regression coefficients for the delivery time data.

```
data1 = read.csv('exam2-1.csv')
head(data1)
```

```
##   cases distance time
## 1    10       30   24
## 2    15       25   27
## 3    10       40   28
## 4    20       18   33
## 5    25       22   35
## 6    18       31   32
```

```
data1.m12 = lm(time ~ cases + distance, data=data1)
data1.m12
```

```
##
## Call:
## lm(formula = time ~ cases + distance, data = data1)
##
## Coefficients:
## (Intercept)        cases     distance
##      5.8933       0.9206       0.3286
```

P1: (c)

Compute t statistics for testing the effect of each input variable. Explain what type of effect is being tested here.

```
coef(summary(data1.m12))
```

```
##              Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) 5.8932559 2.62973524  2.241007 4.471588e-02
## cases       0.9206362 0.06870747 13.399360 1.402168e-08
## distance    0.3286414 0.06589144  4.987620 3.157935e-04
```

P1: (d)

Compute $\mathrm{SS_R}(X_1)$ and $\mathrm{SS_R}(X_2|X_1)$. Explain what each sum of squares represents.

```
anova(data1.m12)
```

```
## Analysis of Variance Table
##
```

```
## Response: time
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## cases      1 308.008 308.008 154.901 3.217e-08 ***
## distance   1  49.464  49.464  24.876 0.0003158 ***
## Residuals 12  23.861   1.988
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

P1: (e)

Test for a marginal effect of x_2 against a model which includes no other input variables. (Compute the test statistic and p-value.) Provide an interpretation of the result, stated in the context of the problem.

```
data1.m0 = lm(time ~ 1, data=data1)
data1.m2 = lm(time ~ distance, data=data1)
anova(data1.m2)
```

```
## Analysis of Variance Table
##
## Response: time
##           Df Sum Sq Mean Sq F value Pr(>F)
## distance   1   0.47  0.4672  0.0159 0.9014
## Residuals 13 380.87 29.2974
```

```
anova(data1.m0,data1.m2)  # equivalent to anova(data1.m2)
```

```
## Analysis of Variance Table
##
## Model 1: time ~ 1
## Model 2: time ~ distance
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     14 381.33
## 2     13 380.87  1   0.46715 0.0159 0.9014
```

P1: (f)

Test for a partial effect of x_2 against a model which includes x_1. (Compute the test statistic and p-value.) Provide an interpretation of the result, stated in the context of the problem.

```
data1.m1 = lm(time ~ cases, data=data1)
anova(data1.m1,data1.m12)
```

```
## Analysis of Variance Table
##
## Model 1: time ~ cases
## Model 2: time ~ cases + distance
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1     13 73.325
## 2     12 23.861  1    49.464 24.876 0.0003158 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

P1: (g)

Compute the correlation matrix. What feature of multidimensional modeling is illustrated in this problem?

```
cor(data1)
```

```
##               cases    distance        time
## cases     1.0000000 -0.40529762  0.89872861
## distance -0.4052976  1.00000000 -0.03500075
## time      0.8987286 -0.03500075  1.00000000
```

Problem 2

A bakery is interested in the best formulation for a new product. A small-scale experiment is conducted to investigate the relationship between the product satisfaction (y), and the moisture content (input 1) and sweetness (input 2) of the product. The input variables have been coded (x_1, x_2) for ease of calculation. The data is available on Blackboard as a csv file.

P2: (c)

Fit an interaction model using the coded variables. Compute the regression coefficient estimates and their standard errors.

```
data2 = read.csv('exam2-2.csv')
data2
```

```
##    moisture x1 sweetness x2 satisfaction
## 1         4 -3         2 -1           65
## 2         4 -3         4  1           75
## 3         4 -3         2 -1           61
## 4         4 -3         4  1           76
## 5         6 -1         2 -1           75
## 6         6 -1         4  1           86
## 7         6 -1         2 -1           72
## 8         6 -1         4  1           85
## 9         8  1         2 -1           87
## 10        8  1         4  1           87
## 11        8  1         2 -1           80
## 12        8  1         4  1           88
## 13       10  3         2 -1           91
## 14       10  3         4  1           96
## 15       10  3         2 -1           89
## 16       10  3         4  1           99
```

```
data2.xmod = lm(satisfaction ~ x1*x2, data=data2)
summary(data2.xmod)
```

```
##
## Call:
## lm(formula = satisfaction ~ x1 * x2, data = data2)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -2.900 -1.938 -0.500  1.938  4.950
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  82.0000     0.6435 127.433  < 2e-16 ***
## x1            3.9750     0.2878  13.813 9.94e-09 ***
## x2            4.5000     0.6435   6.993 1.45e-05 ***
## x1:x2        -0.5750     0.2878  -1.998   0.0689 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.574 on 12 degrees of freedom
## Multiple R-squared:  0.9531, Adjusted R-squared:  0.9413
## F-statistic: 81.23 on 3 and 12 DF,  p-value: 3.069e-08
```

P2: (d)

Write the estimated regression as a function of x_1 for $x_2 = 1, 0, -1$.

```
b0 = coef(data2.xmod)[1]
b1 = coef(data2.xmod)[2]
b2 = coef(data2.xmod)[3]
```

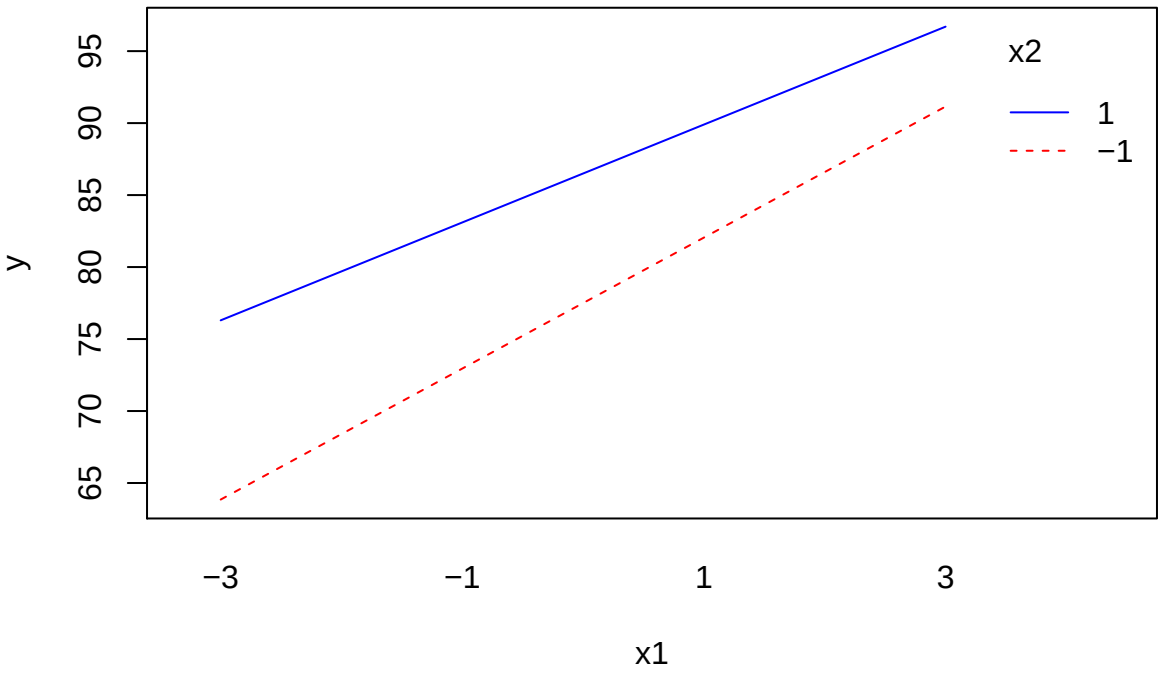
```
b12 = coef(data2.xmod)[4]
reg.est.x1 = matrix(c(b0+b2,b1+b12,  # x2=+1
                      b0,b1,         # x2=0
                      b0-b2,b1-b12), # x2=-1
                    nrow=3,byrow=T)

dimnames(reg.est.x1) = list(c("x2=+1","x2=0","x2=-1"),c("intercept","slope"))
reg.est.x1
```

```
##       intercept slope
## x2=+1      86.5 3.400
## x2=0       82.0 3.975
## x2=-1      77.5 4.550
```

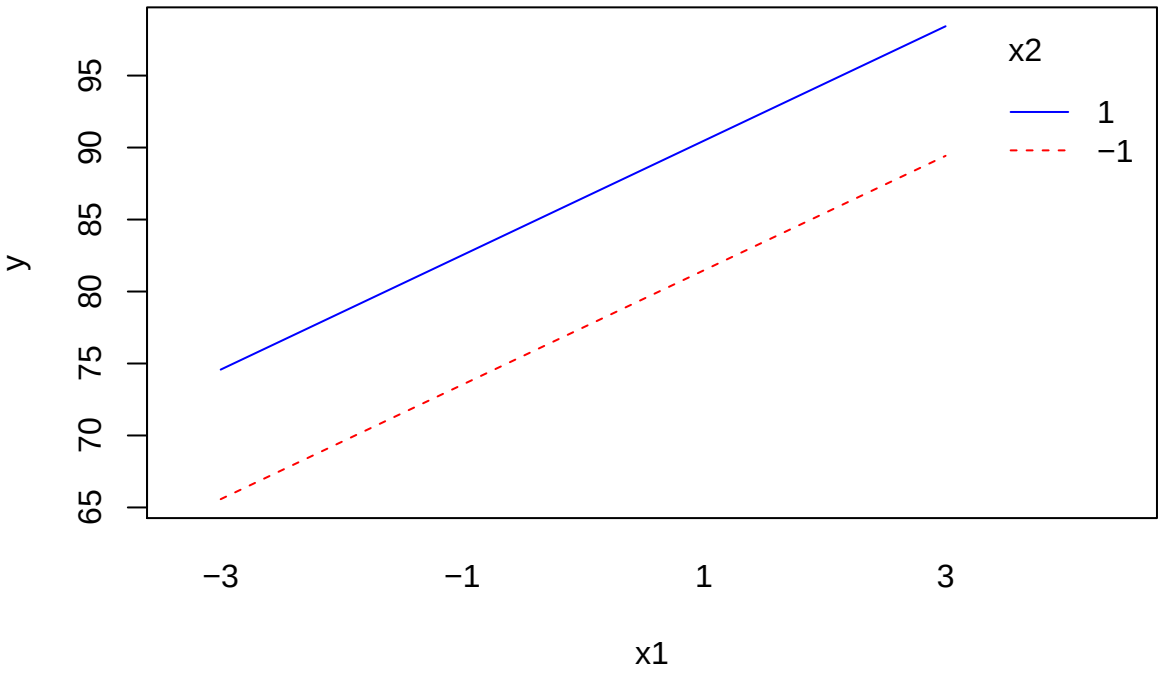
P2: (e)

Create interaction plots for both the interaction model and the additive effects model.

```
# interaction model:
data2.xmod.fits = predict(data2.xmod)
interaction.plot(data2$x1,data2$x2,data2.xmod.fits,
            col=c("red","blue"),
            trace.label="x2",
            xlab="x1",
            ylab="y")
```




```
# additive model
data2.amod = lm(satisfaction ~ x1+x2,data=data2)
data2.amod.fits = predict(data2.amod)
interaction.plot(data2$x1,data2$x2,data2.amod.fits,
            col=c("red","blue"),
            trace.label="x2",
            xlab="x1",
            ylab="y")
```



P2: (f)

Test for an interaction effect. (Compute the test statistic and p-value.)

```
anova(data2.amod, data2.xmod)
```

```
## Analysis of Variance Table
##
## Model 1: satisfaction ~ x1 + x2
## Model 2: satisfaction ~ x1 * x2
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1     13 105.95
## 2     12  79.50  1     26.45 3.9925 0.06888 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Problem 3

An engineer is interested in comparing three chemical processes (categorical input with groups A,B,C) for manufacturing a compound. It is suspected that the impurity (continuous input x) of the raw material will affect the yield (response variable y) of the product. The data is available on Blackboard as a csv file.

P3: (e)

Compute interval estimates for each of the effect parameters.

```
library(matlib)
data3 = read.csv('exam2-3.csv')
data3$process = as.factor(data3$process)

contrasts(data3$process) = contr.treatment(3,base=3)
data3.amod = lm(yield ~ impurity+process,data=data3)

b.hat = coef(data3.amod)
dfe = nrow(model.matrix(data3.amod)) - ncol(model.matrix(data3.amod))
V = vcov(data3.amod)
a = c(0,0,1,-1)

b.hat.12 = a %*% b.hat
se.12 = sqrt(a %*% V %*% a)
```

```
b.hat.12.lower = b.hat.12 - qt(.975,dfe) * se.12
b.hat.12.upper = b.hat.12 + qt(.975,dfe) * se.12

confint(data3.amod)
```

```
##                 2.5 %    97.5 %
## (Intercept) 1.5446801 5.7006292
## impurity    0.1806097 0.9967766
## process1    4.3264225 7.3832047
## process2    2.2017852 5.0353307
```

```
cat("beta2-beta1",c(b.hat.12.lower,b.hat.12.upper))
```

```
## beta2-beta1 0.5974664 3.875045
```

P3: (f)

Create a scatterplot of the data with the estimated regression lines.

```
intercept.3 = b.hat[1]
intercept.1 = b.hat[1]+b.hat[3]
intercept.2 = b.hat[1]+b.hat[4]
slope = b.hat[2]

cat(slope, intercept.1, intercept.2, intercept.3)
```

```
## 0.5886932 9.477468 7.241213 3.622655
```

```
attach(data3)

plot(impurity[process == "A"],yield[process == "A"],
     xlab='impurity',ylab='yield',pch=1,
     xlim=c(min(impurity)-1,max(impurity)+1),
     ylim=c(min(yield)-1,max(yield)+1))

points(impurity[process=="B"], yield[process=="B"], pch=2)
points(impurity[process=="C"], yield[process=="C"], pch=15)

abline(intercept.1,slope,lty=1)
abline(intercept.2,slope,lty=2)
abline(intercept.3,slope,lty=3)
```