Unit 5 – Regression and Correlation
Practice Problems (1 of 3)

Due: Thursday October 17, 2024
Last date to submit late for credit (-20 points): Thursday October 24, 2024

Before you begin. Download from the course website
simplelinear.xlsx
ers.Rdata

1.

This exercise gives you practice doing a simple linear regression using **simplelinear.xlsx.** This data set has n=31 observations of boiling points (Y=boiling) and temperature (X=temp). You will be exploring the following two simple linear models:

(i) $Y = \beta_0 + \beta_1 X$; where Y=boiling and X=temp

(ii) newy $= \beta_0 + \beta_1 X$; where newy = $100*\log_{10}(y)$ and where y=boiling and X=temp

1a. Create a new variable newy = $100*\log_{10}$(boiling)

1b. For each model, obtain:

a. The fitted line estimates of $\hat{\beta}_0$ and $\hat{\beta}_1$
b. Analysis of variance table
c. R^2 = % of the variability in the outcome explained by the fitted line
d. Scatter plot with overlay of fitted line

1c. In 3-5 sentences, write a one-paragraph interpretation of your two model fits.

#2.

Note – This question does NOT require use of software (R or otherwise!)

This exercise gives you practice working with a fitted model that is provided to you. A psychiatrist wants to know whether the level of pathology (Y) in psychotic patients 6 months after treatment could be predicted with reasonable accuracy from knowledge of pretreatment symptom ratings of thinking disturbance (X_1) and hostile suspiciousness (X_2).

2a. The least squares estimation equation involving both independent variables is given by

$$Y = -0.628 + 23.639(X_1) - 7.147(X_2)$$

Using this equation, determine the predicted level of pathology (Y) for a patient with pretreatment scores of 2.80 on thinking disturbance and 7.0 on hostile suspiciousness. How does the predicted value obtained compare with the actual value of 25 observed for this patient?

2b. Using the analysis of variance tables below, carry out the *overall* F test for each of three models: i) model with X_1 alone; ii) model with X_2 alone; and iii) model with both X_1 and X_2.

Source	DF	Sum of Squares
Regression on X_1	1	1546
Residual	51	12246

Source	DF	Sum of Squares
Regression on X_2	1	160
Residual	51	13632

Source	DF	Sum of Squares
Regression on X_1, X_2	2	2784
Residual	50	11008

2c. Based on your results in part (b), how would you rate the importance of the two variables in predicting Y?

2d. What are the R^2 values for the three regressions referred to in part (b)?

2e. Based on the above, in your opinion, which is the best model involving either one or both of the two independent variables?

#3.

Note – This question does NOT require use of software (R or otherwise!) with one exception: to obtain p-values for parts a-c. Tip – Use Art of Stat if you like!

This exercise gives you practice working with analysis of variance tables. In an an experiment to describe the toxic action of a certain chemical on silkworm larvae, the relationship of $\log_{10}$(dose) and $\log_{10}$(larva weight) to $\log_{10}$(survival) was sought. The data, obtained by feeding each larva a precisely measured dose of the chemical in an aqueous solution and then recording the survival time (ie time until death) are given in the table. Also given are relevant computer results and the analysis of variance table.

Larva	1	2	3	4	5	6	7	8
Y = $\log_{10}$(survival time)	2.836	2.966	2.687	2.679	2.827	2.442	2.421	2.602
X_1=$\log_{10}$(dose)	0.150	0.214	0.487	0.509	0.570	0.593	0.640	0.781
X_2=$\log_{10}$(weight)	0.425	0.439	0.301	0.325	0.371	0.093	0.140	0.406

Larva	9	10	11	12	13	14	15
Y = $\log_{10}$(survival time)	2.556	2.441	2.420	2.439	2.385	2.452	2.351
X_1=$\log_{10}$(dose)	0.739	0.832	0.865	0.904	0.942	1.090	1.194
X_2=$\log_{10}$(weight)	0.364	0.156	0.247	0.278	0.141	0.289	0.193

#3 - continued.

$$Y = 2.952 - 0.550\ (X_1)$$
$$Y = 2.187 + 1.370\ (X_2)$$
$$Y = 2.593 - 0.381\ (X_1) + 0.871\ (X_2)$$

Source	DF	Sum of Squares
Regression on X_1	1	0.3633
Residual	13	0.1480

Source	DF	Sum of Squares
Regression on X_2	1	0.3367
Residual	13	0.1746

Source	DF	Sum of Squares
Regression on X_1 , X_2	2	0.4642
Residual	12	0.0471

3a. Test for the significance of the overall regression involving both independent variables X_1 and X_2.

3b. Test to see whether using X_1 alone significantly helps in predicting survival time.

3c. Test to see whether using X_2 alone significantly helps in predicting survival time.

3d. Compute R^2 for each of the three models.

3e. Which independent predictor do you consider to be the best single predictor of survival time?

3f. Which model involving one or both of the independent predictors do you prefer and why?

Supplement - Learn R (nothing to turn in)
Simple Linear Regression

This illustation of R for simple linear regression is taken from last year's R lesson 06 titled, "Introduction to Linear Regression in R".

Dataset Used: **ers.Rdata**
Source: Chatterjee, S; Handcock MS and Simonoff JS A Casebook for a First Course in Statistics and Data Analysis. New York, John Wiley, 1995, pp 145-152.

Introduction to the New York Auto Club Data
The data are from the New York Auto Club. There are n=28 observations of p=12 variables. Of interest is the relationship between the number of calls to the autoclub in relationship to the weather.

This illustration considers just two variables: Y=calls and X=low. R is used to produce numerical and graphical descriptions of the data and perform a simple linear regression.

```
Initialize session
setwd("/cloud/project")                                          # Set working directory
options(scipen=999)                                              # Turn off scientific notation
rm(list = ls())                                                  # Clear the Decks

Input Rdata. Inspect
load(file="ers.Rdata")                                          # load(file="FULL NAME IN QUOTES")
str(ersdata)                                                    # str() to inspect

## 'data.frame':    28 obs. of  12 variables:
##  $ day    : int  12069 12070 12071 12072 12073 12074 12075 12076 12077 12078 ...
##  $ calls  : int  2298 1709 2395 2486 1849 1842 2100 1752 1776 1812 ...
##  $ fhigh  : int  38 41 33 29 40 44 46 47 53 38 ...
##  $ flow   : int  31 27 26 19 19 30 40 35 34 32 ...
##  $ high   : int  39 41 38 36 43 43 53 46 55 43 ...
##  $ low    : int  31 30 24 21 27 29 41 40 38 31 ...
##  $ rain   : int  0 0 0 0 0 0 1 0 1 0 ...
##  $ snow   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ weekday: int  0 0 0 1 1 1 1 0 0 1 ...
##  $ year   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ sunday : int  0 1 0 0 0 0 0 0 1 0 ...
##  $ subzero: int  0 0 0 0 0 0 0 0 0 0 ...
##  - attr(*, "datalabel")= chr ""
##  - attr(*, "time.stamp")= chr ""
##  - attr(*, "formats")= chr [1:12] "%8.0g" "%8.0g" "%8.0g" "%8.0g" ...
##  - attr(*, "types")= int [1:12] 252 252 251 251 251 251 251 251 251 251 ...
##  - attr(*, "val.labels")= chr [1:12] "" "" "" "" ...
##  - attr(*, "var.labels")= chr [1:12] "" "" "" "" ...
##  - attr(*, "version")= int 8

Clean data.
library(tidyverse)                                              # select() in {dplyr} included in {tidyverse}

mydata <- ersdata %>%
          dplyr::select(low,calls)                              # select() to choose variables X=low and Y=calls

mydata$low <- as.numeric(mydata$low)                            # as.numeric( ) to convert to numeric
mydata$calls <- as.numeric(mydata$calls)

mydata                                                          # show (okay since n=28 is modest)

##    low calls
## 1   31  2298
## 2   30  1709
## 3   24  2395
```

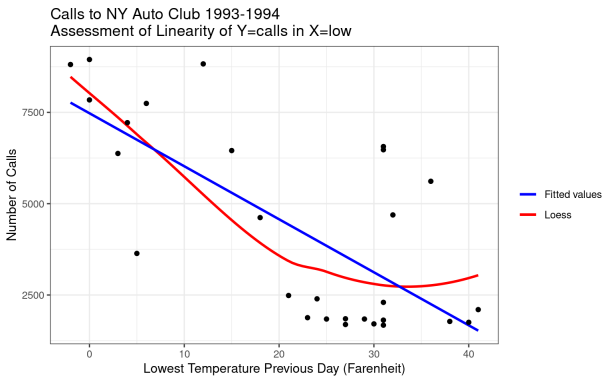
```
## 4   21  2486
## 5   27  1849
## 6   29  1842

---  several rows omitted ---

## 22  18  4619
## 23  31  6476
## 24  32  4692
## 25   5  3638
## 26   0  8947
## 27  31  6564
## 28  36  5613
```

Explore Data: Assess missing values, range, outliers

```
library(summarytools)

myvars <- c("low", "calls")                          # GOOD TO KNOW!  Handy for choosing variables to examine

descr(mydata[myvars],                                # Note - Square brackets to identify columns of dataframe
      stats=c("n.valid", "min","max"),
      transpose=TRUE)
## Descriptive Statistics
## mydata
## N: 28
##
##               N.Valid       Min       Max
## ----------- --------- --------- ---------
##       calls     28.00   1674.00   8947.00
##         low     28.00     -2.00     41.00
```

Explore Data: ggplot() to assess linearity - WITH AESTHETICS

```
library(ggplot2)

ggplot(data=mydata) +                                                    # required:  data =
  aes(x=low, y=calls) +                                                  # required:  aes(x=, y=)
  geom_smooth(method = "loess",span=1, aes(color="Loess"), se=FALSE) +   # loess fit, se=FALSE to remove CI
  geom_smooth(method = "lm", aes(color="Fitted values"), se=FALSE) +     # linear fit, se=FALSE to remove CI
  geom_point() +
  scale_colour_manual(name="",values=c("blue", "red")) +                 # blue for fit, red for loess
  xlab("Lowest Temperature Previous Day (Farenheit)") +
  ylab("Number of Calls") +
  ggtitle("Calls to NY Auto Club 1993-1994\nAssessment of Linearity of Y=calls in X=low") +
  theme_bw()
```



The scatterplot on the previous page suggests, as we might expect, that lower temperatures are associated with more calls to the NY Auto Club. We also see that the data are a bit messy. The loess smoother suggests some departure from linearity but we won't worry about that for now.

lm() to fit linear regression model

```
lm_fit <- lm(calls ~ low, data=mydata)                                  # lm(y ~ x, data=)
```

Show Fit - Basic

```
lm_fit                                                                  # show saved model object

##
## Call:
## lm(formula = calls ~ low, data = mydata)
##
## Coefficients:
## (Intercept)          low
##      7475.8       -145.2

#(lm(calls ~ low, data=mydata))                                         # Also works. Remove hashtag to execute
```

The fitted model is thus

$$\widehat{calls} = 7475.8 - 145.2 \cdot low$$

Show Fit - summary( ) to obtain more information

```
summary(lm_fit)

##
## Call:
## lm(formula = calls ~ low, data = mydata)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -3112  -1468   -214   1144   3588
##
## Coefficients:
##             Estimate Std. Error t value       Pr(>|t|)
## (Intercept)  7475.85     704.63  10.610 0.000000000061 ***
## low          -145.15      27.79  -5.223 0.000018649091 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1917 on 26 degrees of freedom
## Multiple R-squared:  0.5121, Adjusted R-squared:  0.4933
## F-statistic: 27.28 on 1 and 26 DF,  p-value: 0.00001865
```

Tip - How to show betas and associated confidence interval limits

```
cbind(coef(lm_fit),confint(lm_fit))                                     # show betas and 95% CI - default

##                          2.5 %     97.5 %
## (Intercept) 7475.849 6027.4605 8924.23745
## low         -145.154 -202.2744  -88.03352

cbind(coef(lm_fit),confint(lm_fit,level=.90))                           # show betas and 90% CI

##                            5 %      95 %
## (Intercept) 7475.849 6274.0188 8677.6792
## low         -145.154 -192.5508  -97.7571
```

TIP! names() to show names of internal objects created by R

```
# model
cat("\nObjects in model\n")
names(lm_fit)                                                    # names(SAVED MODEL)

##
## Objects in model
##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"

#names(lm(calls ~ low, data=mydata))                             # Also works.  Remove hashtag to use.

# summary of model
cat("\nObjects in summary()\n")
names(summary(lm_fit))

##
## Objects in summary()
##  [1] "call"          "terms"         "residuals"     "coefficients"
##  [5] "aliased"       "sigma"         "df"            "r.squared"
##  [9] "adj.r.squared" "fstatistic"    "cov.unscaled"
```

TIP - Now that you know the names, here is how to display internal objects created by R

```
cat("\nBetas\n")
lm_fit$coefficients                                             # show betas

##
## Betas
## (Intercept)         low
##    7475.849    -145.154

cat("\nR-squared\n")
summary(lm_fit)$r.squared                                       # show R Squared

##
## R-squared
## [1] 0.5120567
```

Analysis of Variance Table and Overall F Test - Basic

```
anova(lm_fit)

## Analysis of Variance Table
##
## Response: calls
##           Df    Sum Sq   Mean Sq F value     Pr(>F)
## low        1 100233719 100233719  27.285 0.00001865 ***
## Residuals 26  95513596   3673600
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Putting this all together (note- some of the notes below draw from earlier output)
Remarks

- *The fitted line is* $\widehat{\text{calls}} = 7{,}475.85 - 145.15*[\text{low}]$
- *R^2 = .51 indicates that 51% of the variability in calls is explained.*
- *The overall F test significance level "PROB > F" < .0001 suggests that the straight line fit performs better in explaining variability in calls than does* $\bar{Y}$ *= average # calls*
- *From this output, the analysis of variance is the following:*

Source	Df	Sum of Squares	Mean Square
Model "Regression"	1	MSS = $\sum_{i=1}^{n}\left(\hat{Y}_i - \bar{Y}\right)^2$ = 100,233,719	MSS/1 = 100,233,719
Residual "Error"	(n-2) = 26	RSS = $\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2$ = 95,513,596	RSS/(n-2) = 3,673,600
Total, corrected	(n-1) = 27	TSS = $\sum_{i=1}^{n}\left(Y_i - \bar{Y}\right)^2$ = 195,747,315	

Report Your Model - XY Scatter with overlay fit and 95% CI of estimated MEANS
Tip - Plot your line and confidence band first, then plot your points on top

```
library(ggplot2)
ggplot(data=mydata) +                                                    # Required layer: data=
 aes(x=low, y=calls) +                                                   # Required layer: aes=
 geom_smooth(method=lm, level=.95, se=TRUE) +                            # Required layer: geom_smooth()
 geom_point() +                                                          # Required layer: geom_point()
 xlab("Lowest Temperature Previous Day (Farenheit)") +                   # Optional
 ylab("Number of Calls") +                                               # Optional
 ggtitle("Simple Linear Regression of  Y=calls on X=low\n95% CI of Means") +   # Optional
 theme_bw()                                                              # Optional
```