

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

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

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

1b. For each model, obtain:

- i. The fitted line estimates of $\hat{\beta}_0$ and $\hat{\beta}_1$
- ii. Analysis of variance table
- iii. R^2 = % of the variability in the outcome explained by the fitted line
- iv. Scatter plot with overlay of fitted line

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

```
setwd("/cloud/project")
options(scipen=1000) # scipen=1000 turns off scientific notation
rm(list=ls()) # rm(list=ls()) clears the workspace/environment
```

```
Input data.
library(readxl)
dfboiling <- read_excel("simplelinear.xlsx") # During import, I named the dataframe dfboiling for ease
```

```
Create newy
dfboiling$newy <- 100*log10(dfboiling$boiling)
```

```
Model i: y=boiling x=temp
#a) Fitted line estimates of intercept and slope
m1 <- lm(boiling ~ temp, data=dfboiling)
coefficients(m1)
## (Intercept)      temp
## -65.3429977    0.4437942
```

```
#b) Analysis of Variance (anova) table
temp <- anova(m1) # saving anova(m1) will let me suppress stars in next line of code
print(temp, signif.stars=FALSE) # option signif.stars=FALSE suppresses stars
## Analysis of Variance Table
##
## Response: boiling
##           Df Sum Sq Mean Sq F value    Pr(>F)
## temp       1  450.56   450.56   336.6 < 0.00000000000000022
## Residuals 29   38.82     1.34
```

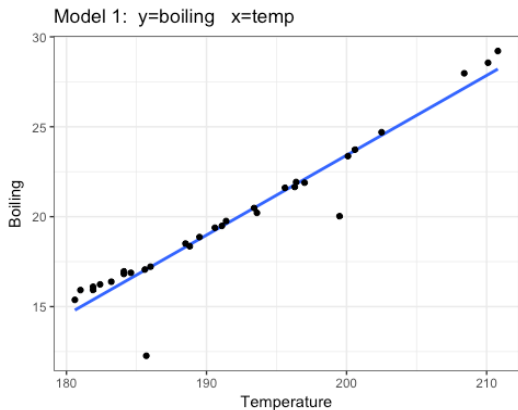
```
#c) R-squared
summary(m1)$r.squared
## [1] 0.9206773
```

Tip! Issue the command `str(m1)` to obtain list of stored quantities

Model i: Scatterplot w Overlay of Fitted Line

```
library(ggplot2)
ggplot(data=dfboiling, aes(x=temp,y=boiling)) +
  geom_smooth(method=lm, se=FALSE) +
  geom_point() +
  xlab("Temperature") +
  ylab("Boiling") +
  ggtitle("Model 1: y=boiling x=temp") +
  theme_bw()
```

ggplot requires data=, aes() and geom_NAME()
Plot line first. se=FALSE suppresses the CI band
geom_point() plots the points on top of line
Additional Layers are optional (but nice!)



Model ii: $\text{newy} = 100 \cdot \log_{10}(\text{boiling})$ x=temp

```
#a) Fitted line
m2 <- lm(newy ~ temp, data=dfboiling)
coefficients(m2)
## (Intercept)      temp
## -48.8582854    0.9261547
```

#b) Anova table

```
temp <- anova(m2)
print(temp, signif.stars=FALSE)
## Analysis of Variance Table
##
## Response: newy
##          Df Sum Sq Mean Sq F value    Pr(>F)
## temp      1 1962.2  1962.25   223.69 0.00000000000003623
## Residuals 29   254.4     8.77
```

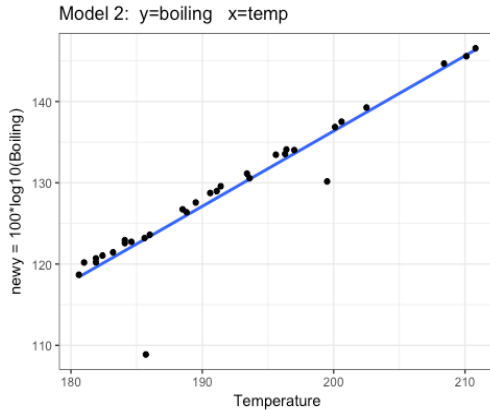
#c) R-squared

```
summary(m2)$r.squared
## [1] 0.8852331
```

Model ii: Scatterplot w Overlay of Fitted Line

```
library(ggplot2)
ggplot(data=dfboiling, aes(x=temp,y=newy)) +
  geom_smooth(method=lm, se=FALSE) +
  geom_point() +
  xlab("Temperature") +
  ylab("newy = 100*log10(Boiling)") +
  ggtitle("Model 2: y=boiling x=temp") +
  theme_bw()
```

geom_smooth() to plot line first
geom_point() to layer scatter plot on top



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

In this sample of $n=31$ observations, the scatter plot reveals two outlying values. Their inclusion may or may not be appropriate.

In an analysis that includes all $n=31$ observations, a simple linear regression of $y=\text{boiling point}$ on $x=\text{temperature}$ explains more of the variability in the outcome than a simple linear regression of $\text{newy} = 100 \cdot \log_{10}(\text{boiling})$ on $x=\text{temperature}$ ($R^2 = 92\%$ versus 89%).

Take care! - It would not make sense to compare the residual mean squares of the two models because the scales of measurement involved are different.

#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?

$\hat{Y} = 15.53$, obtained as follows:

$$\begin{aligned}\hat{Y} &= -0.628 + 23.639 \cdot X_1 - 7.147 \cdot X_2 \\ &= -0.628 + (23.639 \cdot 2.80) - (7.147 \cdot 7.0) \\ &= 15.53\end{aligned}$$

The predicted value of 15.53 is lower than 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

Model Containing X_1 ALONE

$$F = \left(\frac{\text{SSQ Regression on } X_1 / \text{DF Regression}}{\text{SSQ residual} / \text{DF Residual}} \right) = \left(\frac{1546 / 1}{12,246 / 51} \right) = 6.4385$$

on DF=1,51
p-value=0.01427

Application of the null hypothesis model has led to an extremely unlikely result (p-value = .014), prompting statistical rejection of the null hypothesis. The fitted linear model in X_1 explains statistically significantly more of the variability in level of pathology (Y) than is explained by \bar{Y} (the intercept model) alone.

R code for p-value

```
pf(6.4385, df1=1, df2=51, lower.tail=FALSE)
[1] 0.01426712
```

Model Containing X_2 ALONE

$$F = \left(\frac{\text{SSQ Regression on } X_2 / \text{DF Regression}}{\text{SSQ Residual} / \text{DF Residual}} \right) = \left(\frac{160 / 1}{13,632 / 51} \right) = 0.5986$$

on DF=1,51
p-value=0.44268

Here, application of the null hypothesis model has **not** led to an extremely unlikely result (p-value = .44). The null hypothesis is therefore **not rejected**. The fitted linear model in X_2 **does not** explain statistically significantly more of the variability in level of pathology (Y) than is explained by \bar{Y} (the intercept model) alone.

R code for p-value

```
pf(0.5986, df1=1, df2=51, lower.tail=FALSE)
[1] 0.4426844
```

Model Containing X_1 and X_2

$$F = \left(\frac{\text{SSQ regression on } X_1 \text{ and } X_2 / \text{Regression df}}{\text{SSQ residual} / \text{Residual df}} \right) = \left(\frac{2,784 / 2}{11,008 / 50} \right) = 6.3227$$

on DF=2,50
p-value=0.00356 →

Last but not least, here, application of the null hypothesis model has led to an extremely unlikely result (p-value = .00356), prompting statistical rejection of the null hypothesis. The fitted linear model in X_1 and X_2 explains statistically significantly more of the variability in level of pathology (Y) than is explained by \bar{Y} (the intercept model) alone.

```
R code for p-value
pf(6.3227,df1=2,df2=50,lower.tail=FALSE)
[1] 0.003564679
```

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

X_1 explains a significant proportion of the variability in Y when modelled as a linear predictor.
 X_2 does not. (However, we don't know if a different functional form might have been important.)

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

Total SSQ= (Regression SSQ) + (Residual SSQ) is constant.
 Therefore total SSQ can be calculated from just one anova table:

$$\text{Total (SSQ)} = 1,546 + 12,246 = 13,792$$

$$\begin{aligned} R^2(X_1 \text{ only}) &= (\text{Regression SSQ})/(\text{Total SSQ}) \\ &= (1546)/(13,792) = 0.1121 \end{aligned}$$

$$R^2(X_2 \text{ only}) = (160)/(13,792) = 0.0116$$

$$R^2(X_1 \text{ and } X_2) = (2784)/(13,792) = 0.2019$$

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

Eliminate from consideration model with X_2 only.

Compare model with X_1 alone versus X_1 and X_2 using partial F test.

$$\begin{aligned} \text{Partial } F &= \frac{\{(\text{SSQ Regression on } X_1, X_2) - (\text{SSQ Regression on } X_1)\} / \text{VDF}}{\text{SSQ Residual for model w } X_1, X_2 / \text{Residual DF}} = \frac{(2784 - 1546) / 1}{(11,008) / 50} \\ &= 5.6263 \text{ on } \text{DF} = 1, 50 \end{aligned}$$

P-value = 0.02162

Addition of X_2 to model containing X_1 is statistically significant (p-value = .02). →

More appropriate model includes X_1 and X_2

```
R code for p-value
pf(5.6263,df1=1,df2=50,lower.tail=FALSE)
[1] 0.0215831
```

#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 experiment to describe the toxic action of a certain chemical on silkworm larvae, the relationship of $\log_{10}(\text{dose})$ and $\log_{10}(\text{larva weight})$ to $\log_{10}(\text{survival})$ 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
$Y = \log_{10}(\text{survival time})$	2.836	2.966	2.687	2.679	2.827	2.442	2.421	2.602
$X_1 = \log_{10}(\text{dose})$	0.150	0.214	0.487	0.509	0.570	0.593	0.640	0.781
$X_2 = \log_{10}(\text{weight})$	0.425	0.439	0.301	0.325	0.371	0.093	0.140	0.406

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

$$Y = 2.952 - 0.550 (X_1)$$

$$Y = 2.187 + 1.370 (X_2)$$

$$Y = 2.593 - 0.381 (X_1) + 0.871 (X_2)$$

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 .

X_1 and X_2

$$F = \frac{(\text{SSQ regression on } X_1 \text{ and } X_2) / 2}{(\text{SSQ Residual}) / 12} = \frac{(0.4642) / 2}{(0.0471) / 12} = 59.18 \text{ on } DF = 2, 12$$

$$P\text{-value} < 0.0001$$

Application of the null hypothesis model has led to an extremely unlikely result ($p\text{-value} = .0001$), prompting statistical rejection of the null hypothesis. The fitted linear model in X_1 and X_2 explains statistically significantly more of the variability in $\log_{10}(\text{survival time})$ (Y) than is explained by \bar{Y} (the intercept model) alone.

R code for p-value

```
pf(59.1818, df1=2, df2=12, lower.tail=FALSE)
[1] 0.0000006083442
```

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

X_1 alone

$$F = \frac{(0.3633) / 1}{(0.1480) / 13} = \frac{(\text{SSQ Regression on } X_1) / 1}{(\text{SSQ Residual}) / 13} = 31.9115 \text{ on } DF = 1, 13$$

$$P\text{-value} = 0.00008$$

Application of the null hypothesis model has led to an extremely unlikely result ($p\text{-value} = .00008$), prompting statistical rejection of the null hypothesis. The fitted linear model in X_1 explains statistically significantly more of the variability in $\log_{10}(\text{survival time})$ (Y) than is explained by \bar{Y} (the intercept model) alone.

R code for p-value

```
pf(31.9115, df1=1, df2=13, lower.tail=FALSE)
[1] 0.00007942699
```

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

X_2 alone

$$F = \frac{(\text{SSQ Regression on } X_2) / 1}{(\text{SSQ Residual}) / 13} = \frac{(0.3367) / 1}{(0.1746) / 13} = 25.07 \text{ on } DF = 1, 13$$

$$P\text{-value} = 0.00027$$

Application of the null hypothesis model has led to an extremely unlikely result (p-value = .00027), prompting statistical rejection of the null hypothesis. The fitted linear model in X_2 explains statistically significantly more of the variability in $\log_{10}(\text{survival time})$ (Y) than is explained by \bar{Y} (the intercept model) alone.

R code for p-value

```
pf(25.07, df1=1, df2=13, lower.tail=FALSE)
```

```
[1] 0.0002399706
```

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

$$\begin{aligned} \text{TotalSSQ} &= 0.5113 \\ R^2(X_1 \text{ and } X_2) &= 0.4642 / 0.5113 = 0.9079 \\ R^2(X_1 \text{ alone}) &= 0.3633 / 0.5113 = 0.7105 \\ R^2(X_2 \text{ alone}) &= 0.3367 / 0.5113 = 0.6585 \end{aligned}$$

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

Using just the criteria of the overall F test and comparison of R^2 , the single predictor model containing X_1 is better.

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

Partial F for comparing model with X_1 alone versus model with X_1 and X_2

$$= \frac{(\Delta \text{Regression SSQ}) / (\Delta \text{Regression df})}{(\text{Full model Residual SSQ}) / (\text{Full model Residual df})} = \frac{(.4642 - .3633) / (2 - 1)}{.0471 / 12}$$

$$= 25.707006$$

R code for p-value

```
pf(25.07, df1=1, df2=12, lower.tail=FALSE)
```

```
[1] 0.0003057104
```

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

This illustration 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)

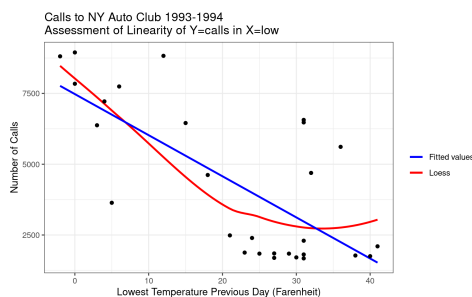
myvars <- c("low", "calls") # GOOD TO KNOW! Handy for choosing variables to examine

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) +
  aes(x=low, y=calls) +
  geom_smooth(method = "loess", span=1, aes(color="Loess"), se=FALSE) + # required: data =
  geom_smooth(method = "lm", aes(color="Fitted values"), se=FALSE) + # required: aes(x=, y=)
  geom_point() + # Loess fit, se=FALSE to remove CI
  scale_colour_manual(name="", values=c("blue", "red")) + # Linear fit, se=FALSE to remove CI
  xlab("Lowest Temperature Previous Day (Fahrenheit)") +
  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 $\hat{calls} = 7,475.85 - 145.15*[low]$
- $R^2 = .51$ 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	$MSS = \sum_{i=1}^n (\hat{Y}_i - \bar{Y})^2 = 100,233,719$	$MSS/1 = 100,233,719$
Residual "Error"	(n-2) = 26	$RSS = \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 = 95,513,596$	$RSS/(n-2) = 3,673,600$
Total, corrected	(n-1) = 27	$TSS = \sum_{i=1}^n (Y_i - \bar{Y})^2 = 195,747,315$	

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

Required Layer: data=
Required Layer: aes=
Required Layer: geom_smooth()
Required Layer: geom_point()
Optional
Optional
Optional
Optional

