In this lab we will look at how R can eliminate most of the annoying calculations involved in (a) using Chi-Squared tests to check for homogeneity in two-way tables of catagorical data and (b) computing correlation coefficients and linear regression estimates for quantitative response-explanatory variables.

1 Chi-square tests

Lets suppose we rolled a six-sided die 150 times and recorded the number of times each outcome (1-6) occured. The data is

Face	1	2	3	4	5	6
Number of times	22	21	22	27	22	36

We want to check, statistically, if the die we rolled is fair. We know how to do this using the Chi-squared statistic. If the die were fair, how many times, in 150 rolls, would we would *expect* to roll a 1, or a 3? Correct. The probability, for a fair die, of rolling any number is 1/6. So the expected number of 1's or 3's or 6's rolled is $150/6 = 25$.

To do: Perform a chi-square test of the Num Hypothesis: The Die is Fair on the data.

That may have a been a bit of work, but you should have arrived at the result that the data does NOT allow us to reject the hypothesis that die is fair. How to get to this conclusion using R? Rather simple. First let's enter the data and the expected data as a table.

```
> freq = c(22,21,22,27,22,36)
> freq

[1] 22 21 22 27 22 36

> sum(freq)

[1] 150
```

Now we can let R do all the work. Just like *t-tests* and *z-tests*, there is a one line R command to do *chi-squared-tests*.

```
> chisq.test(freq)

        Chi-squared test for given probabilities

data:  freq
X-squared = 6.72, df = 5, p-value = 0.2423
```

Simple and easy. But what does it mean?

To do: Explain the output from R. Does it agree with what you found by hand?

Problem In an effort to increase student retention many colleges have tried bloc programming. Suppose 100 students are broken into two gropus of 50 at random. One half are assingned to a block program, the other to regular college programs. The number of years the students attand collge is then measured. We want to assess whether or not block programming effects student retention. The data is:

Program	1year	2year	3year	4year	5+year
Non-Block	18	15	3	10	4
Block	10	7	7	18	8

We can enter this into R exactly as above.

To do: Work out, by hand the chi-square test for this data. Compute the test-statistic and an estimate of the p of the null-hypothesis. (What is the Null Hypothgesis?) Then enter the data in R and use it to check your work.

2 Linear Regression and Correlation

Lets see what R can do with numerical response and explanatory data. First, load the data set *airquality* which gives information on Bad Air Days in New York.

```
> data(airquality)
> attach(airquality)
> names(airquality)

[1] "Ozone"   "Solar.R" "Wind"    "Temp"    "Month"   "Day"
```

We want to see if the temperature on any given day explains the level of pollution, here measured by ozone level. First we need to clean the data up a bit to get rid of days when the measurements were unavailable.

```
> ozone = Ozone[!is.na(Ozone)]
> temperature = Temp[!is.na(Ozone)]
```

The first thing (always) to do is to look at the data.

```
> plot(temperature, ozone)
```

To do: Given the scatter plot of ozone vrs temperature, does there appear to be a linear relation between the two? If so, make a very rough prediction of the linear correlation coefficient. What is the sign?

R can, of course, easily predict the correlation coefficient. It can also easily do the linear regression analyis. For linear regression, the R command is `lm( y  x)` where y is the response and x is the explanatory variable.

```
> cor(temperature, ozone)

[1] 0.6983603

> lm( ozone ~ temperature)

Call:
lm(formula = ozone ~ temperature)

Coefficients:
(Intercept)  temperature
   -146.995        2.429

> plot(temperature, ozone)
> abline(lm( ozone ~ temperature))
```

To do: Explain what all these numbers mean. What is the equation for the best fit linear regression line produced by R? Use this to predict the ozone level when the temperature is 70 degrees and when it is 100 degrees.

To do: Use R to do Problem 10.56 on page 604 of the testbook. produced by R? In other words, input the data in R. Look at scatter plots of weight versus length and weight versus width. Can you tell, by eye, which shows a higher degree of linear correlation?

Use R to compute the best fit linear rgression line in each case. In each case, plot the line on the scatter plot. What are the RESIDUALS on the plot?

3 Confidence intervals for regression parameters

We know how to calculate regression line coefficients (slope: b_1 and intercept: b_0) from a single sample. Lets look, in depth, at how we can use these sample measurements to predict the population regression line coefficients (slope: β_1 and intercept: β_0). We will work through the process 'by hand' using R to all the dirty calculator work. Then we will see that R can do all this automatically.

Consider the data from the fish problem above. For simplicity, lets just consider the relationship between fish length (explanatory variable: x) and fish weight (response variable: y). Put the data in R (you have already done this).

```
> x = c(8.8,19.2,22.5,23.5,24.5,25.5,28.7,30.1,39.0,41.4,42.5,46.6)
> y = c(5.9,100,120,120,150,145,300,300,685,650,820,1000)
> plot(x,y)
> abline(lm(y~x))
> lm (y ~ x)

Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)            x
    -468.58        28.44
```

At the end of the day, to make our Confidence Interval for β_1 based on the observed sample parameter $b_1 = 28.44$, we will need to know the standard error in the slope:

$$\mathrm{SE}_{b_1} = \frac{s}{\sqrt{\Sigma(x_i - \bar{x})^2}}$$

where s is something like the standard deviation of the residuals. What are the residuals? The differences between the linear regression PREDICTIONS for $\tilde{y}$ and the observations y_i. Residual: $e_i = y_i - \tilde{y}_i$. To get s we need to compute:

$$s^2 = \frac{\Sigma e_i^2}{n-2}$$

OK. Let's do it. The only question left is how to get the residuals? We could go ahead and compute the least squares regression line $\tilde{y}_i = b_1 x_i + b_0$ and then at each of the observations we could compute the difference between the prediction and the observation ... or we could get the residual information from R directly. First do the least squares with R and store the data.

```
> lmdata = lm(y ~ x)                 # Least Squares Regression - STUFF stored in 'lmdata'
> residuals = lmdata$residuals       # Get the residuals and store them in 'residuals'
> plot(x,residuals)                  # Make a plot of the residuals - What are they??
```

Now we have these things, we can compute SE_{b_1}.

```
> n = length(x)
> s_square = sum( residuals^2)/(n-2)
> s = sqrt(s_square)
> denominator = sqrt( sum( (x - mean(x))^2) )
> SE = s/denominator
> SE

[1] 2.983964
```

Whew! Super, now we have the standard error in the slope. All we need to do now is find the magic value $t_\star$ corresponding to the 95% CI. We could look this up in a table, or get it from R. How many degrees of freedom? $df = n-2 = 10$ here. Let's use R to find this number (LOOK IT UP and check that R is correct).

```
> tstar = qt(1-.025,10)
> tstar

[1] 2.228139

> CI = tstar*SE
> 28.44 - CI

[1] 21.79131

> 28.44 + CI
```

```
[1] 35.08869
```

We did what we wanted to do!

Surely there must be an easier way. Someone must have told R that this si something we want to do. There is. They have. The info we computed is all there whenever you type `lm(y x)`. Try this:

```
> summary(lmdata)

Call:
lm(formula = y ~ x)

Residuals:
    Min      1Q  Median      3Q     Max
-111.60  -78.55  -49.44   53.35  224.22

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -468.583     93.164   -5.03 0.000514 ***
x             28.439      2.984    9.53 2.47e-06 ***
---
Signif. codes:  0 âĂŸ***âĂŹ 0.001 âĂŸ**âĂŹ 0.01 âĂŸ*âĂŹ 0.05 âĂŸ.âĂŹ 0.1 âĂŸ âĂŹ 1

Residual standard error: 109.8 on 10 degrees of freedom
Multiple R-squared: 0.9008,        Adjusted R-squared: 0.8909
F-statistic: 90.83 on 1 and 10 DF,  p-value: 2.466e-06
```

Do you see where the information SE_{b_1} is reported?

To Do: Repeat the above to calculate a 90% Confidence Interval for population slope fish weight versus fish width. (The data is in the text, Question 10.56). You should be able to do this in 3 LINES of R ... but for practice, also do it piece by piece the way we just did.

To Do: In order to calculate Confidence Intervals for the population intercept β_0 based on a single sample observation of the paramter b_0, we need to calculate the standard error SE_{b_0}. This has a nasty looking formula

$$\text{SE}_{b_0} = s\sqrt{\frac{1}{n} + \frac{\bar{x}^2}{\Sigma(x_i - \bar{x})^2}}$$

Using the data for fish weight and fish length, calculate SE_{b_0} and a 99% C.I. for β_0 given b_0. Do this 'by hand', then look at R's summary of the linear regression model to see if all this hard work has already been done.