**Instructions**: **Identify your test with your name, (last name followed by first name).** Use this WORD document to submit your test answers. I will add my comments directly to your .docx document. Enter your code solution below the problem statement along with any required output or displays. I prefer that you copy and paste results from the console. Be careful with the format of your report. Watch the margins and pagination. Depending on the problem, grading will be based on: 1) the correct result, 2) coding efficiency and 3) graphical presentation features (labeling, colors, size, legibility, etc.). I will be looking for well-rendered displays. Do not print and display the contents of vectors or data frames unless requested by the problem. You should be able to display each solution in fewer than ten lines of code.

**Example Problem with Solution:** Use rbinom() to generate two random samples of size 10,000 from a binomial distribution. For the first sample, use p = 0.45 and n =10. For the second sample use p = 0.55 and n = 10.

1. Convert the sample frequencies to sample proportions and compute the mean number of successes for each sample. Present these statistics.

> set.seed(123)

> sample.one <- table(rbinom(10000, 10, 0.45))/10000

> sample.two <- table(rbinom(10000, 10, 0.55))/10000

> successes <- (seq(0, 10))

> sum(sample.one\*successes)

[1] 4.4827

> sum(sample.two\*successes)

[1] 5.523

1. Present the proportions in a vertical side-by-side barplot color coding the two samples.

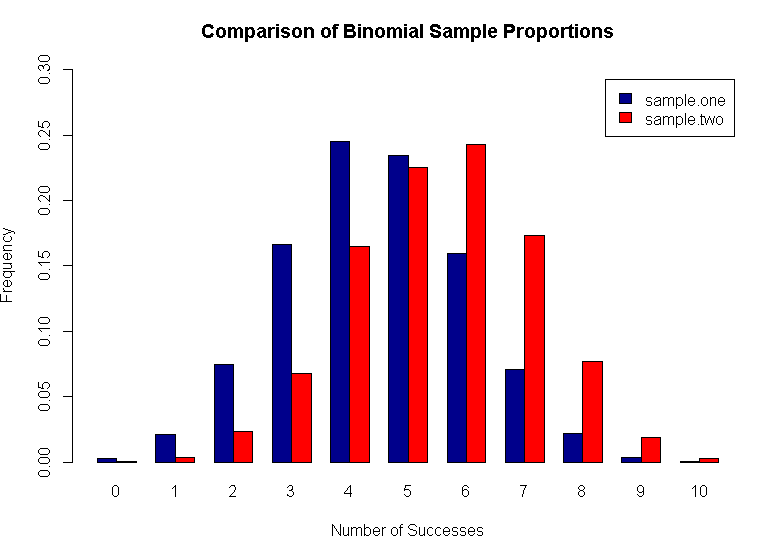
> counts <- rbind(sample.one, sample.two)

> barplot(counts, main="Comparison of Binomial Sample Proportions",

+ ylab = "Frequency", ylim = c(0,0.3),xlab="Number of Successes",

+ beside = TRUE, col=c("darkblue","red"),legend = rownames(counts),

+ names.arg = c("0","1","2","3","4","5","6","7","8","9","10"))



Test Questions (50 points total)

1. R has probability functions available for use (see Davies Chapter 16 and Kabacoff Section 5.2.3). Using one distribution to approximate another is not uncommon.

a) (3 points) Define the vector x <- c(0,1,2,3,4,5,6). Using this vector and the discrete probability functions shown below, calculate the probability of each outcome in x. Row bind the three sets of results, and generate a bar plot. The bar plot should have three color coded vertical bars for each outcome in x. Add additional features as you choose.

* + 1. dbinom(x, 100, 0.01)
    2. dpois(x, 1.0)
    3. dhyper(x, 20, 2000, 100)

b) (5 points) The normal distribution may be used to approximate the binomial   
distribution if np > 5 and np(1-p) > 5. Find the following binomial probabilities using   
dbinom() and pbinom() with a probability p = 0.5 and n = 100. Then estimate the same

probabilities using the normal approximation.

1. The probability of exactly 50 successes.
2. The probability of 42 or fewer successes.
3. The probability of 58 or more successes.

c) (2 points) Use the binomial probabilities from dbinom() with n = 100 and p = 0.01 to calculate the expected value and variance for this distribution. (To do this you will need to use integer values from 0 to 100 as binomial outcomes with their corresponding probabilities.) Calculate the same using the formulas np and np(1-p). Compare.

1. A recurring problem in statistics is the identification of outliers. This problem involves plotting data to display outliers, and then classifying them.

a) (3 points) Use rexp(n=100, rate =1) to general a random sample of 100 values from an exponential distribution. Draw the random sample after using set.seed(123). Do not change this number as your sample will then differ from the answer sheet. If you must draw another sample, start the process with set.seed(123). Present these data in a side-by-side display showing a boxplot and a normal QQ chart. Add features to the plot as you choose.

b) (2 points) Identify the values which are outliers and which are extreme outliers. (Hint - Using boxplot.stats() is a convenient way to do this.)

c) (5 points) Repeat the steps in (a) and (b) on a transformed variable. Use the random  
 sample generated in (a), but transform it to a new variable using the following Box-Cox

Transformation: y = 3\*((x^1/3) – 1). (Note, x^1/3 is the cube root of x.) Address all the   
 requests and questions asked in (a) and (b).

1. Performing hypothesis tests using random samples from two populations is fundamental to statistical inference. The first part of this problem deals with comparing two different diets based on sample means. The ChickWeight data available on R will be used. Execute the following code to prepare a data frame for analysis in this problem.

> data(ChickWeight)

> index <- (ChickWeight$Time == 21)&((ChickWeight$Diet == "1")|(ChickWeight$Diet == "3"))

> result <- subset(ChickWeight[index,], select = c(weight, Diet))

> result$Diet <- factor(result$Diet)

> str(result)

Classes ‘nfnGroupedData’, ‘nfGroupedData’, ‘groupedData’ and 'data.frame':26 obs. of 2 variables:

$ weight: num 205 215 202 157 223 157 305 98 124 175 ...

$ Diet : Factor w/ 2 levels "1","3": 1 1 1 1 1 1 1 1 1 1 ...

The file “result” will have chick weights for two diets identified as diet “1” and diet “3”. Using the file “result”, complete the following two items using R.

a) (4 points) Use the weight data for the two diets to test the null hypothesis of equal population weights for the two diets. Test at the 95% confidence level with a two-sided t-test. This can be done using t.test() in R. Assume equal variances. Report the results.

Working with paired data is another common statistical activity. The ChickWeight data will be used to illustrate how the weight gain from week 20 to 21 may be analyzed. Use the   
following code to prepare pre and post data from diet “3” for analysis.

> data(ChickWeight)

> index <- (ChickWeight$Diet == "3")

> pre <- subset(ChickWeight[index,], Time == 20, select = c(weight))$weight

> post <- subset(ChickWeight[index,], Time == 21, select = c(weight))$weight

b) (2 points) Plot the post data versus the pre data using a scatterplot. Compute the sample variances of pre, post and (post-pre). Compare the sample variances. What does this   
suggest regarding the advantages of performing paired t-tests?

c) (4 points) Construct a 95% confidence interval for the average weight gain from week  
20 to week 21. Do not use t.test(). Write the code for the test including determination of the endpoints of the confidence interval. Present the average difference and the confidence interval. What do you conclude about weight gain?

1. This problem deals with sampling distributions and the central limit theorem. Statistical inference depends on using a sampling distribution for a statistic in order to make confidence statements about unknown population parameters. The central limit theorem is a fundamental part of statistical inference. This problem illustrates how this comes about.

These data deal with the flow of the Nile from 1871 to 1970. Use the code below to prepare the data. This code may be helpful in crafting your solution.

> data(Nile)

> m <- mean(Nile)

> std <- sd(Nile)

> x <- seq(400, 1400,1)

> hist(Nile, freq = FALSE, col = "darkblue", xlab = "Flow",

+ main = "Histogram of Nile River Flows 1871 to 1970")

> curve(dnorm(x, mean=m, sd=std), col="orange", lwd=2, add=TRUE)

a) (3 points) Using the Nile River flow data, calculate skewness and kurtosis using the moments package. Using the Nile data, present a side-by-side display of a Quantile-Quantile plot (using qqnorm() and a boxplot (i.e. par(mfrow = c(1,2). Add features to these displays as you choose. Do these plots reveal any outliers?

b) (4 points) Using set.seed(124) and the data in Nile, generate 1000 random samples of size 25 with replacement. For each sample that is drawn, calculate and store the sample mean. This will require a “for” loop and use of the sample() function . Present a histogram of the sample means with the normal density function superimposed as shown above. Present the average of the sample means and the variance for the sample means shown in the histogram.

c) (3 points) Using set.seed(127) (a different starting point) and the Nile data, generate 1000 random samples of size 100 with replacement. For each sample, calculate and store the sample mean. Calculate the mean value and standard deviation of the 1000 generated means. Compare these summary statistics with the summary statistics calculated in b) above. Discuss the central limit theorem. Is the change in sample variances from b) to c) what would be expected?

1. This problem deals with 2x2 contingency table analysis. This is an example of categorical data analysis (see Kabacoff pages 145-151). This type of analysis is useful for determining if two categorical or nominal variables are associated. The method shown in this problem can be used to screen data for potential predictors that may be used in building a model.

The Seatbelts data set contains monthly road casualties in Great Britain 1969-84. A chi-square test of independence will be used (See Black Section 16.2) to determine if there is a trend in drivers killed over the period 1969-84. The first step is to organize the data for analysis. Use the code below to generate two variables: killed and month (starting in 1969).

a) (3 points) Using Seatbelts, generate a scatter plot of killed versus month. Show vertical and horizontal lines to indicate the median month and the median of killed. This is one way to evaluate variables pairwise for potential association. Label and add features as desired.

b) (2 points) Use table() to generate a 2x2 contingency table showing the fatality count classified by killed and month. Use the uncorrected chisq.test() to test at the 95% confidence level that killed and month are independent. State your conclusion.

c) (5 points) Write a function that computes the uncorrected Pearson’s Chi-squared statistic based on a 2x2 contingency table with margins added. (Check Davies Section 11.1.1 pages 216-219, and Kabacoff Section 20.1.3 pages 473-474)*.* Add margins to the contingency table from (b) and submit to the function you have written. Compare the result with (b). You should be able to duplicate the X-squared value and the p-value.  
  
An example of code to include in the function is shown below. These statements calculate the expected value for each cell in the table. Using these statements the Pearson Chi Square Statistic may be calculated within the function and the value returned.

# To be used with 2x2 contingency tables that have margins added.

e11 <- x[3,1]\*x[1,3]/x[3,3]

e12 <- x[3,2]\*x[1,3]/x[3,3]

e21 <- x[3,1]\*x[2,3]/x[3,3]

e22 <- x[3,2]\*x[2,3]/x[3,3]