Lab 3 - Normality - STA 570

# Project Goals

In this lab assignment, you will be asked to:

* Assess normality for a given data set
* Calculate exact binomial probabilities

# Assessing Normality

It is often necessary to determine whether sample data appear to be from a normally distributed population, and that determination is helped with the construction of a histogram and normal quantile plot. Refer to Data Set 1 “Body Data” in Appendix B. For ONLY the last four variables (columns), determine whether the data appear to be from a normally distributed population.

* Please type the remainder of the lab in a new RMarkdown file entitled Lab3\_lastname.

1. Copy the following code into an R chunk and run it. This code will first construct a histogram for each of the four variables and then constructs a quantile plot for each variable. **Based on both plots for each variable, list the variables that do not appear to be normally distributed.** Explain. Look for overall trends and do not look too closely at the details of the plots. Pay particular attention to the extreme values on each plot.

**Make sure you watch the video.**

#The library function loads or "wakes up" the packages.   
library(tidyr)  
library(ggplot2)  
  
#Next, import the data set. See video for details.  
  
  
data <- read.table(file = "bodydata.txt", sep = "\t", header = TRUE)  
  
#With the tidyr and ggplot2 packages, we can simultaneously look at several of the histograms and normal quantile plots. We will only look at three variables at a time. The code below is only for Height, Waist, Arm circumerference, and BMI. This code is a little complex. Make sure you copy it exactly as you see it.   
data[,12:15] %>% gather() %>% head()  
ggplot(gather(data[,12:15]), aes(value)) +   
 geom\_histogram(bins = 10) +   
 facet\_wrap(~key, scales = 'free\_x')

#Similarly, we can look at the normal quantile plots for each column. The quantile plots are also called QQ plots.   
ggplot(gather(data[,12:15]), aes(sample=value)) +   
 stat\_qq() +   
 facet\_wrap(~key, scales = 'free\_y')

# Binomial Probabilities

Section 6-6 described a method for using a normal distribution to approximate a binomial distribution. Many technologies are capable of generating probabilities for a binomial distribution. Instead of using a normal approximation to a binomial distribution, **use the following code to find the exact binomial probabilies in Exercises 9-12 of Section 6-6. You will then calculate the normal approximations and compare the values to determine if the normal approximation to the binomial distribution is actually a good approximation in these examples. Note: you will just look and compare, there is no particular criterion here. Just ask yourself “do these two probabilities seem ‘close’?”**

1. In a new R chunk, run the following code, state the probabilities, determine if the normal approximation is adequate.

#We can use the following information to approximate a normal distribution  
#mu = np, where n is the number of subjects, and p is probability  
#sigma = square root of npq, where q = 1-p  
  
#Problem 9: Find the probability that at least 40 of the 100 subjects have blue eyes.   
x = 40  
n = 100  
p = .35  
pbinom(x, size = n, prob = p, lower.tail = FALSE, log.p = FALSE)  
  
#The Normal approximation code is   
pnorm(x, mean = n\*p, sd = sqrt(n\*p\*(1-p)), lower.tail = FALSE)  
  
#Problem 10: Find the probability that at least 49 of the 100 subjects have blue eyes.   
x = 49  
n = 100  
p = .35  
pbinom(x, size = n, prob = p, lower.tail = FALSE, log.p = FALSE)  
  
#The Normal approximation code is   
pnorm(x, mean = n\*p, sd = sqrt(n\*p\*(1-p)), lower.tail = FALSE)  
  
#Problem 11: Find the probability that fewer than 5 of the 100 subjects have green eyes, with p = .12  
#We use x = 4 for binomial, since it is fewer than 5  
x = 4  
n = 100  
p = .12  
pbinom(x, size = n, prob = p, lower.tail = TRUE, log.p = FALSE)  
  
#The Normal approximation code is  
pnorm(x, mean = n\*p, sd = sqrt(n\*p\*(1-p)), lower.tail = TRUE)  
  
#Question 12: Find the probability that 33 or fewer of the 100 subjects have brown eyes, with p = .4  
x = 33  
n = 100  
p = .4  
pbinom(x, size = n, prob = p, lower.tail = TRUE, log.p = FALSE)  
  
#The Normal approximation code is   
pnorm(x, mean = n\*p, sd = sqrt(n\*p\*(1-p)), lower.tail = TRUE)