**Q1:**

mean=10.4

sd=2.4

norm\_17<-rnorm(n=17,mean,sd)

norm\_30<-rnorm(n=30,mean,sd)

norm\_300<-rnorm(n=300,mean,sd)

norm\_3000<-rnorm(n=3000,mean,sd)

**Q2:**

png(filename=here("eco\_634\_2021","lab\_04\_hist\_01.png"),width=1500,height=1600,units="p”",res=180)

par(mfrow=c(2,2))

hist(norm\_17,main="Hist.of 17 normally distributed random pnts.")

hist(norm\_30,main = "Hist. of 30 normally distr. random pnts.")

hist(norm\_300,main ="Hist. of 300 normally distr. random pnts.")

hist(norm\_3000,main ="Hist. of 3000 normally distr. random pnts.")

dev.off()

**Q3:** Upload figure

**Q4:** With increased sample size the distribution appears more normal. The histograms with smaller samples are more prone variation from the mean. This is representative in my histograms as the histogram with the smallest sample size (17) looks the least normally distributed (around the mean) than the histogram with a sample of 3000. The histogram with a sample size of 30 and 300 appear progressively more normal.

**Q5:** This occurs because histograms with larger sample sizes are less prone to variation from the mean. As sample size increases, the distribution appears more normal. More representatives are provide a better picture of the distribution.

**Q6:** The parameters and their values for the standard normal distribution are the mean and the standard deviation. The values for the histograms we just made were mean=10.4, and sd=2.4.

**Q7:**

x = seq(-20, 20, length.out = 1000)

y = dnorm(x,mean=10.4,sd=2.4)

svg(filename=here("eco\_634\_2021","norm\_1.svg"),width=7,height=9)

plot(x, y, type = "l", xlim = c(0, 20),main="normal distribution with a mean 10.4 and sd 2.4")

abline(h = 0)

dev.off()

**Q8:** Upload figure

**Q9:**

#create the dataset

set.seed(100)

n\_pts = 100

x\_min = -10

x\_max = 10

x = runif(n = n\_pts, min = x\_min, max = x\_max)

dat = data.frame(x = x, y\_observed = rpois(n\_pts,15))

#this is the code to make the figure

#create color

mycol <- rgb(100, 80,20 , max = 100, alpha = 50)

mycol2 <- rgb(20, 60, 20, max = 100, alpha = 80)

mycol3<- rgb(100,0,80,max=100,alpha=25)

png(filename=here("eco\_634\_2021","lab\_04\_q\_10.png"),width=1500,height=1600,units="px",res=180)

par(mfrow = c(2, 2))

#scatterplot

plot(dat,cex=0.8,col=mycol,pch=19)

plot(dat,cex=2,col=mycol2,pch=18)

#histogram

hist(dat$x,col=mycol3)

#boxplot

boxplot(dat)

dev.off()

**Q10:** Upload figure

**Q11:**

#create the dataset

set.seed(100)

n\_pts = 100

x\_min = -10

x\_max = 10

x = runif(n = n\_pts, min = x\_min, max = x\_max)

dat = data.frame(x = x, y\_observed = rpois(n\_pts,15))

#create the figure (with custom color)

mycol <- rgb(100, 80,20 , max = 100, alpha = 50)

png(filename=here("eco\_634\_2021","lab\_04\_q\_12.png"),width=1500,height=1600,units="px",res=180)

plot(dat,cex=0.8,col=mycol,pch=19)

guess\_x<-0

guess\_y<-15

guess\_slope<-0.2

x = runif(n = n, min = 1, max = 10)

#line\_point\_slope<-Vectorize(line\_point\_slope)

curve(line\_point\_slope(x,guess\_x,guess\_y,guess\_slope),add=T)

dev.off()

**Q12:** Upload Image

**Q13:**

#create the dataset

set.seed(100)

n\_pts = 100

x\_min = -10

x\_max = 10

x = runif(n = n\_pts, min = x\_min, max = x\_max)

dat = data.frame(x = x, y\_observed = rpois(n\_pts,15))

#column of predicted y-values

dat$y\_predicted<-line\_point\_slope(dat$x,guess\_x,guess\_y,guess\_slope)

#column of residuals

dat$resids<-dat$y\_predicted-dat$y\_observed

Q14: Upload two figures, one of the histogram one of the scatterplot.

#histogram of the models residuals

png(filename=here("eco\_634\_2021","lab\_04\_q\_14\_histogram.png"),width=1500,height=1600,units="px",res=180)

hist(dat$resids,main="Histogram of Model Residuals",xlab="Residual Values", ylab="Frequency")

dev.off()

#scatterplot of the models predicted values on x and residuals on y

png(filename=here("eco\_634\_2021","lab\_04\_q\_14\_scatterplot.png"),width=1500,height=1600,units="px",res=180)

plot(dat$y\_predicted,dat$resids,xlab="predicted y values",ylab="residual values",main="Residual Scatterplot")

dev.off()

