Brian Kang

HW Set #8

# hw\_lect13\_1

set.seed(123)

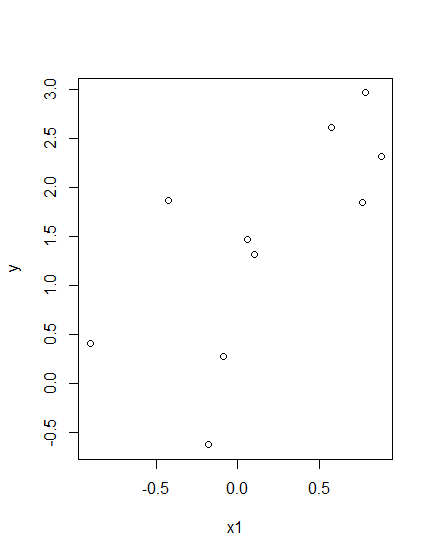
n = 10

x1 = runif(n,-1,1)

y = 1 + 2\*x1 + rnorm(n,0,1)

# a)

plot(x1,y)



# b)

lm.1 <- lm(y~x1)

print(summary(lm.1)$r.squared)

# 0.4827701

# c)

x2 = runif(n,-1,1)

x3 = runif(n,-1,1)

x4 = runif(n,-1,1)

x5 = runif(n,-1,1)

# d)

lm.2 <- lm(y~x1+x2+x3+x4+x5)

print(summary(lm.2)$r.squared)

# 0.6700543

# hw\_lect13\_2

x1 = c(8.9, 36.6, 36.8, 6.1, 6.9, 6.9, 7.3, 8.4, 6.5, 8.0, 4.5, 9.9, 2.9, 2.0)

x2 = c(31.5, 27.0, 25.9, 39.1, 39.2, 38.3, 33.9, 33.8, 27.9, 33.1, 26.3, 37.0, 34.6, 36.4)

y = c(14.7, 48.0, 25.6, 10.0, 16.0, 16.8, 20.7, 38.8, 16.9, 27.0, 16.0, 24.9, 7.3, 12.8)

# a)

x3 = x1^2

x4 = x2^2

x5 = x1\*x2

lm.3 <- lm(y~x1+x2+x3+x4+x5)

print(summary(lm.3)$coefficients[,1])

# b)

# x1: An increase in x1 (depth) by 1 is associated with

# an average decrease in y by 16.475

# x2: An increase in x2 (content) by 1 is associated with

# an average incease in y by 12.827

# x3: An increase in x3 (x1^2) by 1 is associated with

# an average incease in y by 0.096

# x4: An increase in x4 (x2^2) by 1 is associated with

# an average decrease in y by 0.243

# x5: An increase in x5 (x1\*x2) by 1 is associated with

# an average incease in y by 0.499

# c)

summary(lm.3)$r.squared

# 75.613% of the variation in y (strength) can be explained

# by the relationship with the multiple x values

# d)

summary(lm.3)$sigma

# The typical error in the prediction of y (strength)

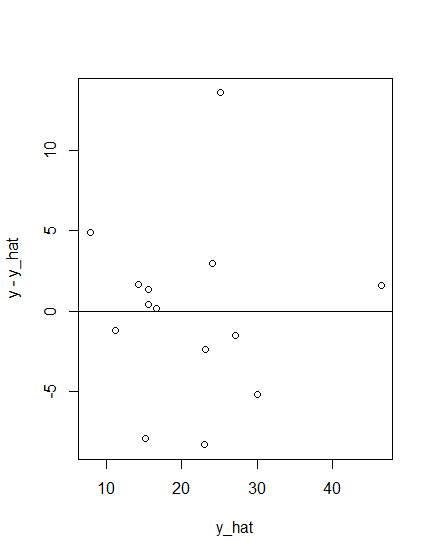
# about the fit is about 7.023

# e)

y\_hat = predict(lm.3)

plot(y\_hat, y-y\_hat)

abline(h=0)



# It is unsafe to conclude that the fit is appropriate

# for the data. At first glance the plots may seem random

# but if we disregard the one point that is very high up

# we can see that the plots are generally low valued.

# Also, the packed cluster near the zero line may show

# that this fit was unlikely able to capture the full

# extent of the compact data points.

# f)

lm.4 <- lm(y~x1+x2)

# g)

summary(lm.4)$r.squared

# 44.703% of the variation in y (strength) can be explained

# by the fit of its relationship with x1 (depth) and

# x2 (content).

# h)

# It is true that at least one of the terms in the equation

# contributed to the increase in R2. HOWEVER, we must

# also remember that by adding more fits and higher order

# terms, the R2 naturally increases (overfitting).

# i)

summary(lm.4)$sigma

# It is true that S\_e has decreased from 9.02 to 7.02

# by adding more higher order terms; however, considering

# how many extra terms we added, only a decrease by 2

# seems not very helpful, a sign for potential overfitting.

# hw\_lect13\_3

# a)

dat1 <- read.table("https://www.stat.washington.edu/marzban/390/summer18/hw\_3\_dat1.txt", header = T)

plot(dat1)

# From plotting the scatterplot of the x values v. y value

# both scatterplot shows a saddle nonlinearly curved plot,

# so we can suspect an interaction here.

# There also seems to be no collinearity since the

# the scatter plots of the x variables seems to be very

# random (they don't really interact).

t1 <- lm(dat1$y~dat1$x1+dat1$x2+dat1$x1:dat1$x2)

print(summary(t1)$r.squared)

print(summary(t1)$sigma)

# Just from adding the interaction, R2 increased a lot from

# 0.235 to 0.9235, and the S\_e decreases from 6.946

# to 2.017. adding additional terms did not improve as

# much as adding the interation term did.

# R2 = 0.9235, S\_e = 2.017

# b)

dat2 <- read.table("https://www.stat.washington.edu/marzban/390/summer18/hw\_3\_dat2.txt", header = T)

plot(dat2)

# From plotting the scatter plots, we see that both x vs y

# show a quadratic nonlinear curve. So a quadratic fit

# should be suitable.

# However, if we look at the scatter plot for x vs x

# we can see that there is a positive linear association,

# hence, it indicates that there is a collinearity.

# Because there is strong collinearity we will drop one

# of the predictors from the model (one of x1 or x2).

t2 <- lm(dat2$y~dat2$x1+I(dat2$x1^2)+I(dat2$x2^2))

print(summary(t2)$r.squared)

print(summary(t2)$sigma)

# Just from adding the first quadratic term R2 increased

# significantly from 0.247 to 0.8639 and S\_e decreasd fom

# 9.021 to 3.839, which is a great improvement.

# I dropped the first x2, but decided to include the

# I(x2^2) because it sufficiently increased R2, from

# 0.8639 to 0.9346, and decreased S\_e, from 3.839 to

# 2.664. Here, we are comparing a 86% vs 93% explanation

# for the fit and a decrease in error variance by 1/3.

# I believe that this is likely not overfitting because

# if we do consider the x2 v. x1, x2 also had a quadratic

# form, and so by doing this we are just taking care of

# of extra information that can be seen in a 3-D graph.

**hw\_lect13\_4 is at bottom**

# --------------------------------------------------------

# hw\_lect14\_1

# See hw\_lect13\_2

# hw\_lect14\_2

# a)

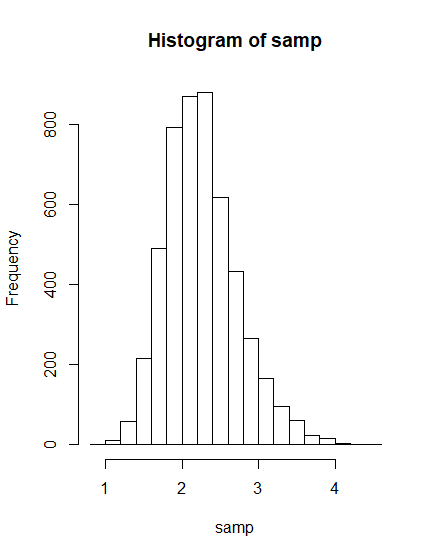
samp = c()

for (ii in 1:5000) {

samp = c(samp, max(rnorm(50,0,1)))

}

hist(samp)



# b)

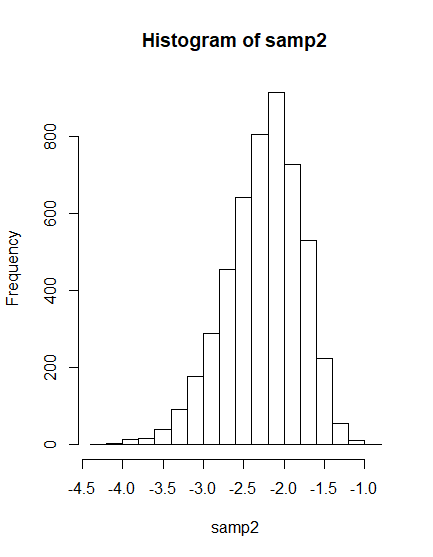
samp2 = c()

for (ii in 1:5000) {

samp2 = c(samp2, min(rnorm(50,0,1)))

}

hist(samp2)



# hw\_lect14\_3

# a)

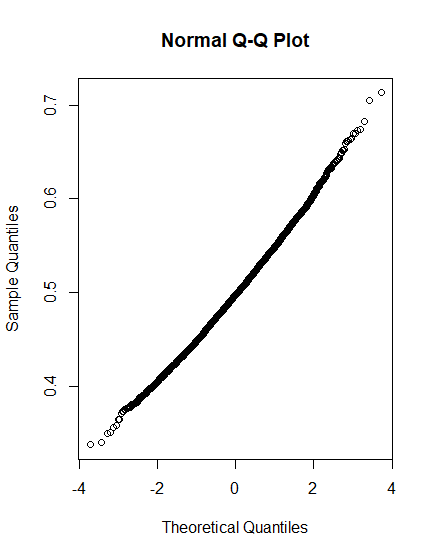
sampe = c()

for (ii in 1:5000) {

sampe = c(sampe, mean(rexp(100,2)))

}

qqnorm(sampe)



# b)

# I estimate that the mean is at 0 and the

# std dev is at 0.5.

# This estimated mean is constistent with the Mu from

# the Normal distribution (qqplot is straight so Normal

# and by default it follows std Normal dist, mu=0).

# However, the std dev estimate is not similar.

# I estimated 0.5, but the Omega/sqrt(n) =

# = (1/Lambda)/sqrt(n) = (1/2)/10 = 0.05, which is

# smaller than my graphical estimate.

**hw\_lect14\_4 & 5.47 is at bottom**

