Bolas\_WFC198\_mid-term

WFC 198 - Sampling animal populations

Lab midterm A01

Ellie Bolas

###### Instructions: Please complete this R script according to the

###### tasks below. At the end, “knit” your script into an MS Word

###### report. If you have partial code that won’t knit because it

###### is incomplete, please comment it out; we will look at that

###### partial code and give partial credit where appropriate.

###### Submit your “knitted” report on Canvas under the correct

###### assignment.

###### The lab mid-term has five main Tasks, followed by two bonus

###### Tasks. You do not need the Bonus Tasks to get full credit

###### but they can be used to make up for lost points in the main

###### part of the exam.

###### TOTAL POINTS, main questions: 23

###### TOTAL BONUS POINTS: 2

###### Task 1 - Read in and summarize data (7 Pts)

### On Canvas, under the Lab midterm assignment, you can download a .csv file called

### “Data.LabmidtermA01.csv”, with two variables measured for a set of sampling locations.

### Download the file (into the folder that is your working directory) and load it into R;

### then produce some data summaries. For b through e, make sure your knitted report

### contains the code and the numerical answers.

## a) Read in the data file (1 Pt)

getwd()

## [1] "/Users/clmuser/Desktop/Lab5"

dat <- read.csv("~/Desktop/Data.LabmidtermA01.csv") #read in data

## b) What are the column names of the data? (1 Pt)

head(dat)

## Quality Cover  
## 1 4.562253 0.36014205  
## 2 3.745070 0.42830228  
## 3 6.153045 0.58092673  
## 4 6.486104 0.02531582  
## 5 6.417473 0.74022618  
## 6 2.238260 0.90866932

names<-colnames(dat)  
names #names of column heads

## [1] "Quality" "Cover"

## c) What is the sample size (number of sampling locations)? (1 Pt)

numlocations <-nrow(dat)  
numlocations #number of rows/locations data was collected

## [1] 40

## d) How many mising data points (NA) are there? Note: if data is missing for one column,

## it is missing for all columns. (2 Pts)

missing <- sum(is.na(dat$Quality)) #can just use Quality column to look for NAs  
missing

## [1] 2

## e) For each of the two variables in the data file, calculate the mean with a single

## R command (2 Pts)

sum.both<-summary(dat)   
  
sum.both #gives the mean for each variable along with min, max, and median

## Quality Cover   
## Min. :1.754 Min. :0.01395   
## 1st Qu.:3.789 1st Qu.:0.19585   
## Median :4.857 Median :0.41695   
## Mean :4.794 Mean :0.41576   
## 3rd Qu.:6.108 3rd Qu.:0.55702   
## Max. :8.133 Max. :0.98402   
## NA's :2 NA's :2

#alternative option, get the mean for each variable on it's own  
avg.quality<- mean(dat$Quality, na.rm = TRUE)  
avg.quality

## [1] 4.793917

avg.cover <- mean(dat$Cover, na.rm = TRUE)  
avg.cover

## [1] 0.4157587

###### Task 2 - Analyzing data (3 Pts)

### Analyze the data you just read in with a linear regression model. Quality is the

### response variable, and Cover is the predictor variable. After running the model,

### produce a summary of the results using the appropriate R command.

### Tip: the function to run a linear regression is lm(). You only need to supply

### the “formula” and the “data” arguments.

#Quality = y, Cover = x  
lrmod <- lm(Quality~Cover, data = dat)  
lrmodresults<-summary(lrmod)  
lrmodresults

##   
## Call:  
## lm(formula = Quality ~ Cover, data = dat)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.9542 -1.0438 0.0403 1.3742 3.2311   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.9460 0.4681 10.567 1.4e-12 \*\*\*  
## Cover -0.3657 0.9421 -0.388 0.7   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.58 on 36 degrees of freedom  
## (2 observations deleted due to missingness)  
## Multiple R-squared: 0.004169, Adjusted R-squared: -0.02349   
## F-statistic: 0.1507 on 1 and 36 DF, p-value: 0.7001

###### Task 3 - Predictions (5 Pts)

## a) Create a vector of possible values for “Cover”, ranging from 0 to 1, at 0.02-unit

## intervals (2 Pts). Tip: Use the seq() function.

covnew <- seq(0,1,0.02)

## b) Calculate expected values for “Quality” for the predictor values created in Task 3a

## using parameter estimates from Task 2. DO NOT use the predict() function (3 Pts)

## Tip: In a linear regression, the expected value of Y is the value of Y we would see if

## there was no random variation: expected Y = beta0 + beta1\*X ## Note: IF (and only if!!) you were unable to fit the linear regression model under ## Task 2, perform these calculations with other numbers (but otherwise correctly) ## and you will get partial credit

coef(lrmod) #gives me the parameters estimated by the model

## (Intercept) Cover   
## 4.9459738 -0.3657324

exp.q <- 4.9459738 -.3657324\*covnew  
exp.q

## [1] 4.945974 4.938659 4.931345 4.924030 4.916715 4.909401 4.902086  
## [8] 4.894771 4.887457 4.880142 4.872827 4.865513 4.858198 4.850883  
## [15] 4.843569 4.836254 4.828939 4.821625 4.814310 4.806995 4.799681  
## [22] 4.792366 4.785052 4.777737 4.770422 4.763108 4.755793 4.748478  
## [29] 4.741164 4.733849 4.726534 4.719220 4.711905 4.704590 4.697276  
## [36] 4.689961 4.682646 4.675332 4.668017 4.660703 4.653388 4.646073  
## [43] 4.638759 4.631444 4.624129 4.616815 4.609500 4.602185 4.594871  
## [50] 4.587556 4.580241

###### Task 4 - Plotting (6 Pts)

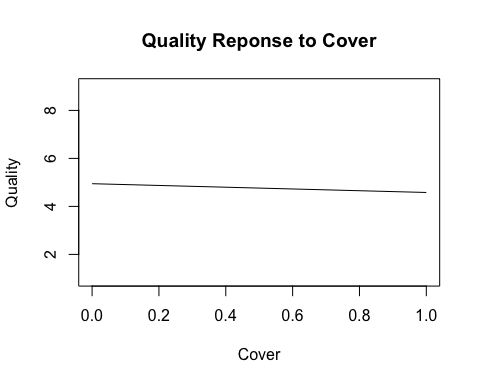
## a) Plot the expected values for “Quality” calculated under Task 3b against the possible

## values for “Cover” generated under Task 3a; the plot should display a line (not dots)

## and both axes should be properly labeled; set your y axis to range from 1 to 9 (4 Pts)

## Tip: You can set the range of you y axis using the “ylim” argument

plot(x=covnew, y=exp.q, type = "l", main = "Quality Reponse to Cover", ylab= "Quality", xlab="Cover", ylim = c(1,9))



## b) Add the original data (which you read in under Task 1) to this plot as points (2 Pts)

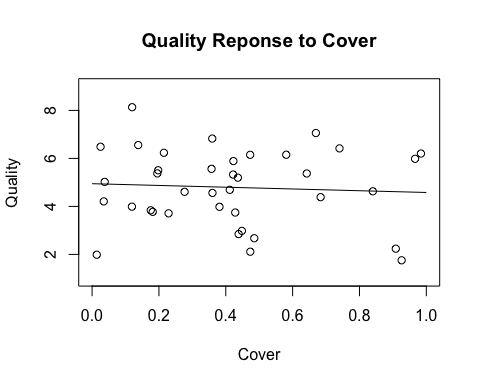
## Note: IF (and only if!) you are unable to produce the plot under 4a, make a new plot

## of the “Quality” column against the “Cover” column from the data read in under

## Task 1, with appropriately labeled axes, and set your y axis to range from 1 to 9

## for partial credit

plot(x=covnew, y=exp.q, type = "l", main = "Quality Reponse to Cover", ylab= "Quality", xlab="Cover", ylim = c(1,9))  
points(x = dat$Cover, y = dat$Quality)



###### Task 5 - Knitting

## Knit your script into an MS Word report (2 Pt)

## NOTE: If you want bonus points, knit AFTER answering the bonus tasks

###### BONUS TASKS - You do not need to complete these for full credit, but these can be used

###### to make up for lost points in the main questions

#### BONUS 1 - 1 Pt

## Create a vector that only holds the numerical (i.e., not NA) values of the “Quality”

## column from the data read in under Task 1

qvector <- dat[,"Quality"]  
qvector<- qvector[!is.na(qvector)]  
qvector

## [1] 4.562253 3.745070 6.153045 6.486104 6.417473 2.238260 2.847298  
## [8] 7.057585 1.754102 6.203228 4.208011 6.233765 4.690594 3.772540  
## [15] 1.986694 2.674007 4.629009 8.133295 5.506637 5.565117 5.022541  
## [22] 2.979788 5.888934 3.983728 3.839172 3.711061 5.983098 5.197261  
## [29] 5.371987 3.989229 4.384488 4.607967 2.112012 6.149815 6.826718  
## [36] 5.374674 6.555503 5.326797

#### BONUS 2 - 1 Pt

## Create a vector that holds the values of the “Cover” column from the data read in under

## Task 1 in ascending order

cvector <- dat[,"Cover"]  
cvector <- cvector[!is.na(cvector)]  
cvector <- sort(cvector, decreasing = FALSE)  
cvector

## [1] 0.01394515 0.02531582 0.03487068 0.03771285 0.11915845 0.11981790  
## [7] 0.13795291 0.17553216 0.18143922 0.19519925 0.19781111 0.21483806  
## [13] 0.22884530 0.27679021 0.35741289 0.35952577 0.36014205 0.38113406  
## [19] 0.41195458 0.42195475 0.42275947 0.42830228 0.43612783 0.43841056  
## [25] 0.44828734 0.47293607 0.47333755 0.48529254 0.58092673 0.64221628  
## [31] 0.66961646 0.68377413 0.74022618 0.84006959 0.90866932 0.92613875  
## [37] 0.96636794 0.98401852