#LOOK at your dataset by clicking on its name in the Environment window or by typing the following command:

View(COPD)

#LOOK at different variables in dataset:

colnames(COPD)

#HISTOGRAM TO CHECK DATA DISTRIBUTION: $ indicates that MWT1Best variable is found withhin the COPD dataframe

hist(COPD$MWT1Best)

###  ADJUST GRAPH IN SEVERAL WAYS  ###

#Change title: main = "new figiure title"  in function parenthesis

#xlab="New x axis label" or ylab = "New y axis label

#Change # of bins displayed using the command breaks = specifying # of bins you want dipslayed.

#CHANGE HISTOGRAM TITLE and the x axis label to only say "MWT1Best" instead of "COPD$MWTBest"

##BETTER RESULTION/DISTRIBUTION SYMMETRY via 12 bins instead of 6 (better resolution), the command to use is:

hist(COPD$MWT1Best, main = "Histogram of MWT1Best", xlab = "MWT1Best", breaks=12)

#CHECK OUTLIERS: high value above 650.  For this specific value, lets check it using subset(dataframe, variable > x)

subset(COPD, MWT1Best>650)

#Observation number 100 is ID 108 with corresponding values with MWT2 699 which is high but not impossible (CODE ABOVE)

#CHECK THE HIGH AND LOW VALUES SIMULTANEOUSLY:

subset(COPD, MWT1Best>600|MWT1Best<150)

#above code gives ID sample values for 58 and 108

#HISTOGRAM  FOR FEV1:

hist(COPD$FEV1, main="Histogram of FEV1",xlab="FEV1")

#SUMMARY STATS FOR DATASET:

summary(COPD) #GIVES summary of whole data set;

#see specific #'s: mean(), range(), IQR(), sd()

#na.rm = TRUE gives all NULL/missing values

#list() helps you looks at all statistics together

#SUMMARY STATS FOR MWT1BEST:

list("Summary" = summary(COPD$MWT1Best), "Mean" = mean(COPD$MWT1Best, na.rm=TRUE),

     "Standard Deviation" = sd(COPD$MWT1Best, na.rm=TRUE), "Range" = range(COPD$MWT1Best, na.rm=TRUE),

     "Inter-Quartile Range" = IQR(COPD$MWT1Best, na.rm=TRUE))

##SUMMARY STATS FOR FEV1:

list("Summary" = summary(COPD$FEV1), "Mean" = mean(COPD$FEV1, na.rm=TRUE), "Standard Deviation" = sd(COPD$FEV1, na.rm=TRUE), "Range" = range(COPD$FEV1, na.rm=TRUE),

     "Inter-Quartile Range" = IQR(COPD$FEV1, na.rm=TRUE))

#We can see that MWT1 has one patient with a missing value and all values complete for FEV1. The mean and median values are similar for FEV1 but not so MWT1Best

##so this suggests the distribution is a bit skewed.

#CALCULATE CORRELATION COEFFICIENT AND LOOK AT SCATTERPLOT: basic command is plot(x,y):

plot(COPD$FEV1, COPD$MWT1Best, xlab="FEV1", ylab = "MWT1Best")

#scatterplot shows an even distribution over both lung function and walking distance with

##no discontinuations in either, or obvious outliers.

###So, it is reasonable to calculate the Pearson’s correlation coefficient to assess a linear

####association between the two variables. But we will also have a look at the Spearman’s for comparison.

#####The basic command for a correlation test is cor.test(x,y)

#####where you can specify which method you want to use using the command method = "pearson" or "spearman".

####You need to remove missing values, otherwise you will have an error message.

###To do this, use the command use = “complete.obs

#PEARSONS vs SPEARMAN COEFFICIENT:

cor.test(COPD$FEV1, COPD$MWT1Best, use = "complete.obs", method="pearson")

cor.test(COPD$FEV1, COPD$MWT1Best, use = "complete.obs", method="spearman")

#Pearson’s correlation coefficient is 0.47 and the 95% confidence interval

##suggests the population coefficient is likely to lie somewhere between 0.3 to 0.6.

###The p value is less than 0.001 so there is very strong evidence against the null hypothesis of the coefficient being zero.

The Spearman’s correlation coefficient provides a similar result with estimated correlation of 0.45.

So, you have examined the raw data and found that it was fine, and that there was only one missing value.

You have discovered moderate correlation between walking distance and lung function.

#VIEW HISTOGRAM TO CHECK DATA DISTRIBUTION:

hist(COPD$MWT1Best)

hist(COPD$AGE)

# if you want histogram where figure title and the x axis label only say "Age" instead of "COPD$MWTBest"

##and also 12 bins instead of 10 (better resolution), the command to use is:

hist(COPD$AGE, main = "Histogram of AGE", xlab = "AGE", breaks=12)

#HISTOGRAM COMMAND ABOVE FOR AGE: shows slight left skew with peak around 75 y/o w/ 30 mbrs in this group.

##youngest group is 40 while oldest group is 90.  Median values centered b/t 60-80.

#CHECK # b/t 70-80 on HIST OF AGE:

subset(COPD,AGE < 80|AGE > 70)

#CHECK OVER 80:

subset(COPD, AGE > 85)

subset (COPD, AGE >90)

#FOR AGE CHECK SUMMARY STATS: using fxn () -->THIS GIVES ALL SUMMARY STATS summary(COPD$AGE)

mean(COPD, AGE)

mean(COPD$AGE)

sd(COPD$AGE)

range(COPD$AGE)

IQR(COPD$AGE)

na.rm = TRUE

summary(COPD$AGE)

summary(COPD$AGE, na.rm = TRUE)

#SCATTERPLOT AGE AND WALKING: #scatter plot shows slight negative correlation b/t walking distance and age

plot(COPD$AGE, COPD$MWT1Best, xlab = "AGE", ylab = "MWT1Best")

#CALCULATE PEARSONS AND SPEARMANS COEFFICIENT FOR AGE AND WALKING:

cor.test(COPD$MWT1Best, COPD$AGE, use = "complete.obs", method = "pearson")

cor.test(COPD$MWT1Best, COPD$AGE, use = "complete.obs", method = "spearman")

summary(COPD$AGE)

summary(COPD$MWT1Best)

#LINEAR REGRESSION FORMULA IN R:

##lm(outcome~predictor, data = dataframe)

#lm(MWT1Best~FEV1, data = COPD)

MWT1Best\_FEV1 <- lm(MWT1Best~FEV1, data = COPD)

summary(MWT1Best\_FEV1)

#ABOVE! Output gives residuals, coefficient est, std errors, p-val and regression equation coefficients

##The INTERCEPT indicates 'a', FEV1 indicates 'B' (linear effect of lung fxn)

###Can use above euqaiton to predict mean val of walk dist(MWT1Best) for fiven value of lung fxn(FEV1) using values from data set

#95% CONFIDENCE INTERVAL:

confint(MWT1Best\_FEV1)

confint(COPD$MWT1Best\_FEV1)

par(mfrow=c(2,2))

cor.test(COPD$MWT1Best, COPD$AGE, use = "complete.obs", method = "spearman")

rm(list=ls())

lm(MWT1Best~FEV1, data = COPD)

#CHECK MODEL ASSUMPTIONS GRAPHICALLY FOR LINEAR REGRESSION:

#1st plot is CONSTANT VARIANCE to cehck homegeneity of variance and linear relation(if you see no pattern in this graph, then asusmptions are met)

#2nd plot is Q-Q: checks that residuals follow a normal dist. Points should fall on line if normality assumption is met

#3rd plot allows to detect for heterogeneity of variance

#4th plot: detects points that have large impact on regression coefficients

par(mfrow=c(2,2)) #must be above plot command to view all at once

plot(MWT1Best\_FEV1)

par(mfrow=c(1,1))

COPD<-read.csv("COPD\_student\_dataset.csv")

#FIT REGRESSION MODEL WITH AGE AND WALKING DISTANCE:

##REMEMBER! walking distance is the outcome variable and age is the predictor variable.

COPD<-read.csv("COPD\_student\_dataset.csv")

MWT1Best\_AGE <- lm(MWT1Best~AGE, data=COPD)

summary(MWT1Best\_AGE)

confint(MWT1Best\_AGE)

###ABOVE! Model has been fitted: MWT1Best=a+B\*AGE

##WHERE a = 616.45 and B= 3.10

#R2 = 0.0529

###ASSUMPTIONS of Linear Regression are: there is linearity b/t outcome and predictor;

##the outcome variable is normally distributed across the predictor variable

#If above assumptions met, then residuals (dist. b/t observed values and fitted reg. line) follow

##a normal dist. w/ mean zero and constant variance across predictor values:

###Residuals ~ Normal (0, σ2)

#ABOVE can be assessed via several plots:

##QQ plot assesss normality:

predictedVals <- predict(MWT1Best\_AGE) #get predicted values for the model

residualVals <- residuals(MWT1Best\_AGE) # get residuals b/t model and data

par(mfrow=c(2,2)) # set plotting format

plot(MWT1Best\_AGE) # See residual plots, QQ plot

#ABOVE! commands produce multiple plots but only QQ plot is included for clarity

##QQ plot suggests some violoation of normality.  Plot shows values lying off straight line including middle section.

###Reclal that QQ plot is of quartiles of residuals against quartiles of theoretical normal distribution;

####if residuals are normal, then observations lie on straight line

#####ABOVE is backed up by viewing HISTOGRAM OF RESIDUALS:

hist(residualVals, main = "Histogram of residuals", xlab = "Residuals")

#main=command to change title of plot. xlab = modifies x-axis label.

##Only hist(residualVals) is necessary command

###residualVals is not a command, its a variable that was generated previously (see above command)

#HOW TO FIT A MULTIPLE REGRESSION MODEL IN R:to quantify r/s b/t FEV1 AND MWT1BEST for COPD pt's:

##FORMULA!  Y = α + β1\*X1 + β2\*X2 + ε

###   Y = outcome (dependent) variable

###   X1 = first predictor (independent) variable

###   X2 = second predictor (independent) variable

###   a = intercept (avg Y when X1 = X2 = 0).  Note: 'a' is unit specific

###   B1 = slope of line (change in Y for a 1 unit increase in X1 when X2 is constant). B1 is unit specific

###   B2 = slope of line (change in Y for a 1 unit increase in X2 when X1 is constant). B2 is unit specific

###   ε is the random variation in Y, i.e. the residuals

#CODE TO RUN MULTIPLE LINEAR REGRESSION: very similar to simple lin. reg except we're adding 2 predictor variables

##Model name <- lm(outcome ~ predictor1 + predictor2, data =dataframe)

MWT1Best\_FEV1\_AGE <- lm(MWT1Best~FEV1+AGE, data = COPD)

#WHERE:

##MWT1Best\_FEV1\_AGE is name of model

##MWT1Best is the outcome variable

##FEV1 is the first predictor variable

##AGE is the second predictor variable

#SUMMARY STATS FOR MWT1Best\_FEV1\_AGE:

summary(MWT1Best\_FEV1\_AGE)

confint(MWT1Best\_FEV1\_AGE)

#REMEMBER! the INTERCEPT indicates the regression constant 'a'

##FEV1 indicates linear effect of lung function

###B1 and AGE indicate linear effect of AGE, B2.

####R can run multi-linear regressions with >2 predictors and process is same as above

#MULTIPLE LINEAR REGRESSION W/ MWT1Best and FVC (2nd test of lung fxn)

lr1 <-lm(MWT1Best~FVC, data =COPD) #Run the regression assigning output to new variable lr

summary(lr1) #View the output of the regression

confint(lr1) #View 95% confidence intervals of the regression

#ABOVE model has been fitted MWT1best= α+β∗FVC

##where α = 254.95 and β=48.63 and adjusted R2 = 0.19

lr2 <-lm(MWT1Best~AGE, data = COPD)

summary(lr2)

confint(lr2)

#WHERE: α = 616.45 and β=−3.10, and adjusted R2 = 0.04

lr3 <- lm(MWT1Best~FVC+AGE, data = COPD)

summary(lr3)

confint(lr3)

#ABOVE Model has been fitted: MWT1best= α+β1∗FVC+ β2∗AGE

##where α = 425.38, β1=46.06 and β2 = -2.33.

###COMPARING the output of multiple regression w/ two models w/ single predictor variables

####we can see that both coefficients have been slightly reduced in the multi-reg. model.

#####FVC coefficeint has reduced from 48.6 to 46.1 and

#AGE coefficient has reduced from -3.1 to -2.3.

##p-value for FVC remains small (<.001), but:

###p-value for AGE has increased to .0059

####Remember! this change is b/c the coeff's are now adjusted coeff's:

##### 'a' is est. walking dist we'd expect for peeps age 0 y/o and w/ FV1 OF 0

#B1 is avg increase in walking distance for every 1 unit increase in FVC, keeping age held constant

##B2 is avg increase in walking distance for every one year increase age, keeping FVC constant

####Comparing adjusted R2 stats we see that mutlivariable model now explains 21% of variance in data

#####this is VERSUS 19% for model w/ FVC and 4% for model w/ AGE.

#Whilst the adjusted R2 is marginally higher in this model w/ both FVC and age

##compared w/ just FVC, we might question inclusion of AGE w/ p-value of 0.059 (which doesnt allow us to reject null hypothesis w/ coefficient at 9 and 5% sig. threshold)

###Later in course we'll cover model development and variable selection seeing that we shouldnt rely soley on significance testing.

#This model doesn't show any indication of colinearity (remember this is where there's strong lenear r/s b/t predictors that cause problems in

##estimation of model parameters)

###CHECK THIS W/ SCATTER PLOT AND CALCULATING CORRELEATION COEFFICIANT (pearson's or spearman's when only 2 predictors)

#however, not possiblewhen # of predictors increased beyond 2 (in which we'd need to

##produce a correlation matrix or calculate the VARIANCE INFLATION FACTOR -VIF)

#SCATTER PLOT:  AGE and FVC makes fairly random scatter of points:

plot(COPD$AGE, COPD$FVC, xlab ="AGE", ylab ="FVC")

#SPEARMAN'S coefficient:

cor.test(COPD$AGE, COPD$FVC, use="complete.obs", method="spearman")

##Spearman corr. coeff = -0.18 means only weak r/s b/t AGE and FVC (thus colinearity unlikley to be issue)

#RECALL THAT: 1.)there's linearity b/t the outcome and predictor variables.

##2.) outcome variable is normally distributed across the predictors

###3.) variance of outcome vriable is constant across the values of predictor variables

#Even w/ multiple predictors, if these assumptions are satisfied then residuals (distance

##b/t the observed values and the fitted regression line) follow a normal distribtuion w/ mean 0 and constant variance across the predictor values:

#Residuals ~ Normal (0, σ2)

##IMAGINE fitting this model: MWT1best= α+β1∗FVC+ β2∗AGE+ β3 ∗FEV1

###would this maybe not be a good idea?  FVC and FEV1 are both measures of lung function so see what we get?

#GETTING TO KNOW THE DATASET:

dim(COPD) #gives # of columns and rows

head(COPD) #this gives first few rows of each variables; good mix of continuous, binary, categorical

#INSPECT each variable, especially each one that you plan to use w/ analysis:

class(COPD$AGE)

#AGE is integer (technically a continous variable), but pt age is recorded to the year

##in the dataset.  R will treat variable as con't unless we tell it otherwise.

summary (COPD$AGE) #Get summary (min, max, IQR, etc)

##summary() also tells you if any missing vlaues, but none here

###but, mean and median values are closer to max value, which telsl you

###BUT, always graphically assess data b/f assuming about distribution

hist(COPD$AGE)

#Slight negative skew (tail at lower end longer than upper)

class(COPD$CAT)

summary(COPD$CAT)

hist(COPD$CAT)

#summary() and hist() showed a large outlier of 188 on CAT

## this is impossible d/t CAT range 0-40.

#INSPECT COPDSEVERITY:

class(COPD$COPDSEVERITY)

##COPD is a "factor" variable which is how R treats categorical data

#CHECK DISTRIBUTION OF ENTRIES:  use code table()

table(COPD$COPDSEVERITY, exclude =NULL)

#Including argument exclude =NULL to include missing values.

#BINARY VARIABLES are treated similar to categorical ones.

##R by default treats variables with numbers as integers, so good

###to check that a variable is meant to be a categorical one:

class(COPD$gender)

COPD$gender <- as.factor(COPD$gender)

class(COPD$gender)

#NOW, gender is registered as "categorical variable" and will be

##treated properly in models.  Lets check DISTRIBUTION of gender:

table(COPD$gender, exclude =NULL)

### ABOVE are the BASIC INSPECTIONS we must make before doing any analysis!

#HOW TO CHECK YOUR DATA IN R:

##see what other VARIABLE TYPES we have

###INTEGER: numeric value with no fraction

###CHARACTER: text/string variable, enclosed by apostrophes

###LOGICAL: objects (TRUE OR FALSE) that encode a logic

###FACTOR: categorical/nominal variables w/ levels of data

class(COPD$AGE) #integer

class(COPD$SGRQ) #numeric

class(COPD$COPDSEVERITY) #factor

#RUN THE REGRESSION: MWT1Best= α+ β∗copd

##MWT1Best is outcome variable, copd is a numeric predictor variable, w/ 4 categorical levels

####check if these variables are recorded correctly in R

class(COPD$MWT1Best)

class(COPD$copd)

#BOTH are saved as integers, but copd needs to be changed to "factor" variable

COPD$copd <- factor(COPD$copd) #converts copd to factor type

class(COPD$copd) #checks variable type, now is listed factor (was integer)

#VISUALIZE STRUCTURE OF DATA:

str(COPD$copd) #lists structure of variable within COPD dataset

#RUN A REGRESSION for 3 levels of output w/in copd variable

lr1 <- lm(MWT1Best~copd, data = COPD) #sets lr1 as vector for regression on MWT1Best= α+ β∗copd

summary(lr1) #views output of the regression

#IF YOU WANT TO CHANGE A VARIABLE FROM A TYPE

##as.numeric()

##as.character()

##as.integer()

#ONLY 3 COEFFICIENTS of levels d/t comparison w/ 4th level which is referred to as "Reference Group"

##Lets change the reference category of a categorical variable since when running regression on walking and COPD

###the linear reg. model automatically COPD level 1 (ie mild) as the reference category

####if we want to use "severe" group as ref. category for COPD severity (ie level 3):

COPD$copd <- relevel(COPD$copd, ref = 3)

lr1 <- lm(MWT1Best~copd, data = COPD) #sets lr1 as vector for regression on MWT1Best= α+ β∗copd

summary(lr1)

#CREATE NEW VARIABLES FROM OLD ONES:

##now want to create a new variable that indicates presences of <1 comorbidity

###or absence of comorbidities for variables:  Diabetes, muscular, hypertension, AtrialFib, and IHD

####call this variable 'comorbid'

#1.) check that all variables saved correctly as binary: 1 for positive, 0 for negative (thus we expect variables to be saved as factors)

##a) if they aren't, they need to be changed via factor()

#2.) create a vector for comorbid.

comorbid <-length(COPD$Diabetes) #this makes new vector 'comorbid' same length as diabetes variable (ie 101)

#3.) Now, assign each value 0 if no disease and 1 if <1 comorbidity

comorbid[COPD$Diabetes == 1 | COPD$muscular == 1 | COPD$hypertension == 1 | COPD$AtrialFib ==1 |

            COPD$IHD == 1] <-1

#ASSIGN a new value of 1 in comorbid vector if there's disease

comorbid[is.na(comorbid)] <- 0

#assigns a value of 0 in comorbid vector to the NA values in the vector

comorbid <- factor(comorbid)

#converts comobid to a factor vector

#ONCE THIS IS COMPLETE, you can review data several ways:

print(comorbid) # looks at data in comorbid vector

str(comorbid) #checks structure of data

comorbid[15] # looks at 15th value in the vector

COPD$comorbid <-comorbid #adds this new variable to existing COPD dataset so we can use following command

### BRANSON 8/23/2019

#1. Inspect the dataset for missing values and outliers:

##examine the datatype and distribution for all of these variables

###This can be done using the describe() function from the ‘Hmisc’ package.

#his function allows you to examine the different variables, providing the number of values, the range of the values, the number of missing values, the mean, and the different

##quartiles of values in our variables

describe(COPD)

#For categorical variables, you can tabulate the data using the CrossTable() function

##from the ‘gmodels’ package, then use the sum(is.na()) functions to check for missing values.

CrossTable(COPD$copd)

sum(is.na(COPD$copd))

#For continuous variables, such as MWT1Best, use

##the summary() command to check mean, median, minimum, maximum,

#1st and 3rd Quartiles, and # of missing values (NAs)

summary(COPD$MWT1Best)

#QUICK VISUAL CHECK for Continuous variabless, use hist()

##this helps check distribution/outliers

hist(COPD$AGE)

#FIND ANY EXTREME VALUES AND decide whether this value is impossible (possibly d/t error)

##in which case the value should be excluded (if just unusual value it should be left)

###If there's potential error, ask research to go back to orginal data and check prior to further analysis

#2)EXAMINE RELATIONSHIP B/T POSSIBLE PREDICTOR VALUES:

##After you are FAMILIAR W/ EACH VARIABLE, next step is to examine

###Relationship b/t predictors using corrleations and tabulations

#for CONTINUOUS VARIABLES, we use pairwise correlations and scatterplot matricies

##for CATEGORICAL VARIABLES, we use Cross Tabulations

#At start of this course, we learned correlation coefficient:

##cor.test().  To see PAIRWISE CORRELATION COEFFICIENTS only for continuous

###variables, we use command cor().

#the output of this command will be Pearson's corr. coef.

##If you want Spearmans, specificy w/ cor() method = 'spearman'

#et’s say you want Pearson’s pairwise correlation coefficient for the

##variables AGE, PackHistory, FEV1, FEV1PRED, FVC, CAT, HAD, and SGRQ

###Need to create CORRELEATION MATRIX to view diff. corr. coeffs:

#CORRELATION MATRIX ----

my\_data <- COPD[,c("AGE", "PackHistory", "FEV1", "FEV1PRED", "FVC", "CAT", "HAD", "SGRQ")]

#Create a new vector inclduing the variables to be analyzed

cor\_matrix <- cor(my\_data)  #Create a correleation matrix variable that are to be analyzed

#TO VIEW OUTPUT of cor\_matrix

cor\_matrix

round(cor\_matrix,2) #Round the values of the correlation to 2 dec. pts

#Here's easy way to visualize correlation using pairs() function:

pairs(~AGE+PackHistory+FEV1+FEV1PRED+FVC+CAT+HAD+SGRQ, data = COPD)

#To examine associations b/t CATEGORICAL VARIABLES, use cross tabulations:

##CrossTable(mydata$myrowvar, mydata$mycolvar)

###doe this for IHD and HTN variables

CrossTable(COPD$hypertension, COPD$IHD)

#This cross tabulation shows you that 8 patients had IHD only,

##and 11 patients had hypertension only,

###ne person had both hypertension and IHD and 81 had neither

####So, you would not be concerned

#if you wanted to include both of these in the model.

#3. Fit a simple linear regression model

##t’s useful to assess the relationship for each

###of variable in turn with the outcome

#fIT A REGRESSION MODEL WITH JUST A SINGLE PREDICTOR:

##this gives you chance to spot any unusual data that may be

###d/t errors in data or coding of variable.  Also helps

#to anticipate what might happen if fitting multivariable model

#Recall that fitting a linear regression model uses the lm() function in the following format:

##model\_name <- lm(outcome ~ predictor, data =dataframe)

##summary(model\_name)

##confint(model\_name)

##Summary

#To summarise, it’s important that you get to know your data really

#well before you start modelling.

#The first thing is examining variable distributions using summary statistics, tabulations and graphs.

#The next stage is examining the relationship between candidate predictors using cross tabulations and correlations.

#nd finally, get a feel for the relationships

##between each of the candidate predictors and the outcome by fitting a regression

#model for each variable in turn.

#This strategy will generate a fair amount of output for you to look through. My approach would be:

##1.) to set up the statistical code up in advance

##2.) run all the analyses

##3.) then grab a coffee - take some time to look over and absorb the results

#REMINDER: Good Practice Steps to better understand your dataset

##check each variable:

install.packages("Hmisc") #Downloads Hmisc package

library(Hmisc) #Loads the library

describe(COPD) #look @ variables in dataset to spot any outliers for further inspection

#CATEGORICAL VARIABLES: copd, gender, smoking, comorbid

##CHECK whther this are listed as CATEGORICAL; if not, recode them as "factor variables"

class(COPD$gender) #shows as factor, if it didnt, type:

##COPD$gender <- factor (COPD$gender)

#CHECK CATEGORICAL VARIABLES: describe()

describe(COPD$gender)

##CHECK CATEGORICAL VARIABLES using CrossTable() and sum() functions from 'gmodels' package

CrossTable(COPD$gender)

sum(is.na(COPD$gender))

#CONTINUOUS VARIABLES: AGE, PackHistory, CAT, FEV1, FEV1PRED, FVCPRED, HAD, SGRQ

##check summary stats and histogram

summary(COPD$AGE)

#HISTOGRAMS ----

hist(COPD$AGE, main = "Histogram of Age", xlab = "AGE")

##TITLE of Hist: 'main = '; x axis label: 'xlab = '

hist(COPD$PackHistory)

hist(COPD$CAT)

#LOOK FOR OUTLIERS and errors; see value 188 @ CAT.  This is

##incorrect since test range is 0-40; this value should be exluded

COPD$CAT[COPD$CAT > 40] <- NA

describe(COPD$CAT) #NOTICE there is now one missing value

##and the range is from 3 - 32

#EXAMINE RELATIONSHIP b/t PREDICTOR VARIABLES using CROSS TABULATIONS for

##CATEGORICAL VARIABLES and pairwise corrleations;

###also make SCATTERPLOT MATRICES for CONTINUOUS VARIABLES

#TO CREATE CROSS TABULATION: install 'gmodels' via:

##install.packages("gmodels")

library(gmodels)

CrossTable(COPD$gender, COPD$IHD)

#examine associations between categorical variables

##doing this for categorical "predictor" variables shows there

###aren't any obvious associations b/t candidate categorical predictor variables

#FOR CONTINUOUS VARIABLES: check pairwise corrleations and scatterplots

##CORRELATION MATRIX ---

my\_data <- COPD[,c("AGE", "PackHistory", "CAT", "FEV1", "FEV1PRED", "FVC", "FVCPRED", "HAD", "SGRQ")]

  #Creates new vector including the variables to be analyzed

cor.test <- cor(my\_data, method = "spearman")

  #creates correlation matrix of variables to analyze

cor\_matrix

  #views correlation matrix

round(cor\_matrix, 4)

  #round the values of corr. matrices to 4 decimal places

#SCATTERPLOT MATRIX:

pairs(~AGE+PackHistory+FEV1+FEV1PRED+FVC+CAT+HAD+SGRQ, data = COPD)

#ABOVE shows 4 measures of lung function highly correlated, which

##raises concern if we should include all variables in same model.

###If NOT, which ones should we include in final model and why?

#Think about possible associations b/t continuous predictor variables and

##categorical predictor variables?  How might we explore this?

#3) FINAL STEP: Fit simple LINEAR REGRESSION models b/t outcome of each candidate predictor variables:

lr1 <- lm(MWT1Best ~ gender, data = COPD)

summary(lr1)

confint(lr1)

lr2 <- lm(MWT1Best ~FEV1, data = COPD)

summary(lr2)

confint(lr2)

#Model w/ FEV1 exploains most variance in data b/c it had the largest adjusted R2 stat.

  #this seems like a reasonable justification to use FEV1 over other 3 variables

#COPD severity and presence of comorbidities were the only categorical predictor variables that

  #were statistically significant associated w/ walking distance

#Increasing COPDS severity was r/t decreasing walking distance; as was preseence of comorbity

#For continuous variables, AGE, PackHistory, COPD assessment test (CAT), FEV1, FEV1PRED, FVC,

  #FVCPRED, HAD, SGRQ were all statistically r/t walking disance (MWT1Best)

#The direction of these associaitons were also as expected.

  #Ex: increased age was r/t decreased walking distance, but increased lung function was r/t increased walking distance

#E-Activity: Multivariable Regression Model in R:

mlr1 <- lm(MWT1Best~FEV1+AGE+factor(gender)+factor(COPDSEVERITY)+factor(comorbid), data = COPD)

summary(mlr1)

confint(mlr1)

#NOTE: you can add function factor() directly in your regression to be quicker.  But,

  #this doesn't change how the variable is saved, ex: 'gender is INCORRECTLY registered as integer variable

##so by adding factor() function in regression, its treated as factor for that regression, but

  #variable iteself still recorded as an integer variable.  Refer to the earlier

  ##reading to see how to change variable data types

#in this MULTIVARIABLE MODEL the coefficients are similar to coefficients est. in

  #simple models (except COPD); However, AGE and presenece of <1 comorbid are now the

  ##only significant variables in the outcome

  ###The adjusted R2 shows you that the model explains 29% of the variance in the outcome

#For AGE the model est. that avg walking distance decreases by 3.2 mtrs for e/ year increase in age;

  #after adjusting for FEV1, gender, COPD severity and presence of <1 comorbid.

#the COPDSEVERITY coefficients are much reduced in multivariable model compared to simple models

  #but only the very severe estimate is statisticlaly significant.

#Examination of confidence interval for this coefficient shows that it ranges from

  #-254.9 to -14., not very precise.  This could be expalined by very small numbers

  ##in this category and something we need to WATCH OUT for when fitting multivariable regression models

#FEV1 and "moderate" and 'severe' COPD are no longer significant in multivariable model

  #the lack of change for most coefficients b/t simple models and multivariable should reassure you that

  ##colinearity is unlikely a problem in the model.  This can be explored by examining VIF

#DOWNLOAD 'mctest' to use fxn imcdiagF()

imcdiag(model.matrix(mlr1)[,-1], mlr1$model[1], method= 'VIF')

  #this suggests potential problems with FEV1 and COPDSEVERITY; in fact, it doesn't seem unreasonable that FEV1 (a measure of lung capacity)

    #and COPDSEVERITY rating might be associated.  Explore what impact each of these variables is having on your final model b/f moving on

#RUNNING A MULTIPLE REGRESSION:

  #Regression models don't always assume predictor effects are additive; predictors can influence interactions between other predictors

#BINARY PREDICTORS: check interaction b/t Afib and Diabetes:

  #MWT1best= α+ β1∗Diabetic+ β2∗AtrialFib+ β3∗Diabetic∗AtrialFib

    ##WHERE: Diabetic= 0 if No, 1 if Yes.  Afib = 0 if no, 1 if yes

      ###Diabetic\*AtrialFib = 0 if Neither, 1 if Both

#To Fit this Model, need to make Diabetic\*AtrialFib variable:

  #NOTE! R can't do this if variables saved as factors.  They need to remain/be changed to integers using 'as.integer() functino:

COPD$Diabetes <- c(0,1)[as.integer(COPD$Diabetes)] #This didn't work (see other code below that works)

COPD$AtrialFib <- c(0,1)[as.integer(COPD$DAtrialFib)]

COPD$Diabetes <- integer(COPD$Diabetes)

class(COPD$Diabetes)

COPD$AtrialFib <- integer(COPD$AtrialFib)

class(COPD$AtrialFib)

#now, CREATE NEW VARIABLE w/:

DAF <- COPD$Diabetes\*COPD$AtrialFib

  #Then code for regression:

r1 <- lm(MWT1Best~factor(Diabetes)+factor(AtrialFib)+factor(DAF), data=COPD)

summary(r1)

confint(r1)

r2 <- lm(MWT1Best~factor(Diabetes)

         +factor(AtrialFib)+factor(Diabetes\*AtrialFib), data=COPD)

#REMEMBER: Diabetes and AtrialFib must be saved as integers, otherwise it won't work

#TRY OTHER PREDICTORS FOR MULTIREGRESSION: perhaps smoking, gender, or IHD and Diabetes

COPD$smoking <- c(0,1)[as.integer(COPD$smoking)]

COPD$gender <- c(0,1)[as.integer(COPD$gender)]

COPD$smoking <- integer(COPD$smoking)

class(COPD$smoking)

COPD$gender <- integer(COPD$gender)

class(COPD$gender) #saved as numeric??? not integer (see above what it was saved originally as)

#now, CREATE NEW VARIABLE w/:

DAF <- COPD$smoking\*COPD$gender

#Then code for regression:

r3 <- lm(MWT1Best~factor(smoking)+factor(gender)+factor(DAF), data=COPD)

summary(r3)

confint(r3)

Call:

  lm(formula = MWT1Best ~ factor(Diabetes) + factor(AtrialFib) +

       factor(DAF), data = COPD)

Residuals:

  Min      1Q  Median      3Q     Max

-218.15  -51.88   18.70   51.85  270.86

Coefficients:

  Estimate Std. Error t value Pr(>|t|)

(Intercept)          428.14      10.39  41.200  < 2e-16 \*\*\*

  factor(Diabetes)1     -7.69      28.02  -0.274  0.78436

factor(AtrialFib)1   -72.05      29.21  -2.467  0.01541 \*

  factor(DAF)1        -130.11      47.70  -2.727  0.00759 \*\*

  ---

  Signif. codes:  0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 86.32 on 96 degrees of freedom

(1 observation deleted due to missingness)

Multiple R-squared:  0.3635,       Adjusted R-squared:  0.3437

F-statistic: 18.28 on 3 and 96 DF,  p-value: 1.841e-0

#which equates to:

#MWT1best= 428.1−7.7∗Diabetic−72.0∗AtrialFib−130.1∗(Diabetic∗AtrialFib)1

  #This can be interpreted as follows:

      #A person with diabetes and no atrial fibrillation has an estimated average walking distance of 420.4 metres.

#MWT1best= 428.1−7.7∗Diabetic

    #=428.1 − 7.7

    #=420.4

      #A person with atrial fibrillation and no diabetes has an estimated average walking distance of 356.1 metres.

#MWT1best= 428.1−7.7∗Diabetic−72.0∗AtrialFib−130.1∗(Diabetic∗AtrialFib)1  =428.1−7.7−72.0−130.1

  #=218.3

    #A person with both diabetes and atrial fibrillation has an estimated average walking distance of 218.3 metres.

#The above output can be obtained via code 'predition()' - but you must DOWNLOAD the 'prediction' package.

  #NOTE: below is 'list() thats not necessarily necessary.  But, it allows you to view all 3 prediciton() commands in one output

    #It's also NECESSARY within the 'prediction()' command to view variable outputs:

r2 <- lm(MWT1Best~factor(Diabetes)

         +factor(AtrialFib)+factor(Diabetes\*AtrialFib), data=COPD)

install.packages("prediction")

library(prediction)

list("Diabetes" = prediction(r2, at = list(Diabetes = c(0,1))), "AtrialFib" = prediction(r2, at = list(AtrialFib = c(0,1))),

        "Diabetes\*AtrialFib" = prediction(r2, at = list(Diabetes = c(0,1), AtrialFib = c(0,1))))

#EXPLORE OTHER INTERACTIONS: look at interaction b/t IHD and Diabetes

  #FIRST fit model w/out interaction effect: MWT1best= α+ β1∗Diabetic+ β2∗IHD

r4 <- lm(MWT1Best~factor(Diabetes)+factor(IHD), data = COPD)

summary(r4)

confint(r4)

#We might suspect that predicative  effects of diabetes and IHD depend on eachother.  This can be explored via:

  #MWT1best= α+ β1∗Diabetic+ β2∗IHD+ β3∗Diabetic∗IHD

r5 <-lm(MWT1Best~factor(Diabetes)+factor(IHD)+factor(Diabetes\*IHD), data=COPD)

summary(r5)

confint(r5)

###The effect of the interaction term in this model has been far less dramatic on the regression coefficients and,

  #in fact, there is no evidence of an interaction effect between IHD and diabetes.

#INTERPRETING BINARY AND CATEGORICAL VARIABLES IN MULTIVARIABLE LINEAR REGRESSION MODEL:

  #Say we're interested in effect of AGE, FEV1, gender, smoking, and MWT1Best in COPD patients

mlr1 <- lm(MWT1Best~AGE+FEV1+factor(gender)+factor(smoking), data = COPD)

summary(mlr1)

confint(mlr1)

#When INTEPRETING THIS OUTPUT, important to know how BINARY VARIABLES and CATEGORICAL VARIABLES have been coded

  #CHECK THIS FOR SMOKING: 1 if yes, 2 if ex-smoker

  # '' '' GENDER: 0 if Female, 1 if Male

    ##Also need to know what reference group is for each variable; in current model, reference category for gender is 'female' and for smoking is 'current'

        ###In R, the default refrence group is one with smallest code value

  #HOW does thsi effect interpetation of R output? For the coefficient of gender, a one unit increase represents a change from female to male;

    ##for smoking, a 1 unit increase changes from 'current smoker' to 'ex-smoker'

#If we're interested in change from being 'exsmoker' to 'current smoker' we need to change reference category: COPD$smoking <- relevel(COPD$smoking, ref = 2)

#Note: don’t forget that the variable smoking needs to be saved as a factor for the function relevel() to work!

COPD$smoking <- factor(COPD$smoking)

class(COPD$smoking)

COPD$smoking <- relevel(COPD$smoking, ref = 2)

  #NOW, run a NEW OUTPUT for above analysis:

mlr1 <- lm(MWT1Best~AGE+FEV1+factor(gender)+factor(smoking), data = COPD)

summary(mlr1)

confint(mlr1)

#CHECK VISUALLY TO HELP INTERPRETATION OF #'s:

plot(MWT1Best~AGE+FEV1+factor(gender)+factor(smoking), data = COPD)

#For the above, recall that changing the reference group of binary variable doesn't change the regression coefficient, it only changes the 'sign' of regression coefficient

  #SEE for yourself comparing two different regression coefficeints

#We can change reference group for categorical predictor variables.  If model has a categorical variable, it's important to cehck default group b/f interpreting results.

  #The model may not be what you want; We learnt thatpredictors may not be independtly additive to trend and may depend on another variable.

  ##If this is case, the variables are INTERACTING.

  #We've been through examples of interactions b/t TWO variables AND binary and continous variables ...

    #You can also include interactions b/t more than than 2 variables, but INTERPRETATION CAN BE TRICKY, think carefully about fitting such models

###Remember, BEFORE INCLUDING ANY INTERACTIONS, check the CODING OF BINARY VARIALBES to make sure the GROUPS ARE CODED AS '0' & '1'.

  #Ex: For smoking status, we noticed that smoking is in fact coded as 1 for ex-smokers and 2 for current smokers.  If you leave coding like this you will need to be

  ##careful when interpreting results; it EASES INTERPRETATION OF INTERACTIONS if you perform a RECODE so that current smokers are '0' and ex-smokers are '1'.

    #to do this, we must first save smoking as "integer'

COPD$smoking <- integer(COPD$smoking)

class(COPD$smoking)

#Now, we can code smokers as 0 and ex as 1

COPD$smoking[COPD$smoking == 2] <- 0

#Say we believe smoking impacts lung fxn which impacts walking distance.  Explore interactions b/t FEV1 and smoking:

mlr2 <- lm(MWT1Best~AGE+factor(gender)+(FEV1\*factor(smoking)), data = COPD)

summary(mlr2)

confint(mlr2)

plot(MWT1Best~AGE+factor(gender)+(FEV1\*factor(smoking)), data = COPD)

#REMEMBER! this interaction completely cahnges your interpretation of regression coeffecients; plots can help with this, especially with binary and continous interactions

  #However, in this example, NO evidence to support the hypothesis of interaction b/t FEV1 and smoking since p-value = 0.686.  DONT PROCEED WITH THE MODEL

#review all the above as we now know how to fit, assess, and interpret a MULTIVARIABLE LINEAR REGRESSION MODEL w/ a variety of different predictor variables.

  #next, lets learn good steps to build such models

#NOW, BUILD YOUR OWN MULTIVARIATE MODEL: see spiral notes '8/31 to 9/1'

  #POPULATION: COPD patients

  #OUTCOME: depression/anxiety (HAD)

  #MODEL PURPOSE: to discover any possible relationship b/t COPD predictors and anxiety/depression

  #CANDIDATE VARIABLES: all comorbidities, Age, FEV1, SGRQ, gender, smoking status, MWT1best

  #CHOSEN VARIABLES: comorbidities, FEV1, and MWT1Best

#STEPS TO DEVELOP MULTIVARIATE LINEAR REGRESSION MODEL:

  #1.) ID known predictors and their interaction; ok if not @ 5% CI

  #2.) EXAMINE the data

      ##Don't include poor quality/missing data.

        ###NARROW DISTRIBUTIONS: categorical data with very small #'s: rather than excluding, consider

#grouping through a RECODE: IE possibly merge "severe" w/ "very severe" in COPDSEVERITY group

        ###COLINEARITY: check this thru CORRLEATION MATRIX or VARIANCE INFLATION FACTOR;

#if several varialbes highly inflated, consider which to exclude

  #3.) COMBINE SEVERAL VARIABLES INTO ONE: eg if want to examine more than 1 comorbid

  #4.) PRE-SPECIFY interactions of interest; per Khan academy notes state Alpha and CI levels ahead of time

  #5.) DATA REDUCTION METHOD: if too many predictor variables, conside which to exclude

      ##TOO MANY COEFFICIENTS overfits model and explains random error rather than real relationships

#EXAMPLE from Forum: Eglal Elrayah Mohamed

#My question was on factors predictive of best walking distance in COPD patients. So I built a model using the following variable:

#>Mine<-lm(COPD$MWT1Best~COPD$AGE+COPD$gender+COPD$smoking+COPD$FEV1+COPD$HAD+COPD$comorbid,data=COPD)T

#The model explained around 31%

#Then I replaced the smoking status with smoking packs , and I think this is a better model :

#Mine\_2<-lm(COPD$MWT1Best~COPD$AGE+COPD$gender+COPD$PackHistory+COPD$FEV1+COPD$HAD+COPD$comorbid,data=COPD)

#The model explained around 34% of the variability

#FINAL QUIZ: Imagine you try to fit the following MULTIVARIABLE LINEAR REGRESSION:

mlr5 <-lm(HAD~AGE+CAT+gender+comorbid, data = COPD)

summary(mlr5)

confint(mlr5)

plot(HAD~AGE+CAT+gender+comorbid, data = COPD)

hist(HAD~AGE+CAT+gender+comorbid, data = COPD)

hist(COPD$HAD)

hist(COPD$FEV1)

#LINEAR REGRESSION IN R: FINAL CODE

#NEW R SCRIPT:

  #Remove previous variables: rm(list=ls())

  #Set the working directory: setwd("file\_pathway")

  #Load a dataset, labeling it 'COPD': COPD <- read.csv("COPD\_student\_dataset.csv")

#INSTALL NEW R PACKAGE:

  #install.packages("Package name")

  #library(Package name)

#VIEWING THE DATASET:

  #Look at the whole dataset: View(COPD)

  #Print first few rows of your dataset: head(COPD)

  #See number of rows/columns in dataset: dim(COPD)

  #Look at all values in a variable: print(variable)

  #To visualize the structure of data in a variable: str(variable)

  #Look at specific value (x) in a variable: variable[x]

#FOR CONTINUOUS VARIABLES:

  #View # of values, missing values, mean and ranges using the describe() function from 'Hmisc' package

#FOR CATEGORICAL VARIABLES:

  #View # of values, missing values, mean, and ranges using the describe() function from 'Hmisc" package OR tabulate the data to view the # of values and their frequency using CrossTable()

    ##function from 'gmodelsw' package.  To view missing values, type sum(is.na(variable))

  #Viewing the categories and distribution of entries in a categorical variable: table(catvariable)

  #Can add argument exclude = NULL in the function parenthesis to include missing values in the output

#RUNNING A LINEAR REGRESSION:

  #Basic format is:

    ##modelname <- lm(outcome~predictor, data = dataframe)

    #Viewing the regression model output: summary(modelname)

    #Viewing the model 95% confidence intervals: confint (modelname)

#DRAWING Q-Q PLOT, CONSTANT VARIANCE PLOT, AND OTHER DIAGNOSTIC PLOTS:

  #Calc predicted values: predict(modelname)

  #Calc residuals: residuals(modelname)

  #Set a plotting format of 4 graphs: par(mfrow=c(2,2))

  #View the 4 resulting plots: plot(modelname)

#CREATE A HISTOGRAM:

  #Basic format is: hist(variablename)

  #if getting variable from the dataset, the '$' sign allows R to locate the variable (Eg COPD$MWT1Best)

  #to change the title of histogram, use the command: main = "histogram title"

  #Don't forget quotation marks when using text!

  #To change # of bins displayed, use the command 'main =' to specify # of you want to see

  #To see specific valaues in your variable, use 'subset() function, w/ basic code subset(dataframe, variable > 15)

    ##if you want to see values over 15 for that variable. You can add additive rules by including ‘|’, e.g. subset(dataframe, variable1 > 15 | variable2 < 5)

#SUMMARY STATS:

  #Basic summary stats (min, med, max, 1st and 3rd Qu, # of blank cells):

    ##summary(variablename)

  #List of summary stats, including basic summary() outcome, standard deviation, range, and IQR:

    ##list(summary(variablename)), sd(variablename, na.rm = TRUE), range(variablename, na.rm = TRUE), IQR(variablename, na.rm = TRUE))

  #Note that na.rm = TRUE command tells R to remove NA values.  Without this, the error message will be displayed.

#CORRELATION:

  #Scatterplot of two variables: plot(x,y)

  #Correlation coefficient: cor(x,y)

  #The default method is Pearson, but it can be changed to Spearman by adding method = "spearman" in

    ##parenthesis.  Need to remove missing values otherwise error message pops up-

      ###To do this, add use = "complete.obs" in parenthesis

  #Correlation test: cor.test(x,y)

    ##Default method here is also Pearson; need to remove missing values to avoid error message

#CREATING CORRELATION MATRIX:

  #Create a vector w/ variables to include in matrix:

    ##data <-COPD[,c("AGE", "PackHistory", "FEV1")]

    ##CREATE Correlation Matrix Vector, assigning correleation coefficients of the different variables:

      ###cor\_matrix <- cor(data)

  #View the Matrix: cor\_matrix to view output and round(cor\_matrix,2) to round output to 2 decimals

  #Visualize correlation b/t variables, ie correlation plot: pairs(~variable1+variable2+variable3, data =dataframe)

#Multiple Linear Regression:

  #modelname <- lm(output~predictor1 +predictor2, data = dataframe)

  #View regression output: summary(modelname)

  #view model's 95% intervals: confint(modelname)

  #Examine VIF using imcdiagF() function from 'mctest' package

#Regression w/ Categorical Variables -- 2 ways to do this:

  #Check what the variable is saved as, change it to factor variable if it's not saved as such

  #Check what variable has beeen saved as using class() function

  #If not saved as factor, change using factor () command:

    ###variable <-factor(variable)

  #Run the regression as normal

  #Include factor () before the variable in the regression model:

    ##modelname <- lm(outout~predictor1 + factor (predictor2), data = dataframe)

#CHANGING THE REFERENCE CATEGORY OF A VARIABLE:

  #use relevel() function in the following format:

    ##variable <- relevel(variable, ref=newrelevel) w/ newrelevel being new reference level,

    ###written as either numeric (1,2,3,) or character - in which case needs to be written within

      ###apostrophes - "MILD", "SEVERE"...

#CHANGING DATA TYPE FOR A VARIABLE:

  #check what variable has been as class() function

  #CHANGE DATA TYPE:

    #as.numeric()

    #as.character()

    #as.factor()

    #as.integer()

#CREATING A NEW VARIABLE ON R: eg variable 'comorbid'

  #We created variable COMORBID. This variable was to be binary, and indicated presence of

    ##<1 comorbidity ('1), or the complete absence of comorbidities ('0') based on responses to variables:

      ##Diabetes, muscular, hypertension, AtrialFib, and IHD

  #check that all variables are saved as correct datatype

  #create an empty vector of correct length

  #Comorbid will be same length as other variables, so: comorbid <- length(COPD$Diabetes)

  #assign values to a vector

  #We want comorbid = 1 when Diabetes OR muscular OR hypertension OR AtrialFib OR IHD = 1

    #so,  comorbid[COPD$Diabetes == 1 | COPD$muscular == 1 | COPD$hypertension == 1 | COPD$AtrialFib == 1 | COPD$IHD == 1] <- 1

  #assign 1 to values meeting set conditions and NA to those not meeting conditions

  #comorbid = 0 when ALL above variables = 0.

    #So: comorbid[is.na(comorbid)] <-1

      #convert this variable to 'factor'

  #Optional: add he variable to the dataset using the following command:

    ##COPD$comorbid <- comorbid

#REGRESSION WITH INTERACTION EFFECT:

  #Use the same format as a Multiple Linear Regression, but incldue both terms, ie:

    #modelname <- lm(outcome~predictor1 + predictor2 + (predictor1\*predictor2), data=dataframe)

  #interpretation of interaction effect can be simplified using () function from the 'prediction' package