Statistics – Study Guide

EA 30 Spring 2024

2024-04-21

Document History

Round One

Each group will write two questions (and answers). Then at the top of each section, write out a short study guide that might include useful functions to understand, a link to a video, or "psuedo" code that might help.

- Group 1: 1a and 8b – Creating Vectors
- Group 2: 2a and 7b – Creating Dataframe in R
- Group 3: 3a and 6 – Summary Statistics
- Group 4: 4a and 5b – ANOVA
- Group 5: 5a and 3b – T-Tests
- Group 6: 6a and 4b – Correlation/Linear Regression
- Group 7: 7a and 1b – Contingency Tables/Tests for Association
- Group 8: 8a and 2b – Logistic Regression

Here's a link to the google doc

Round Two

New Groups:

- Group 1: vectors and test for association (contingency tables)
- Group 2: dataframes and logistic regression
- Group 3: summary statistics (and a probability distribution function image)
- Group 4: ANOVA and boxplots and causuality
- Group 5: t-tests and histograms
- Group 6: linear regression and correlation (what's the difference)

1. Copy Rmd into Rstudio and knit and make sure it works.
2. For each section, we would like our peers to have some background on topic and why the information is useful. If you link to a video, please describe what the viewer is going to learn.
3. Adjust your question to make it align with the course content. Change the examples to environmental data types. All datasets should have an EA theme.
4. Use vectors and data frames in problem sets as much as possible.
5. Make sure the code works and generates an output. The guide should work!
6. Provide some help to reader to interpret results.
7. Add questions that addresses the flow chart above – ask the reader to consider what type of analysis it might be and why!

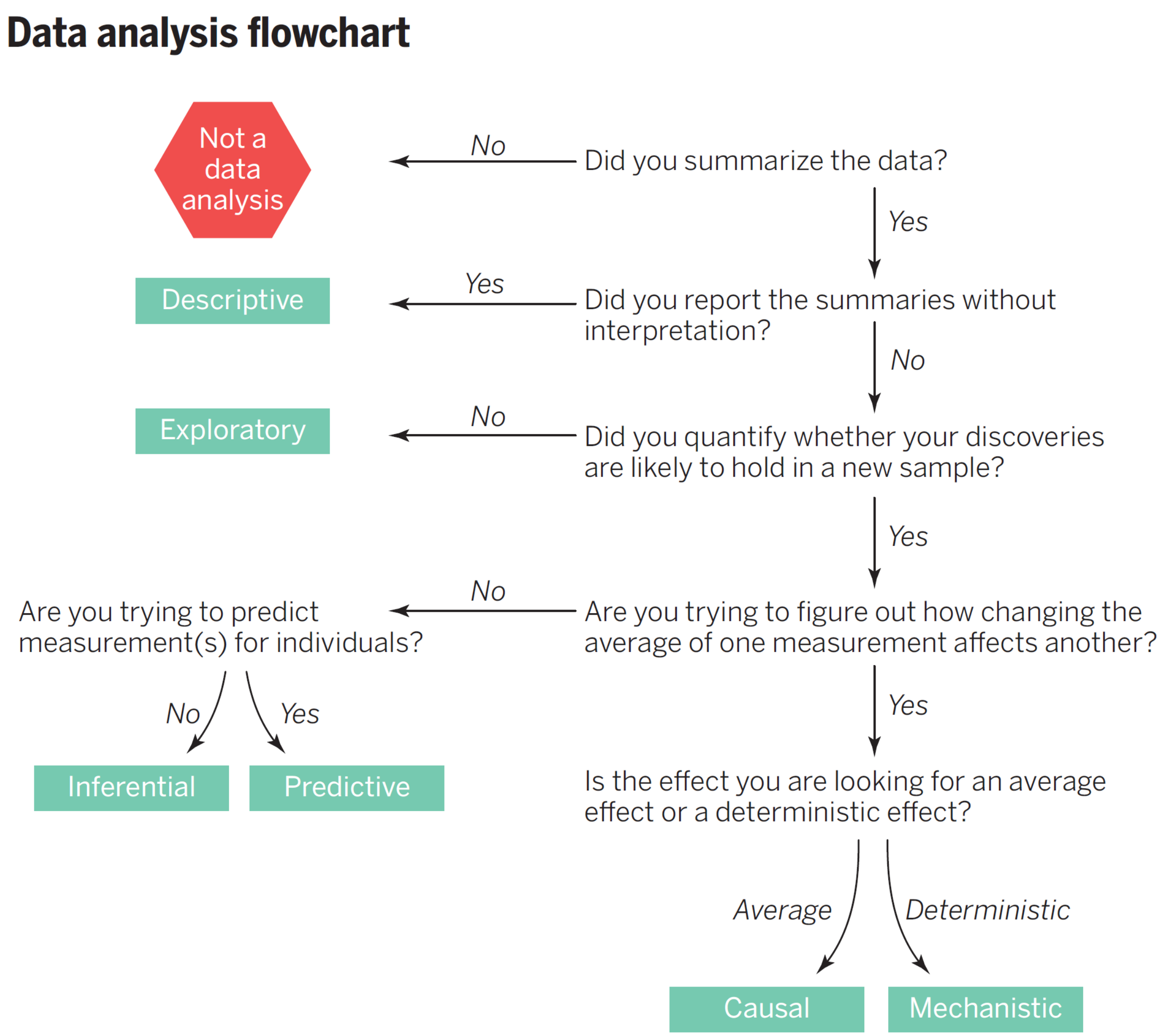


Figure 1: Data Analysis Flow Chart

8. Cut the sections of the Rmd that you are not working on and when you are done (don't cut the preamble stuff).
9. please submit a working Rmd file on today's canvas assignment and change the folks in your group in the "author" in the Rmd file.

Round Three:

Marc's Synthesis

1. I combined each submission, tricky since some didn't cut out the sections they didn't work on, so I had to compare line-by-line. Must be a better way!
2. I formatted each section so I could a "stand alone" lesson that I will make into a video during the week of the 21st of April.

Videos

3. Created a dozen or so videos so that students have a idea of what to do for their pilot project analyses.

Peer Evaluation

4. Each group will watch a movie and make suggestions to improve content and explanation

Group 1: Section 1, Group 2: Section 2 Group 3: Section 3 Group 4: Section 4 Group 5: Section 5 Group 6: Section 6 Group 7: Section 7 Group 8: Section 8

Working with R

1. Vectors

Q1a. What are vectors and how are they used?

Answer: They are very versatile and can store multiple variables in a single date type.

Q1b: Create a vector of character values

Step-by-step guide:

- Determine the strings you want to include in your vector.
- Use the `c()` function to combine these strings.
- Assign the result to a variable.
- Print the variable to see the result.

Utilize the c() function to combine character strings into a vector:

```
fruits_vector <- c("apple", "banana", "cherry")
```

Answer:

```r
print(fruits_vector)
```

```
## [1] "apple"  "banana" "cherry"
```

1c: Create a vector of numbers

Tree hight can be used to estimate the shading of a park. You have surveyed the tree heights in several open areas in the 5C and will use these to estimate the amount of shade on campus.

You've surveyed all the trees in your local park and recorded their height, in feet, as follows: 50, 25, 36, 27, 82, 39, 55, 62

```r
tree_height <- c(50,25,36,27,82,39,55,62,33)
print(tree_height)
```

```
## [1] 50 25 36 27 82 39 55 62 33
```

1d. Vector of Rainfall Data

We going to create a vector of random numbers that represent 25 years of rainfall data using a lognormal distribution with the following parameter: mean = 25, sd = 10.

```r
set.seed(11144)
rainfall_notrend <- rlnorm(n =25, meanlog = 1.5, sdlog =1); #hist(rainfall)
```

To make the data more interesting, we'll add a trend to the data, where amount goes up by 1% of the mean each year. Note: I should have this amoratized using a loop, but didn't have time to write the code.

```r
rainfall_trend <- rainfall_notrend * (1 + seq(0, 0.24, 0.01))
rainfall_trend
```

```
##  [1]  1.2120491  9.5115004  4.7439357  4.1222628  1.7939328 21.0771292
##  [7]  9.2387824  3.5641112  0.5602178  3.7836034  3.4865786  9.0865422
## [13]  5.2431000 32.2317060 36.7557453  5.6849292  2.1658275  4.1065391
## [19]  3.6924331 13.1756911 52.9114984  1.6177637 14.0603812  3.3456330
## [25]  4.7821297
```

(1 + seq(0, 0.24, 0.01)) print(rainfall)

2. Dataframes in R

Introduction

Introductory video: R Tutorial - Using the Data Frame in R

A data frame is a collection of data in rows and columns

- Rows = entries (observations)
- Columns = variables (composted of vectors)

A dataframe can only store vectors of the same length, you you might get an error, see below.

Create a data frame with two vectors (vector_1 and vector_2)

Ranking	Animal	Country	Avg. Sleep Hours
1	Koala	Australia	21
2	Hedgehog	Italy	18
3	Sloth	Peru	17
4	Panda	China	10

Question 2a: Creating a Dataframe:

How do you create a complete data frame with the following information? Draw an example table for all viable scenarios

Scenario 1:

```
Beach = c("Malibu", "Laguna")
Mountain = c("Baldy", "Mammoth")

Scenario1 = data.frame(Beach, Mountain)
show(Scenario1)
```

```
##    Beach Mountain
## 1 Malibu    Baldy
## 2 Laguna  Mammoth
```

```
Desert= c("Sonoma", "Mojave")
Forest = c("Sequoia", "Redwood")
Tundra = c("Siberia", "Iceland")
```

1. Variables (Classes or Categories): Fruit, Vegetable
2. Entries: Apple, Banana, Celery, Kale

Scenario 2:

- Variables: Birds, Dogs, Cats
- Entries: Parrot, Eagle, Husky, Rag Doll, Siamese

Answer:

```
Fruit = c("Apple", "Banana")
Vegetable = c("Celery", "Kale")

Scenario1 = data.frame(Fruit, Vegetable)
```

```
Birds= c("Parrot", "Eagle")
Dogs = c("Husky", "Rag Doll")
Cats = c("Siamese")

Scenario2 = data.frame(Birds, Dogs, Cats)
```

We cannot create a dataframe for scenario 2 because the Dog vector has one less entry than the other two vectors.

MLH NOTE: I didn't get an error, not sure what the goal is here.

Create a dataframe based on the following table:

Answer:

MLH NOTE: produces an error…not sure what the goal of the line Summary Statistics.

```r
ranking <- 1:4
animal <- c('koala', 'hedgehog', 'sloth', 'panda')
country <- c('Australia', 'Italy', 'Peru', 'China')
avg_sleep_hours <- c(21, 18, 17, 10)
super_sleepers <- data.frame(ranking, animal, country, avg_sleep_hours)
print(super_sleepers)
```

```
##   ranking   animal   country avg_sleep_hours
## 1       1    koala Australia              21
## 2       2 hedgehog     Italy              18
## 3       3    sloth      Peru              17
## 4       4    panda     China              10
```

```r
# Summary Statistics -- COMMENTED OUT PRODUCED ERROR
```

2c. Data Frame with Rainfall

We created 25 years of data using a lognormal distribution and then create a trend in the vector. Now we will create a dataframe with the following columns: Year, Rainfall_Notrend, and Rainfall_Trend.

```r
year <- 1995:2019
rainfall.df <- data.frame(year, rainfall_notrend, rainfall_trend)
```

Now let's check the dataframe:

```r
str(rainfall.df)
```

```
## 'data.frame':    25 obs. of  3 variables:
##  $ year            : int  1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 ...
##  $ rainfall_notrend: num  1.21 9.42 4.65 4 1.72 ...
##  $ rainfall_trend  : num  1.21 9.51 4.74 4.12 1.79 ...
```

```r
head(rainfall.df)
```

```
##   year rainfall_notrend rainfall_trend
## 1 1995         1.212049       1.212049
## 2 1996         9.417327       9.511500
## 3 1997         4.650917       4.743936
## 4 1998         4.002197       4.122263
## 5 1999         1.724935       1.793933
## 6 2000        20.073456      21.077129
```

We will analyze this using a 7a Linear Regression of Rainall Data

3. Matrices

Matrices are used to store data in a two- or mult-dimensional format. They are similar to data frames, but all elements must be of the same data type. Once you have created a matrix, you can perform mathematical operations on it, such as addition, subtraction, and multiplication – following the rules of matrix algebra.

For our purposes, we'll make a matrix of the following data:

Step-by-step guide:

-

	Bees	Butterflies	Beetles
Vegetable Garden	1	2	2
Flower Garden	15	6	22
Herb Garden	7	11	15

Q1: Using the data below, create a contingency table

```
pollinators_type <- c("Bees", "Butterflies", "Beetles")
garden_type <- c("Vegetable garden", "Flower garden", "Herb garden")

pollinator_data = c(1, 2, 2, 15, 6, 22, 9, 11, 15)
# Generate fake data
pollination_matrix = matrix(pollinator_data, nrow = 3, ncol = , byrow=TRUE)

pollination_matrix
```

```
##      [,1] [,2] [,3]
## [1,]    1    2    2
## [2,]   15    6   22
## [3,]    9   11   15
```

```
#pollinator_data <- sample(pollinators, 100, replace = TRUE)
#garden_data <- sample(gardens, 100, replace = TRUE)
```

Sometimes it nice to name the rows and columns to make sure you can keep track of the data.

```
colnames(pollination_matrix) <- garden_type
rownames(pollination_matrix) <- pollinators_type
```

```
pollination_matrix
```

```
##             Vegetable garden Flower garden Herb garden
## Bees                       1             2           2
## Butterflies               15             6          22
## Beetles                    9            11          15
```

4. Summary Statistics

So, what are summary statistics? Check out this 5-minute video:

Basic summary statistics in R. Pay close attention to the code used to find the mean, median, and standard deviation.

Question 3a:

Question 3b: When might it be helpful to use summary statistics?

Answer: When you want to evaluate a dataset and learn more information about the distribution! It is a helpful first step before undertaking more complex tests on the data

Question 3b:

In your RStudio, load the dataset cars. This is built into the software!

Now, please provide some of the summary statistics for this dataset. This includes the mean and median for speed and distance. It might be helpful to View the dataset before starting to orient yourself within the data.

No code given…

```r
# No code given
```

View: this is what the first 15 columns should look like

Mean Values: speed = 15.4, distance = 42.98

Median Values: speed = 15.00, 36.00

Mean, Median, and Standard Devivation

Visualing Distrbutions: Histogram

Make a histogram representing the distribution of tree height in your local park!

Answer:

```r
hist(tree_height)
```

9b: Double Variable Histogram

You want to compare tree heights in different nearby parks You record the tree height at another park as: 60,27,70,29,80,39,40,60

Answer:

```r
tree_height <- c(50,25,36,27,82,39,55,62)
tree2_height <- c(60,27,70,29,80,39,40,60)
p1<-hist(tree_height)
```


```r
p2<-hist(tree2_height)
```

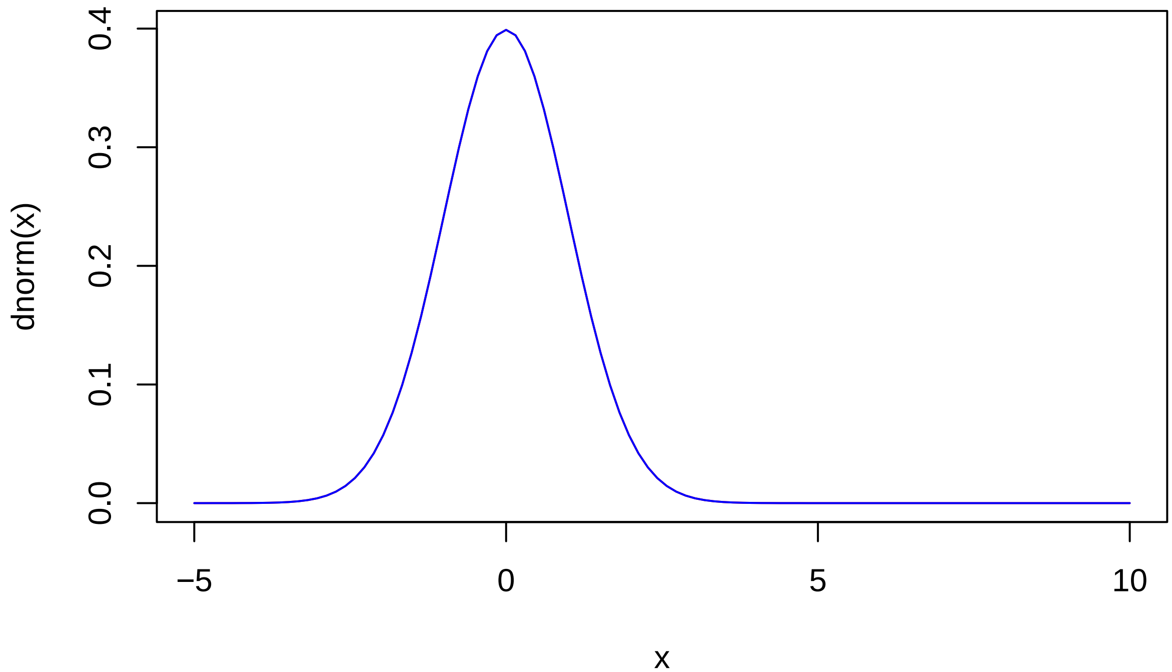
```r
plot( p1, col=rgb(0,0,1,1/4), xlim=c(0,100))  # first histogram
plot( p2, col=rgb(1,0,0,1/4), xlim=c(0,100), add=TRUE)
```

Probabaility Distributions

Instead of using a histogram, we can also use a theoretical probability distribution to represent the data.

In the example below, we used both the normal probability distribution and the lognormal probability distribution. Which one do think models the data better? I suggest you compare thest to a histogram, data from 1d. Vector of Rainfall Data

```r
x= 1:25
x<-seq(from=-5,to=10,length.out=100)
plot(x,dnorm(x), ty='l', col='red')
lines(x,dnorm(x), ty='l', col='blue')
```



Question 1b: Why are vectors commonly used in graphing data?

Role of Summary Statitiscs in Data Analysis

4. Tests for Association

Test for association are use to determine if two or more categorical variables are independent. If we can reject the hypothesis, then we can conclude that there is an association between the variables.

Contingency tables are very useful when we need to condense a large amount of data into a smaller format, to make it easier to gain an overall view of our original data. To create contingency tables in R we will be making use of the table() function.

In statistics, a contingency table (also known as a cross tabulation or crosstab) is a type of table in a matrix format that displays the multivariate frequency distribution of the variables.

– What kind of variables: Categorical (nominal or ordinal with a few categories) – ommon Applications: Association between two categorical variables.

The chi-squared test tests the hypothesis that there is no relationship between two categorical variables. It compares the observed frequencies from the data with frequencies which would be expected if there was no relationship between the variables. We use a fisher test when the sample size is small.

Use the table(row variable $ column variable) function to parse the data https://bookdown.org/kdonovan125/ibis_data_analysis_r4/working-with-tables-in-r.html

Simple explanation from R website https://bookdown.org/kdonovan125/ibis_data_analysis_r4/working-with-tables-in-r.html

Determine if there is an assocation between the pollinator type and garden type (See data in Section 2)

```
fisher.test(pollination_matrix)
```

```
##
##  Fisher's Exact Test for Count Data
```

```
##
## data:  pollination_matrix
## p-value = 0.3241
## alternative hypothesis: two.sided
```

chisq.test(df*Pollinator*, *df*Garden_Type)

5. t-Test: Comparing Two Populations

To calculate a t-test, you need the mean values from each data set, the standard deviation, and how many data values there are.

Pseudo code t.test(x, y, alternative = c("two.sided", "less", "greater"), mu = 0, paired = FALSE, var.equal = FALSE, conf.level = 0.95, ...)

https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/t.test

4a: What is a t-test? What information do you need to know in order to conduct a T-test?

Answer: T-tests are a type of inferential statistic used to determine statistical significance between the means of two variables. T-tests are used to test whether or not a hypothesis is statistically significant.

T-tests can be dependent or independent.

4b: What is a practical use of a t test?

Comparing efficacy of different trial medications or treatments

4c: Perform a two-sample t-test to compare the means of the treatment and control groups.

Fake data set:

```
# Group 1 (Treatment Group):
25, 28, 30, 32, 26
# Group 2 (Control Group):
20, 22, 24, 21, 23
```

NOTE: not sure how this would work given the code above, and change to eval=FALSE because of errors.

What type of graph is useful for showing a t-test?

Boxplot Command: boxplot()

```
# no code given for answer
```

6. ANOVA: Comparing Three or More Populations

Link to Video: Analysis of Variance (ANOVA) in R

(very helpful explanation)

Helpful tips:

The summary() function provides an ANOVA table, which includes statistics such as the F-statistic, p-value, and degrees of freedom. These statistics help to determine whether the differences between group means are statistically significant. You can use ANOVA to analyze data for your NSF experiment! Especially if you have different groups you are trying to compare and analyze b) What is the ANOVA statistical test? Statistics 101: One-way ANOVA, A Visual Tutorial - YouTube ANOVA stands for analysis of variance It is used to categorize differences in data, or variances, into different groups for future statistical tests A one-way ANOVA is for three or more groups of data If there is no variance the ANOVA f-ratio will be 1

Question 5a: What is ANOVA used for?

Answer: ANOVA (Analysis of Variance) in R is a statistical technique used to see how a quantitative dependent variable changed compared to categorical independent variables

What code would you use for ANOVA on this fake dataset?

Generate fake data

```r
group1 <- rnorm(30, mean = 10, sd = 2)
group2 <- rnorm(30, mean = 12, sd = 2)
group3 <- rnorm(30, mean = 15, sd = 2)
```

Combine data into a dataframe

```r
fake_data <- data.frame(
  Group = rep(c("A", "B", "C"), each = 30),
  Dependent_Variable = c(group1, group2, group3)
)
```

Show the first few rows of the dataset

```r
head(fake_data)
```

```
##   Group Dependent_Variable
## 1     A           9.224329
## 2     A          10.147446
## 3     A           7.573901
## 4     A          13.454273
## 5     A           9.233632
## 6     A           9.753319
```

Answer:

```r
result <- aov(Dependent_Variable ~ Group, data = fake_data)
summary(result)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)    
## Group        2  300.5  150.26   46.15 2.18e-14 ***
## Residuals   87  283.3    3.26                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

What is an example of ANOVA code?

```r
# Fake data set
group1 <- c(35, 38, 42, 40, 36)
group2 <- c(28, 32, 30, 34, 31)
group3 <- c(45, 48, 50, 47, 49)

# Combine data into a data frame
data <- data.frame(Group = rep(c("Group1", "Group2", "Group3"), each = 5),
                   Value = c(group1, group2, group3))

# Run ANOVA test
result <- aov(Value ~ Group, data = data)

# Summary of ANOVA
summary(result)
```

```
##             Df Sum Sq Mean Sq F value  Pr(>F)
## Group        2  710.4   355.2   63.05 4.3e-07 ***
## Residuals   12   67.6     5.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

7. Linear Regression

Question 6a: Make a linear regression figure based on the data below

Fake data set: Age: 1, 2, 3, 4, 4, 5, 7, 8, 8, 10, 13 Height(in): 30, 37, 40, 42, 43, 48, 49, 53, 50, 62

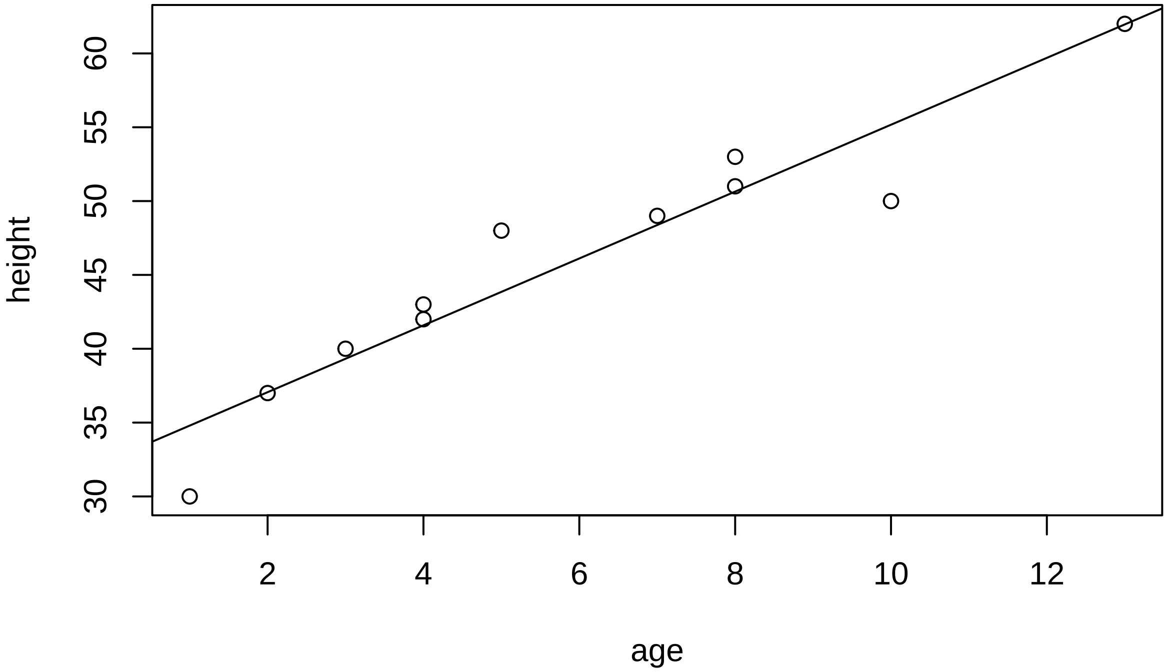
Answer:

```r
age <- c(1, 2, 3, 4, 4, 5, 7, 8, 8, 10, 13)
height <- c(30, 37, 40, 42, 43, 48, 49, 51, 53, 50, 62)

kids <- lm(height ~ age)
summary(kids)
```

```
##
## Call:
## lm(formula = height ~ age)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.1710 -0.0109  0.4131  1.0451  4.1491
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  32.5308     1.7240  18.869 1.52e-08 ***
## age           2.2640     0.2515   9.003 8.51e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.899 on 9 degrees of freedom
## Multiple R-squared:  0.9001, Adjusted R-squared:  0.889
## F-statistic: 81.06 on 1 and 9 DF,  p-value: 8.514e-06
```

```r
plot (height ~ age)
abline (coef(kids))
```



Question 6a: Make a linear regression figure based on the data below

Fake data set: Age: 1, 2, 3, 4, 4, 5, 7, 8, 8, 10, 13 Height(in): 30, 37, 40, 42, 43, 48, 49, 53, 50, 62

Answer:

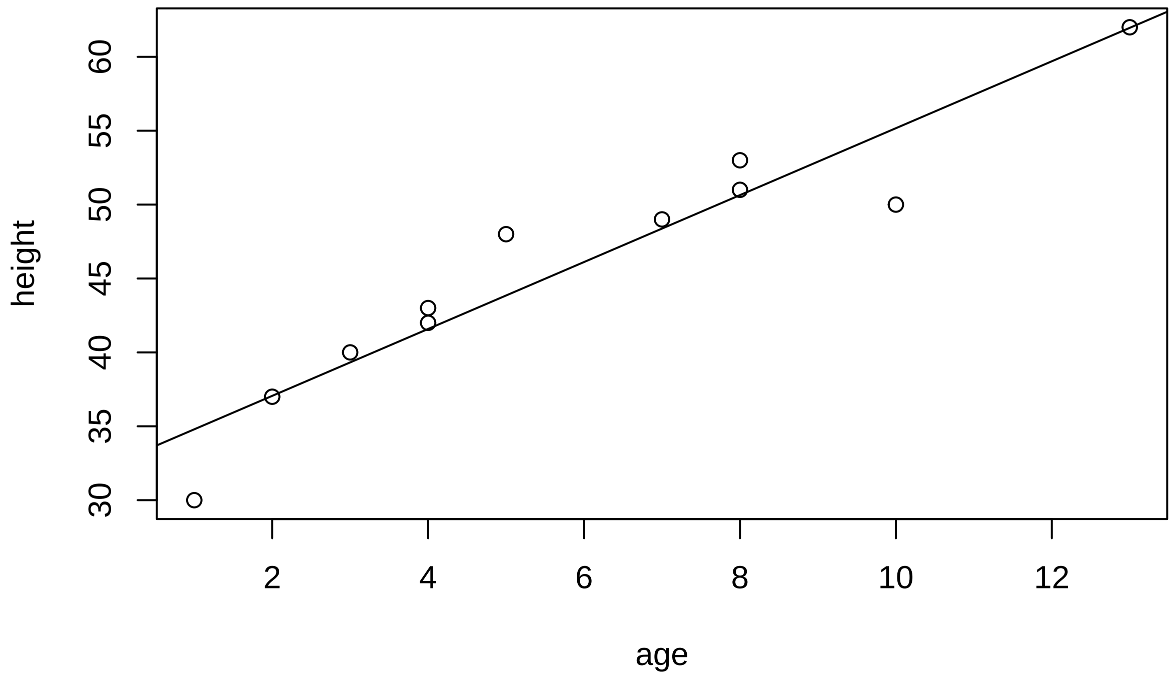
```r
age <- c(1, 2, 3, 4, 4, 5, 7, 8, 8, 10, 13)
height <- c(30, 37, 40, 42, 43, 48, 49, 51, 53, 50, 62)

kids <- lm(height ~ age)
summary(kids)
```

```
##
## Call:
## lm(formula = height ~ age)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.1710 -0.0109  0.4131  1.0451  4.1491
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  32.5308     1.7240  18.869 1.52e-08 ***
## age           2.2640     0.2515   9.003 8.51e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.899 on 9 degrees of freedom
## Multiple R-squared:  0.9001, Adjusted R-squared:  0.889
```

```
## F-statistic: 81.06 on 1 and 9 DF,  p-value: 8.514e-06
```

```r
plot (height ~ age)
abline (coef(kids))
```



Q6b: What relationship does a linear regression model describe?

Answer: A linear regression model describes the relationship between a dependent variable, y, and one or more independent variables, X.

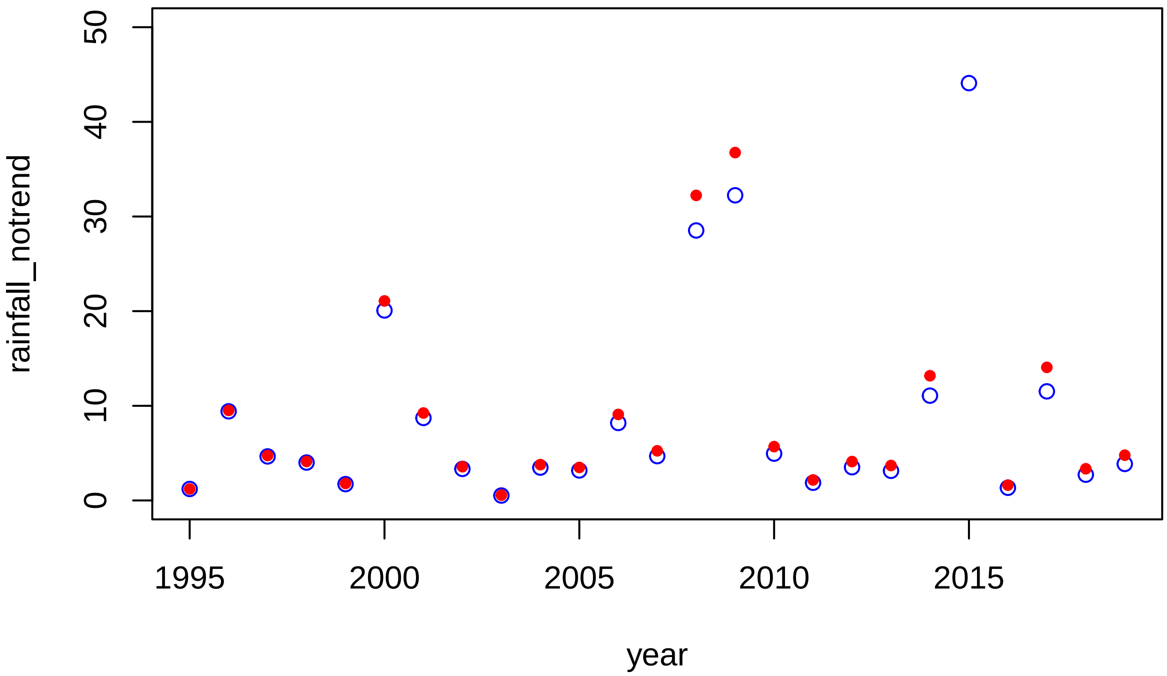
Q: What command in R allows you to review the models coefficients and see detailed information about the model

A:summary(lmheight)

7a Linear Regression of Rainall Data

First, we plot the data:

```r
plot(rainfall_notrend ~ year, rainfall.df, col = "blue", ylim=c(0, 50))
points(rainfall_trend ~ year, rainfall.df, col = "red", pch=20)
```



Then we run the linear regression, let's loot at both the trend and nontrend:

```
lm_notrend <- lm(rainfall_notrend ~ year, data = rainfall.df)
lm_trend <- lm(rainfall_trend ~ year, data = rainfall.df)

summary(lm_notrend)
```

```
##
## Call:
## lm(formula = rainfall_notrend ~ year, data = rainfall.df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.857 -6.655 -4.262  0.392 33.156
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -507.4102   614.3958  -0.826    0.417
## year           0.2572     0.3061   0.840    0.409
##
## Residual standard error: 11.04 on 23 degrees of freedom
## Multiple R-squared:  0.02979,    Adjusted R-squared:  -0.0124
## F-statistic: 0.7061 on 1 and 23 DF,  p-value: 0.4094
```

```
summary(lm_trend)
```

```
##
## Call:
## lm(formula = rainfall_trend ~ year, data = rainfall.df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -11.718  -7.781  -4.704   0.564  39.938
##
```

```
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -716.3243   712.5932  -1.005    0.325
## year           0.3619     0.3551   1.019    0.319
##
## Residual standard error: 12.8 on 23 degrees of freedom
## Multiple R-squared:  0.04323,    Adjusted R-squared:  0.001628
## F-statistic: 1.039 on 1 and 23 DF,  p-value: 0.3186
```

The "Estimate" for the year, is the slope of the line and the p-value is the significance of the slope. In both cases, we cannot reject the null hypothesis that the slope is zero, i.e. there is no trend.

How could we adjust the trend data to make it significant?

8. Logistic Regression

What is a logistic regression? Watch this 8 min video to find out!

Q8a: What is a logistic regression and what are they used for?

Logistic regression estimates the probability of an event occurring, such as voted or didn't vote, based on a given data set of independent variables. They are used commonly for prediction and classification problems. They are best used in scenarios where there is a binary outcome (ie. yes or no, black of white, etc.)

Q8b: What are the characteristics of data sets that can be used to create logistic regression?

answer : Logistics regression graphs can be made with data sets that contain continuous variables (Like temperature, or depth) and a binary/ categorical outcome variable (yes/no , 1/0, on /off). Sample sizes are usually larger and assumes a linear relationship between variables and categorical outcomes.

See this UCLA website that describes how to use a logistic regression in R and gives examples.

```r
Customer_ID <- c(1, 2, 3, 4, 5, 6, 7, 8, 9)
Age <- c(55, 19, 50, 21, 45, 46, 63, 33, 44)
Gender <- c("Male", "Male", "Female", "Male", "Male", "Female", "Female", "Male", "Male")
Item_Purchased <- c("Blouse", "Sweater", "Jeans", "Sandals", "Blouse", "Sneakers", "Shirt", "Shorts", "C
Category <- c("Clothing", "Clothing", "Clothing", "Footwear", "Clothing", "Footwear", "Clothing", "Clot
Purchase_Amount_USD <- c(53, 64, 73, 90, 49, 20, 85, 34, 97)

data <- data.frame(Customer_ID, Age, Gender, Item_Purchased, Category, Purchase_Amount_USD)
```

```r
library(ggplot2)
data$Gender <- factor(data$Gender)
data$Purchase_Above_70 <- ifelse(data$Purchase_Amount_USD > 70, 1, 0)
logit_model <- glm(Purchase_Above_70 ~ Age + Gender + Category, data = data, family = "binomial")
summary(logit_model)
```

```
##
## Call:
## glm(formula = Purchase_Above_70 ~ Age + Gender + Category, family = "binomial",
##     data = data)
##
## Deviance Residuals:
##        1         2         3         4         5         6         7         8
```

```
## -0.45176  -0.83416   0.85408   1.60333  -0.53924  -1.60333   1.04018  -0.66294
##        9
##  0.00022
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)          2.70140    5.41976   0.498    0.618
## Age                 -0.03761    0.09168  -0.410    0.682
## GenderMale          -2.86354    2.74842  -1.042    0.297
## CategoryFootwear    -0.00956    2.16046  -0.004    0.996
## CategoryOuterwear   19.38322 3956.18064   0.005    0.996
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 12.3653  on 8  degrees of freedom
## Residual deviance:  8.5829  on 4  degrees of freedom
## AIC: 18.583
##
## Number of Fisher Scoring iterations: 16
```