assignment_7.2_VenkidusamyKesavAdithya

Kesav Adithya Venkidusamy

10/11/2021

Student Survey Analysis

As a data science intern with newly learned knowledge in skills in statistical correlation and R programming, you will analyze the results of a survey recently given to college students. You learn that the research question being investigated is: "Is there a significant relationship between the amount of time spent reading and the time spent watching television?" You are also interested if there are other significant relationships that can be discovered? The survey data is located in this StudentSurvey.csv file.

```
#Load required libraries for analysis
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(pastecs)
```

```
##
## Attaching package: 'pastecs'

## The following objects are masked from 'package:dplyr':
##
##     first, last
```

```
#Create data frame for the input survey file
stud_survey <- read.csv("E:/Personal/Bellevue University/Course/github/dsc520/data/student-survey.csv")

#Print column and row names
print(attributes(stud_survey))
```

```
## $names
## [1] "TimeReading" "TimeTV"      "Happiness"   "Gender"
##
## $class
## [1] "data.frame"
##
## $row.names
##  [1]  1  2  3  4  5  6  7  8  9 10 11
```

```
#Print the dimension of the dataframe
print(dim(stud_survey))
```

```
## [1] 11  4
```

```
#Calculate the str for the dataframe
print(str(stud_survey))
```

```
## 'data.frame':    11 obs. of  4 variables:
##  $ TimeReading: int  1 2 2 2 3 4 4 5 5 6 ...
##  $ TimeTV     : int  90 95 85 80 75 70 75 60 65 50 ...
##  $ Happiness  : num  86.2 88.7 70.2 61.3 89.5 ...
##  $ Gender     : int  1 0 0 1 1 1 0 1 0 0 ...
## NULL
```

```
#Print the stats of the data
stud_survey |> stat.desc(norm=TRUE)
```

```
##                TimeReading       TimeTV   Happiness        Gender
## nbr.val       11.000000000  11.00000000  11.0000000 11.0000000000
## nbr.null       0.000000000   0.00000000   0.0000000  5.0000000000
## nbr.na         0.000000000   0.00000000   0.0000000  0.0000000000
## min            1.000000000  50.00000000  45.6700000  0.0000000000
## max            6.000000000  95.00000000  89.5200000  1.0000000000
## range          5.000000000  45.00000000  43.8500000  1.0000000000
## sum           40.000000000 815.00000000 806.3800000  6.0000000000
## median         4.000000000  75.00000000  75.9200000  1.0000000000
## mean           3.636363636  74.09090909  73.3072727  0.5454545455
## SE.mean        0.526959154   3.97824663   4.1059981  0.1574591643
## CI.mean.0.95   1.174138165   8.86408589   9.1487338  0.3508408816
## var            3.054545455 174.09090909 185.4514218  0.2727272727
## std.dev        1.747725795  13.19435141  13.6180550  0.5222329679
## coef.var       0.480624594   0.17808327   0.1857668  0.9574271078
## skewness      -0.002533230  -0.11848577  -0.5162276 -0.1582524145
## skew.2SE      -0.001917116  -0.08966855  -0.3906746 -0.1197634442
## kurtosis      -1.642178979  -1.03762883  -0.9143551 -2.1460055096
## kurt.2SE      -0.641769076  -0.40550884  -0.3573331 -0.8386661817
## normtest.W     0.920928865   0.98680678   0.9411966  0.6491717530
## normtest.p     0.326452517   0.99233227   0.5346664  0.0001051734
```

```
#TimeReading is in hours where as TimeTV is in mins. So, converting TimeReading to mins by multiplying
stud_survey_format <- stud_survey %>% mutate(TimeReadingMins = as.integer(60)*TimeReading)
print(str(stud_survey_format))
```

```
## 'data.frame':    11 obs. of  5 variables:
##  $ TimeReading    : int  1 2 2 2 3 4 4 5 5 6 ...
##  $ TimeTV         : int  90 95 85 80 75 70 75 60 65 50 ...
##  $ Happiness      : num  86.2 88.7 70.2 61.3 89.5 ...
##  $ Gender         : int  1 0 0 1 1 1 0 1 0 0 ...
##  $ TimeReadingMins: int  60 120 120 120 180 240 240 300 300 360 ...
## NULL
```

```
# Creating combinations of all the variables present in the data set

rt <- data.frame(stud_survey_format$TimeReadingMins, stud_survey_format$TimeTV)
rh <- data.frame(stud_survey_format$TimeReadingMins, stud_survey_format$Happiness)
rg <- data.frame(stud_survey_format$TimeReadingMins, stud_survey_format$Gender)

th <- data.frame(stud_survey_format$TimeTV, stud_survey_format$Happiness)
tg <- data.frame(stud_survey_format$TimeTV, stud_survey_format$Gender)

hg <- data.frame(stud_survey_format$Happiness, stud_survey_format$Gender)

variable_combine <- list(rt, rh, rg, th, tg, hg)


#1. Use R to calculate the covariance of the Survey variables and provide an explanation of why you wou

i = 1
for (df in variable_combine){
  print(paste0("The covariance for ", i ," set of variables"))
  print(cov(df[1], df[2]))
  i = i + 1
}
```

```
## [1] "The covariance for 1 set of variables"
##                                   stud_survey_format.TimeTV
## stud_survey_format.TimeReadingMins                 -1221.818
## [1] "The covariance for 2 set of variables"
##                                   stud_survey_format.Happiness
## stud_survey_format.TimeReadingMins                   -621.0055
## [1] "The covariance for 3 set of variables"
##                                   stud_survey_format.Gender
## stud_survey_format.TimeReadingMins                -4.909091
## [1] "The covariance for 4 set of variables"
##                           stud_survey_format.Happiness
## stud_survey_format.TimeTV                     114.3773
## [1] "The covariance for 5 set of variables"
##                           stud_survey_format.Gender
## stud_survey_format.TimeTV                0.04545455
## [1] "The covariance for 6 set of variables"
##                              stud_survey_format.Gender
## stud_survey_format.Happiness                  1.116636
```

Covariance results

Covariance helps to find the relation between the variables present in the data set. It is also used to measure the direction of the linear relationship between the data vectors.A positive covariance value indicates a positive linear relationship between the variables, and a negative value represents the negative linear relationship

From the output, we could see the covariance is negative for almost all the pairs of variables except TimeTV and Happiness which makes sense. The increase in happiness is because of time spent on watching TV.

2.Examine the Survey data variables. What measurement is being used for the variables? Explain what effect changing the measurement being used for the variables would have on the covariance calculation. Would this be a problem? Explain and provide a better alternative if needed.

Survey variables Analysis

TimeReading - The time spend reading appears to be measured in hours. So, we have created another variable called "TimeReadingMins" to convert hours to minutes.

TimeTV - The time spend on TV appears to be measured in minutes

Happiness - Happiness score appears to be in the range of 0-100 as min value is 45.67 and Max value is 89.52.

Gender - Gender is measured as binary; either 0 for male or female or vice versa. Need cook book for the confirmation

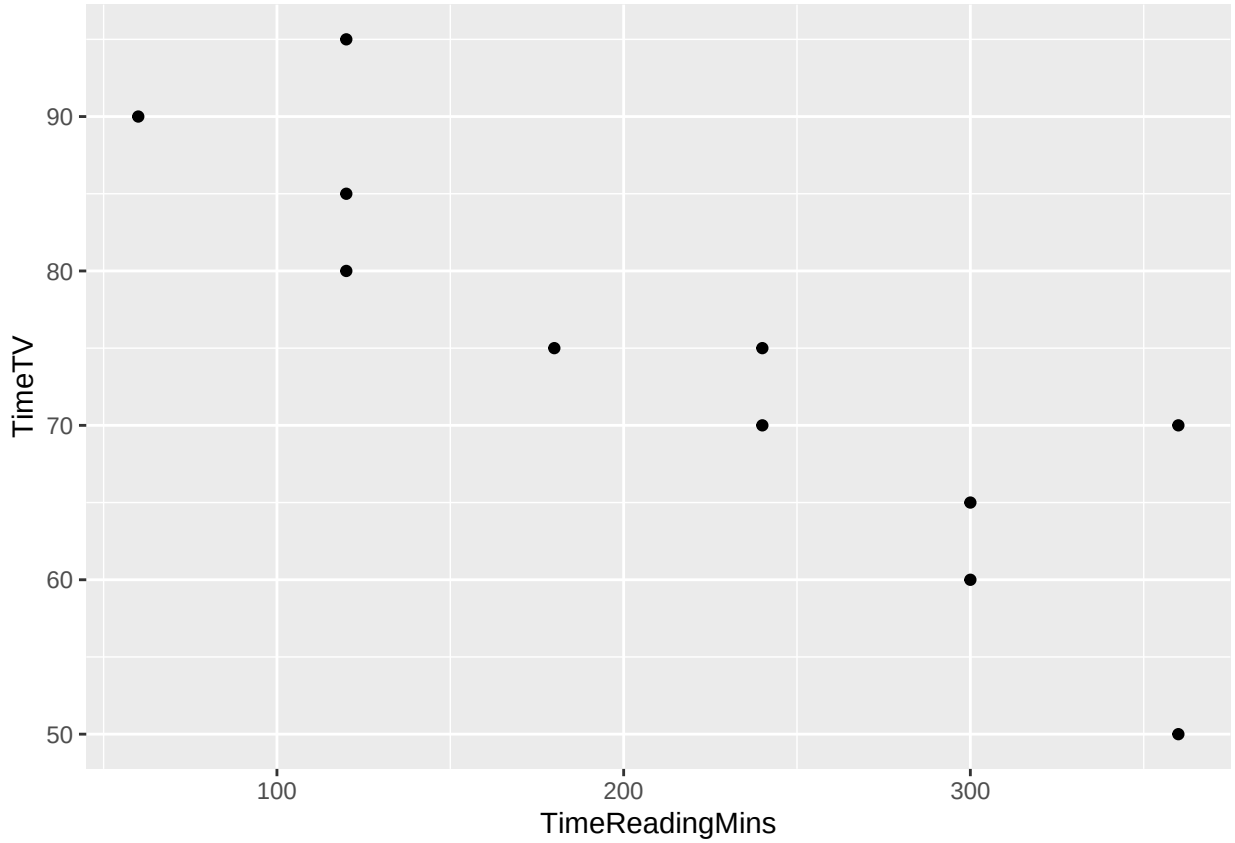
We have changed to TimeReading in hours to mins by creating a variable called TimeReadingMins so that it would be helpful to compare it with TimeTV which is in mins.

```
library(ggplot2)
library(dplyr)

stud_survey <- read.csv("E:/Personal/Bellevue University/Course/github/dsc520/data/student-survey.csv")

#Added variable to convert TimeReading in hours to TimeReading minutes
stud_survey_format <- stud_survey %>% mutate(TimeReadingMins = as.integer(60)*TimeReading)

ggplot(stud_survey_format,aes(x=TimeReadingMins,y=TimeTV)) + geom_point()
```




```
#3. Choose the type of correlation test to perform, explain why you chose this test, and make a predict

#Performing Shapiro test on the data to find if the variables are normally distributed

for (x in stud_survey_format) {
  print(shapiro.test(x))

}
```

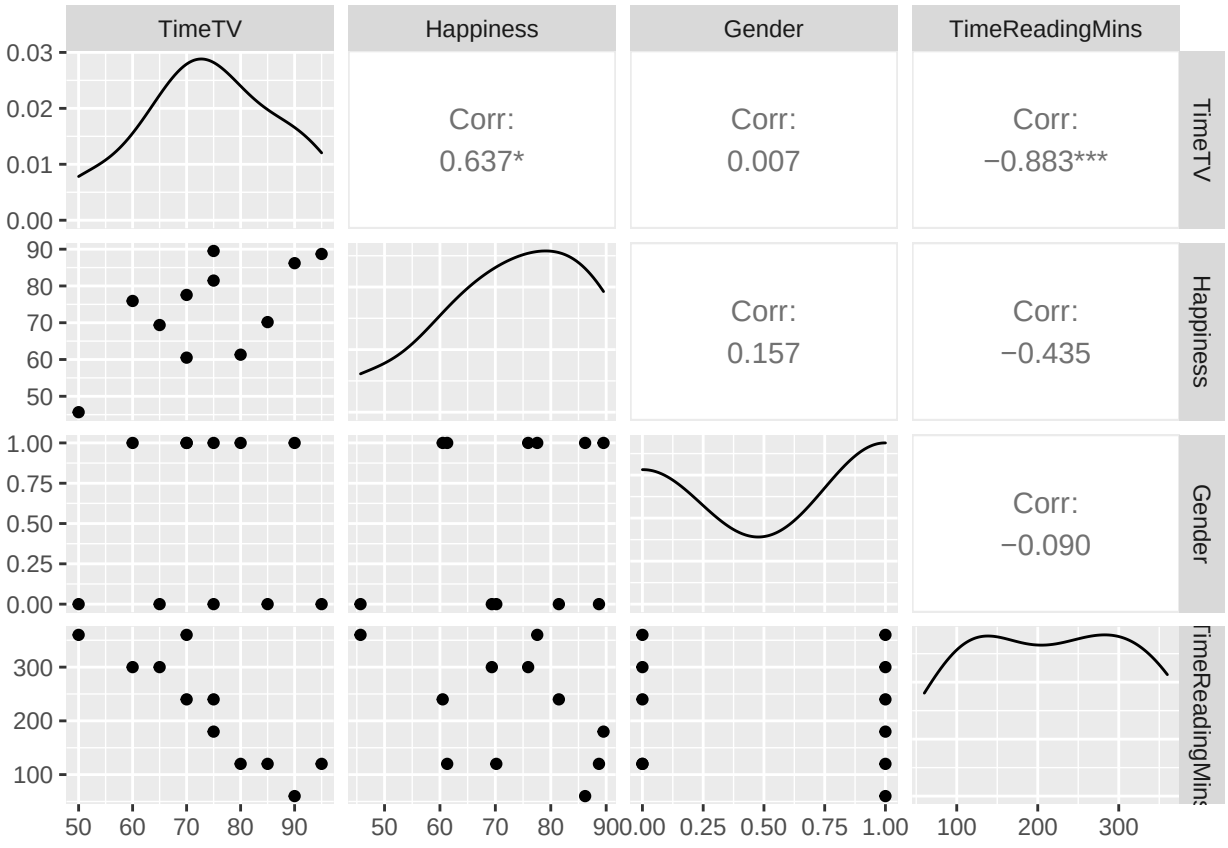
```
##
##  Shapiro-Wilk normality test
##
## data:  x
## W = 0.92093, p-value = 0.3265
##
##
##  Shapiro-Wilk normality test
##
## data:  x
## W = 0.98681, p-value = 0.9923
##
##
##  Shapiro-Wilk normality test
##
## data:  x
## W = 0.9412, p-value = 0.5347
```

```
##
##
##  Shapiro-Wilk normality test
##
## data:  x
## W = 0.64917, p-value = 0.0001052
##
##
##  Shapiro-Wilk normality test
##
## data:  x
## W = 0.92093, p-value = 0.3265
```

```
#We will use GGally package to compare TimeTV with all other variables. Here, High positive correlation

GGally::ggpairs(stud_survey_format[,c(2,3:5)])
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```




```
#Spearman correlation co-efficient would also better choice to calculate the relation between the varia

# Creating combinations of all the variables present in the data set
```

```r
rt <- data.frame(stud_survey_format$TimeReadingMins, stud_survey_format$TimeTV)
rh <- data.frame(stud_survey_format$TimeReadingMins, stud_survey_format$Happiness)
rg <- data.frame(stud_survey_format$TimeReadingMins, stud_survey_format$Gender)

th <- data.frame(stud_survey_format$TimeTV, stud_survey_format$Happiness)
tg <- data.frame(stud_survey_format$TimeTV, stud_survey_format$Gender)

hg <- data.frame(stud_survey_format$Happiness, stud_survey_format$Gender)

variable_combine <- list(rt, rh, rg, th, tg, hg)

spearman_test <- function(df){
  col_names <- colnames(df)
  result <- cor.test(df[,1],df[,2], mode = 'spearman')
  return(result)
}

for (df in variable_combine){
  names <- colnames(df)
  print(names)
  print(spearman_test(df))
}
```

```
## [1] "stud_survey_format.TimeReadingMins" "stud_survey_format.TimeTV"
##
##  Pearson's product-moment correlation
##
## data:  df[, 1] and df[, 2]
## t = -5.6457, df = 9, p-value = 0.0003153
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.9694145 -0.6021920
## sample estimates:
##       cor
## -0.8830677
##
## [1] "stud_survey_format.TimeReadingMins" "stud_survey_format.Happiness"
##
##  Pearson's product-moment correlation
##
## data:  df[, 1] and df[, 2]
## t = -1.4488, df = 9, p-value = 0.1813
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.8206596  0.2232458
## sample estimates:
##       cor
## -0.4348663
##
## [1] "stud_survey_format.TimeReadingMins" "stud_survey_format.Gender"
##
##  Pearson's product-moment correlation
##
```

```
## data:  df[, 1] and df[, 2]
## t = -0.27001, df = 9, p-value = 0.7932
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.6543311  0.5392294
## sample estimates:
##         cor
## -0.08964215
##
## [1] "stud_survey_format.TimeTV"    "stud_survey_format.Happiness"
##
##  Pearson's product-moment correlation
##
## data:  df[, 1] and df[, 2]
## t = 2.4761, df = 9, p-value = 0.03521
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.05934031 0.89476238
## sample estimates:
##      cor
## 0.636556
##
## [1] "stud_survey_format.TimeTV" "stud_survey_format.Gender"
##
##  Pearson's product-moment correlation
##
## data:  df[, 1] and df[, 2]
## t = 0.01979, df = 9, p-value = 0.9846
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.5956354  0.6040812
## sample estimates:
##         cor
## 0.006596673
##
## [1] "stud_survey_format.Happiness" "stud_survey_format.Gender"
##
##  Pearson's product-moment correlation
##
## data:  df[, 1] and df[, 2]
## t = 0.47695, df = 9, p-value = 0.6448
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.4889126  0.6917342
## sample estimates:
##       cor
## 0.1570118
```

```r
# 4.Perform Correlation analysis
#4.1 All Variables

for (df in variable_combine){
  print(cor(df[1],df[2],method="spearman"))
  print("\n")
```

```
}
```

```
##                                   stud_survey_format.TimeTV
## stud_survey_format.TimeReadingMins                -0.9072536
## [1] "\n"
##                                   stud_survey_format.Happiness
## stud_survey_format.TimeReadingMins                   -0.4065196
## [1] "\n"
##                                   stud_survey_format.Gender
## stud_survey_format.TimeReadingMins               -0.08801408
## [1] "\n"
##                             stud_survey_format.Happiness
## stud_survey_format.TimeTV                     0.5662159
## [1] "\n"
##                             stud_survey_format.Gender
## stud_survey_format.TimeTV               -0.02899963
## [1] "\n"
##                               stud_survey_format.Gender
## stud_survey_format.Happiness                 0.1154701
## [1] "\n"
```

```
#4.2 A single correlation between a pair of the variables
cor(stud_survey_format$TimeTV,stud_survey_format$Happiness, method="spearman")
```

```
## [1] 0.5662159
```

```
library(ppcor)
```

```
## Loading required package: MASS
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
pcor(stud_survey_format, method="spearman")
```

```
## Warning in pcor(stud_survey_format, method = "spearman"): The inverse of
## variance-covariance matrix is calculated using Moore-Penrose generalized matrix
## invers due to its determinant of zero.
```

```
## Warning in sqrt((n - 2 - gp)/(1 - pcor^2)): NaNs produced
```

```
## $estimate
##            [,1]       [,2]      [,3]       [,4]       [,5]
## [1,]  1.0000000 -0.9113002 0.3713077 -0.3424044 -1.0000000
## [2,] -0.9113002  1.0000000 0.5576799 -0.3566278 -0.9113002
## [3,]  0.3713077  0.5576799 1.0000000  0.2667871  0.3713077
```

```
## [4,] -0.3424044 -0.3566278 0.2667871  1.0000000 -0.3424044
## [5,] -1.0000000 -0.9113002 0.3713077 -0.3424044  1.0000000
##
## $p.value
##             [,1]        [,2]      [,3]      [,4]         [,5]
## [1,] 0.000000000 0.001630643 0.3651413 0.4064065 3.168317e-44
## [2,] 0.001630643 0.000000000 0.1509249 0.3858561 1.630643e-03
## [3,] 0.365141320 0.150924902 0.0000000 0.5230032 3.651413e-01
## [4,] 0.406406490 0.385856072 0.5230032 0.0000000 4.064065e-01
## [5,]         NaN 0.001630643 0.3651413 0.4064065 0.000000e+00
##
## $statistic
##           [,1]      [,2]      [,3]       [,4]          [,5]
## [1,]  0.000000 -5.421399 0.9795420 -0.8926760 -3.587120e+07
## [2,] -5.421399  0.000000 1.6457094 -0.9350380 -5.421399e+00
## [3,]  0.979542  1.645709 0.0000000  0.6780685  9.795420e-01
## [4,] -0.892676 -0.935038 0.6780685  0.0000000 -8.926760e-01
## [5,]       NaN -5.421399 0.9795420 -0.8926760  0.000000e+00
##
## $n
## [1] 11
##
## $gp
## [1] 3
##
## $method
## [1] "spearman"
```

#Analysis of above results

From the above result, we could interpret, TimeTV and TimeReading are having significant negative correlation. However, TimeTV and Happiness are having significant positive correlation.

```
#4.3 Repeat your correlation test in step 2 but set the confidence interval at 99%

library(ggplot2)
library(dplyr)

stud_survey <- read.csv("E:/Personal/Bellevue University/Course/github/dsc520/data/student-survey.csv")

#Added variable to convert TimeReading in hours to TimeReading minutes
stud_survey_format <- stud_survey %>% mutate(TimeReadingMins = as.integer(60)*TimeReading)

# Creating combinations of all the variables present in the data set

rt <- data.frame(stud_survey_format$TimeReadingMins, stud_survey_format$TimeTV)
rh <- data.frame(stud_survey_format$TimeReadingMins, stud_survey_format$Happiness)
rg <- data.frame(stud_survey_format$TimeReadingMins, stud_survey_format$Gender)

th <- data.frame(stud_survey_format$TimeTV, stud_survey_format$Happiness)
tg <- data.frame(stud_survey_format$TimeTV, stud_survey_format$Gender)

hg <- data.frame(stud_survey_format$Happiness, stud_survey_format$Gender)
```

```r
variable_combine <- list(rt, rh, rg, th, tg, hg)

spearman_test2 <- function(df){
  col_names <- colnames(df)
  result <- cor.test(df[,1],df[,2], mode = 'spearman',exact=FALSE, conf.level = .99)
  return(result)
}

for (df in variable_combine){
  names <- colnames(df)
  print(names)
  print(spearman_test2(df))
}
```

```
## [1] "stud_survey_format.TimeReadingMins" "stud_survey_format.TimeTV"
##
##  Pearson's product-moment correlation
##
## data:  df[, 1] and df[, 2]
## t = -5.6457, df = 9, p-value = 0.0003153
## alternative hypothesis: true correlation is not equal to 0
## 99 percent confidence interval:
##  -0.9801052 -0.4453124
## sample estimates:
##        cor
## -0.8830677
##
## [1] "stud_survey_format.TimeReadingMins" "stud_survey_format.Happiness"
##
##  Pearson's product-moment correlation
##
## data:  df[, 1] and df[, 2]
## t = -1.4488, df = 9, p-value = 0.1813
## alternative hypothesis: true correlation is not equal to 0
## 99 percent confidence interval:
##  -0.8801821  0.4176242
## sample estimates:
##        cor
## -0.4348663
##
## [1] "stud_survey_format.TimeReadingMins" "stud_survey_format.Gender"
##
##  Pearson's product-moment correlation
##
## data:  df[, 1] and df[, 2]
## t = -0.27001, df = 9, p-value = 0.7932
## alternative hypothesis: true correlation is not equal to 0
## 99 percent confidence interval:
##  -0.7618362  0.6755104
## sample estimates:
##         cor
## -0.08964215
##
```

```
## [1] "stud_survey_format.TimeTV"    "stud_survey_format.Happiness"
##
##  Pearson's product-moment correlation
##
## data:  df[, 1] and df[, 2]
## t = 2.4761, df = 9, p-value = 0.03521
## alternative hypothesis: true correlation is not equal to 0
## 99 percent confidence interval:
##  -0.1570212  0.9306275
## sample estimates:
##      cor
## 0.636556
##
## [1] "stud_survey_format.TimeTV" "stud_survey_format.Gender"
##
##  Pearson's product-moment correlation
##
## data:  df[, 1] and df[, 2]
## t = 0.01979, df = 9, p-value = 0.9846
## alternative hypothesis: true correlation is not equal to 0
## 99 percent confidence interval:
##  -0.7182866  0.7246128
## sample estimates:
##         cor
## 0.006596673
##
## [1] "stud_survey_format.Happiness" "stud_survey_format.Gender"
##
##  Pearson's product-moment correlation
##
## data:  df[, 1] and df[, 2]
## t = 0.47695, df = 9, p-value = 0.6448
## alternative hypothesis: true correlation is not equal to 0
## 99 percent confidence interval:
##  -0.6365617  0.7890897
## sample estimates:
##       cor
## 0.1570118
```

```
#4.4 Describe what the calculations in the correlation matrix suggest about the relationship between th

cor(stud_survey_format,method="spearman")
```

```
##                TimeReading      TimeTV  Happiness      Gender TimeReadingMins
## TimeReading      1.00000000 -0.90725363 -0.4065196 -0.08801408      1.00000000
## TimeTV          -0.90725363  1.00000000  0.5662159 -0.02899963     -0.90725363
## Happiness       -0.40651964  0.56621595  1.0000000  0.11547005     -0.40651964
## Gender          -0.08801408 -0.02899963  0.1154701  1.00000000     -0.08801408
## TimeReadingMins  1.00000000 -0.90725363 -0.4065196 -0.08801408      1.00000000
```

```
#The correlation matrix indicates a negative relationship between TimeReading and TimeTV. At the same,

# 5. Calculate the correlation coefficient and the coefficient of determination, describe what you conc
```

```r
cor_coeff <- cor(stud_survey_format$TimeReadingMins, stud_survey_format$TimeTV, method="spearman")
coeff_determine <- cor_coeff^2
print(paste0("Correlation Coefficient: ", cor_coeff))
```

```
## [1] "Correlation Coefficient: -0.907253627251832"
```

```r
print(paste0("Correlation Determin: ", coeff_determine))
```

```
## [1] "Correlation Determin: 0.823109144161606"
```

```r
#The correlation coefficient -.90 indicates strong negative correlation
#However, Correlation determination values of .82 indicates Time Reading shares 82% of variability in T

cor_coeff <- cor(stud_survey_format$TimeTV, stud_survey_format$Happiness, method="spearman")
coeff_determine <- cor_coeff^2
print(paste0("Correlation Coefficient: ", cor_coeff))
```

```
## [1] "Correlation Coefficient: 0.566215948571757"
```

```r
print(paste0("Correlation Determin: ", coeff_determine))
```

```
## [1] "Correlation Determin: 0.320600500417014"
```

```r
#The correlation coefficient .56 indicates strong postive correlation
#However, Correlation determination values of .82 indicates Time TV shares 32% of variability in Happin
```

6. Based on your analysis can you say that watching more TV caused students to read less? Explain.

We could confirm that Time spent on Reading is having strong negative correlation with time spent on watching TV. So, these 2 variables are having negative impact to each other

7. Pick three variables and perform a partial correlation, documenting which variable you are "controlling". Explain how this changes your interpretation and explanation of the results.

```r
library(ggplot2)
library(dplyr)

stud_survey <- read.csv("E:/Personal/Bellevue University/Course/github/dsc520/data/student-survey.csv")

pcor.test(stud_survey$TimeReading, stud_survey$TimeTV, stud_survey$Happiness, method="spearman")
```

```
##     estimate      p.value statistic  n gp   Method
## 1 -0.8990805 0.0004011345 -5.808771 11  1 spearman
```

```r
pcor.test(stud_survey$TimeTV, stud_survey$Happiness, stud_survey$TimeReading, method="spearman")
```

```
##    estimate   p.value statistic  n gp   Method
## 1 0.5137092 0.1288075  1.693531 11  1 spearman
```

```r
pcor.test(stud_survey$TimeReading, stud_survey$Happiness, stud_survey$TimeTV, method="spearman")
```

```
##    estimate   p.value statistic  n gp   Method
## 1 0.3091761 0.3847034 0.9195348 11  1 spearman
```

```r
#Inference: Comparing TimeReading and TimeTV, controlling for Happiness results in a high p-value, but
```