Advertising Budget Project

4/16/2022

library(regclass)

## Warning: package 'VGAM' was built under R version 4.1.2

## Warning: package 'randomForest' was built under R version 4.1.2

library(ggplot2)

Data source: <https://www.kaggle.com/datasets/yasserh/advertising-sales-dataset>

**1. Introduction: Description of the data that you selected:**

We first importing CSV and look at the data

d <- read.csv("data/Advertising Budget and Sales.csv")  
  
head(d,3)

## X TV.Ad.Budget.... Radio.Ad.Budget.... Newspaper.Ad.Budget.... Sales....  
## 1 1 230.1 37.8 69.2 22.1  
## 2 2 44.5 39.3 45.1 10.4  
## 3 3 17.2 45.9 69.3 9.3

Then, we look at the distribution of the variables

summary(d)

## X TV.Ad.Budget.... Radio.Ad.Budget.... Newspaper.Ad.Budget....  
## Min. : 1.00 Min. : 0.70 Min. : 0.000 Min. : 0.30   
## 1st Qu.: 50.75 1st Qu.: 74.38 1st Qu.: 9.975 1st Qu.: 12.75   
## Median :100.50 Median :149.75 Median :22.900 Median : 25.75   
## Mean :100.50 Mean :147.04 Mean :23.264 Mean : 30.55   
## 3rd Qu.:150.25 3rd Qu.:218.82 3rd Qu.:36.525 3rd Qu.: 45.10   
## Max. :200.00 Max. :296.40 Max. :49.600 Max. :114.00   
## Sales....   
## Min. : 1.60   
## 1st Qu.:10.38   
## Median :12.90   
## Mean :14.02   
## 3rd Qu.:17.40   
## Max. :27.00

dim(d)

## [1] 200 5

**Data Cleaning: Use R to remove all the missing values in the data set and use the complete data for the further steps.:**

To clean the data. Let’s look for for missing values.

# creating empty data.frame for missing values  
missing\_values = data.frame()  
# check for missing values  
for(i in 1:length(d)) {   
 missing\_values[i,1] = is.null(d[i])  
}  
rownames(missing\_values) <- c(names(d))  
colnames(missing\_values) = "missing values"  
  
missing\_values

## missing values  
## X FALSE  
## TV.Ad.Budget.... FALSE  
## Radio.Ad.Budget.... FALSE  
## Newspaper.Ad.Budget.... FALSE  
## Sales.... FALSE

Therefore, there are no missing values. Data is complete and ready for analysis.

Then, I would like to change the columns names

d$X = NULL  
names(d)[1] <- "TV.Ad.Budget"  
names(d)[2] <- "Radio.Ad.Budget"  
names(d)[3] <- "Newspaper.Ad.Budget"  
names(d)[4] <- "Sales"

**3. Association Analysis:**

We want to predict variable “Sales” and we set it as “y”.

names(d)[4] <- "y"  
head(d,2 )

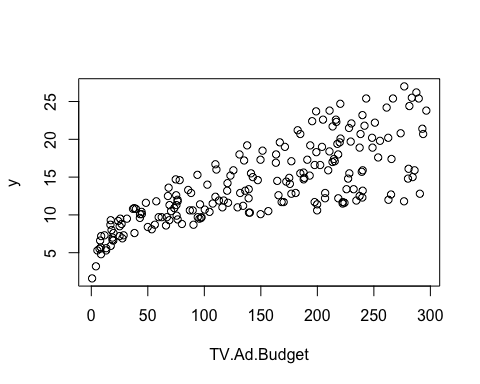
## TV.Ad.Budget Radio.Ad.Budget Newspaper.Ad.Budget y  
## 1 230.1 37.8 69.2 22.1  
## 2 44.5 39.3 45.1 10.4

all\_correlations(d, interest = "y", sorted = "magnitude")

## var1 var2 correlation pval  
## 1 TV.Ad.Budget y 0.7822244 1.467390e-42  
## 2 Radio.Ad.Budget y 0.5762226 4.354966e-19  
## 3 Newspaper.Ad.Budget y 0.2282990 1.148196e-03

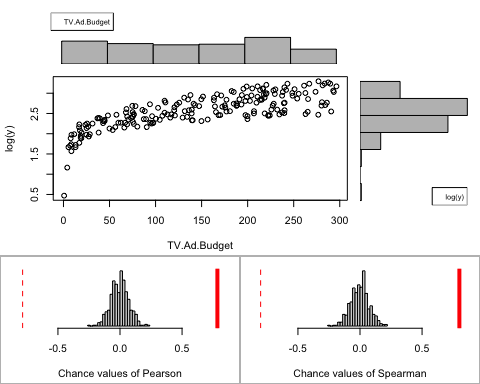
We can see that variable “TV.Ad.Budget” describe the variable “y” the best. Let’s examine association between them.

plot(y~TV.Ad.Budget,data=d)

 We can see that variance is increasing. We can see heteroscedasticity. Let’s try log transformation.

associate(log(y)~TV.Ad.Budget,data=d)

## Association between TV.Ad.Budget (numerical) and log(y) (numerical)  
## using 200 complete cases



## Permutation procedure:  
## Value Estimated p-value  
## Pearson's r 0.7846009 0  
## Spearman's rank correlation 0.8006144 0  
## With 500 permutations, we are 95% confident that:  
## the p-value of Pearson's correlation (r) is between 0 and 0.007   
## the p-value of Spearman's rank correlation is between 0 and 0.007   
## Note: If 0.05 is in this range, increase the permutations= argument.  
##   
##   
##   
## Advice: If stream of points is well described by an ellipse, use Pearson's r.  
## Otherwise, as long as stream is monotonic, use Spearman's rank correlation  
## or try logs, e.g. associate( log10(y)~log10(x) )

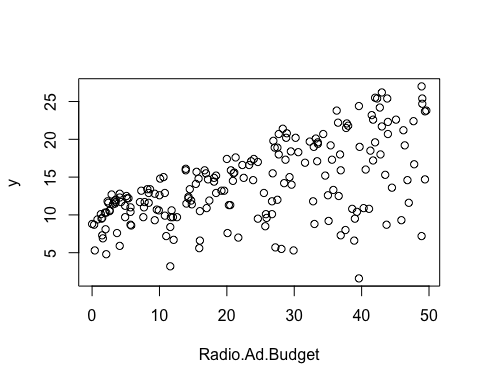
The relationships is not linear but monotonic. Therefore, we use Spearman correlation

r 0.8006144 p-value 0 and 0.007

p-value is less then 0.05 Therefore, log(y)~TV.Ad.Budget relationship statistically significant.

Let’s examine other associations.

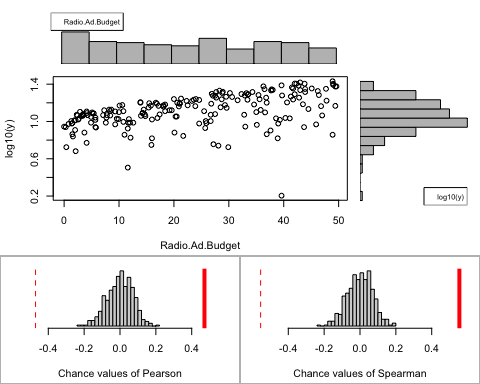
plot(y~Radio.Ad.Budget, data = d)



Let’s use log transformation again. Since th variance is increasing; we see heteroscedasticity.

associate(log10(y)~Radio.Ad.Budget,data=d)

## Association between Radio.Ad.Budget (numerical) and log10(y) (numerical)  
## using 200 complete cases



## Permutation procedure:  
## Value Estimated p-value  
## Pearson's r 0.4711507 0  
## Spearman's rank correlation 0.5543037 0  
## With 500 permutations, we are 95% confident that:  
## the p-value of Pearson's correlation (r) is between 0 and 0.007   
## the p-value of Spearman's rank correlation is between 0 and 0.007   
## Note: If 0.05 is in this range, increase the permutations= argument.  
##   
##   
##   
## Advice: If stream of points is well described by an ellipse, use Pearson's r.  
## Otherwise, as long as stream is monotonic, use Spearman's rank correlation  
## or try logs, e.g. associate( log10(y)~log10(x) )

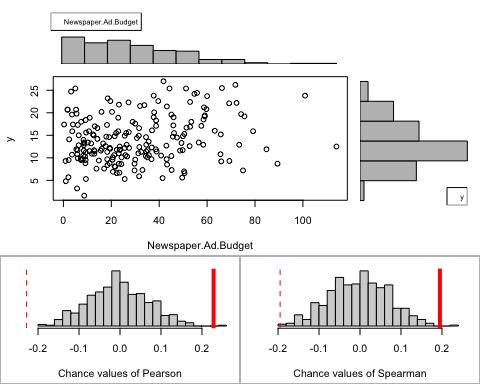
The relationship is linear. Let’s use Person.

Pearson’s r 0.4711507 p-value 0 and 0.007

p-value is less than 0.05. Therefore, log10(y)~Radio.Ad.Budget relationship statistically significant

associate(y~Newspaper.Ad.Budget,data=d)

## Association between Newspaper.Ad.Budget (numerical) and y (numerical)  
## using 200 complete cases



## Permutation procedure:  
## Value Estimated p-value  
## Pearson's r 0.2282990 0.002  
## Spearman's rank correlation 0.1949219 0.002  
## With 500 permutations, we are 95% confident that:  
## the p-value of Pearson's correlation (r) is between 0 and 0.011   
## the p-value of Spearman's rank correlation is between 0 and 0.011   
## Note: If 0.05 is in this range, increase the permutations= argument.  
##   
##   
##   
## Advice: If stream of points is well described by an ellipse, use Pearson's r.  
## Otherwise, as long as stream is monotonic, use Spearman's rank correlation  
## or try logs, e.g. associate( log10(y)~log10(x) )

The relationship looks like ellipse. Let’s use Person.

Pearson’s r 0.2282990 p-value 0 and 0.007

p-value is less than 0.05. Therefore, y~Newspaper relationship statistically significant

**4. Simple Linear Regression:**

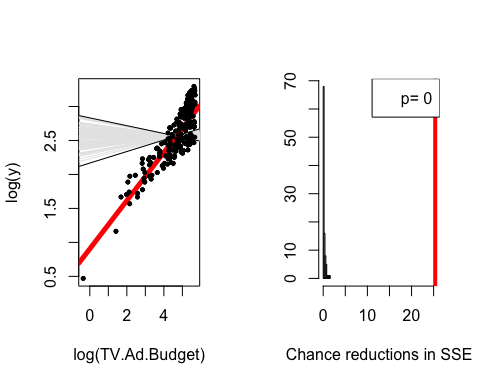
log(y)~log(TV.Ad.Budget)

Let’s use log transformation on both variables to build simple Linear Regression.

l12=lm(log(y)~log(TV.Ad.Budget) , data=d)  
summary(l12)

##   
## Call:  
## lm(formula = log(y) ~ log(TV.Ad.Budget), data = d)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.43349 -0.15917 0.01696 0.16910 0.39399   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.90525 0.07107 12.74 <2e-16 \*\*\*  
## log(TV.Ad.Budget) 0.35504 0.01487 23.87 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2109 on 198 degrees of freedom  
## Multiple R-squared: 0.7421, Adjusted R-squared: 0.7408   
## F-statistic: 569.8 on 1 and 198 DF, p-value: < 2.2e-16

possible\_regressions(l12)

 The slope of our regression is steep. The SSE reduction is much larger than what happens “by chance”. The regression is statistically significant.

confint(l12, level =0.05)

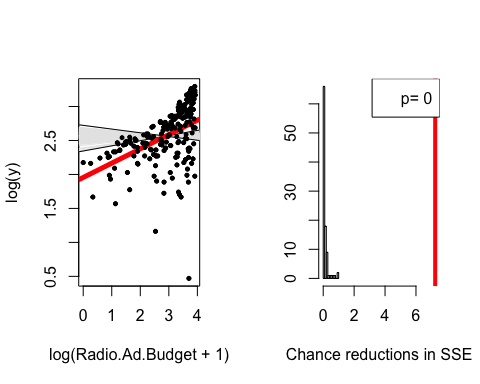
## 47.5 % 52.5 %  
## (Intercept) 0.9007833 0.9097076  
## log(TV.Ad.Budget) 0.3541019 0.3559695

Let’s do the same with log(y) ~ Radio.Ad.Budget+1. We have to add +1 to Radio.Ad.Budget because it has values of “0” what prevent R from creating the linear regression model.

l22=lm(log(y) ~ log(Radio.Ad.Budget+1),data=d)  
summary(l22)

##   
## Call:  
## lm(formula = log(y) ~ log(Radio.Ad.Budget + 1), data = d)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.2655 -0.1755 0.1126 0.2291 0.5169   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.95515 0.08739 22.372 < 2e-16 \*\*\*  
## log(Radio.Ad.Budget + 1) 0.21068 0.02886 7.299 6.87e-12 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3687 on 198 degrees of freedom  
## Multiple R-squared: 0.212, Adjusted R-squared: 0.208   
## F-statistic: 53.28 on 1 and 198 DF, p-value: 6.866e-12

possible\_regressions(l22)

 The slope of our regression is steep. The SSE reduction is much larger than what happens “by chance”. The regression is statistically significant.

confint(l22, level =0.05)

## 47.5 % 52.5 %  
## (Intercept) 1.9496652 1.9606393  
## log(Radio.Ad.Budget + 1) 0.2088682 0.2124927

Let’s compare the selected linear models

1. log(y)~log(TV.Ad.Budget) Residual standard error: 0.2109 Multiple R-squared: 0.7421, Adjusted R-squared: 0.7408
2. log(y)~TV.Ad.Budget Residual standard error: 0.2575 Multiple R-squared: 0.6156, Adjusted R-squared: 0.6137
3. log10(y) ~ log(Radio.Ad.Budget+1) Residual standard error: 0.3687 on 198 degrees of freedom Multiple R-squared: 0.212, Adjusted R-squared: 0.208 F-statistic: 53.28 on 1 and 198 DF, p-value: 6.866e-12

We also examened yNewspaperAd.Budget relationship. However, the association would not be in the limit of 10 pages. 4) y~Newspaper.Ad.Budget Residual standard error: 5.092 Multiple R-squared: 0.05212, Adjusted R-squared: 0.04733

Based on R square, and Residual standard error, the best linear regression that explain variable y (Sales) the best is log(y)~log(TV.Ad.Budget) (1).