Tutorial on Multiple Regression

Devarshi Pancholi

7/29/2019

R Markdown

Task: The purpose of this analysis/tutorial is to use multiple regression to accurately forecast cost to advertize in a magazine based on various attributes in our AdCost dataset.

The Steps we will take are:

1) Load the dataset in RStudio and have a look at the data.
2) Look at the scatterplot based on the dataset.
3) Check the correlation between variables.
4) Build the multiple regression model.
5) Plot the original data and the regression line.
6) Plot the standardized residuals vs. fitted values.

Part 1: Load the libraries and dataset in RStudio.

```
#loading the libraries we will use for this exercise
library(readr)
library(ggplot2)

#loading the dataset in RStudio
AdCost<-read_csv(file= "/Users/devarshipancholi/Desktop/AdCost.csv")
```

Now lets have a look at our dataset

```
#printing the data to the pdf file
print(AdCost)
```

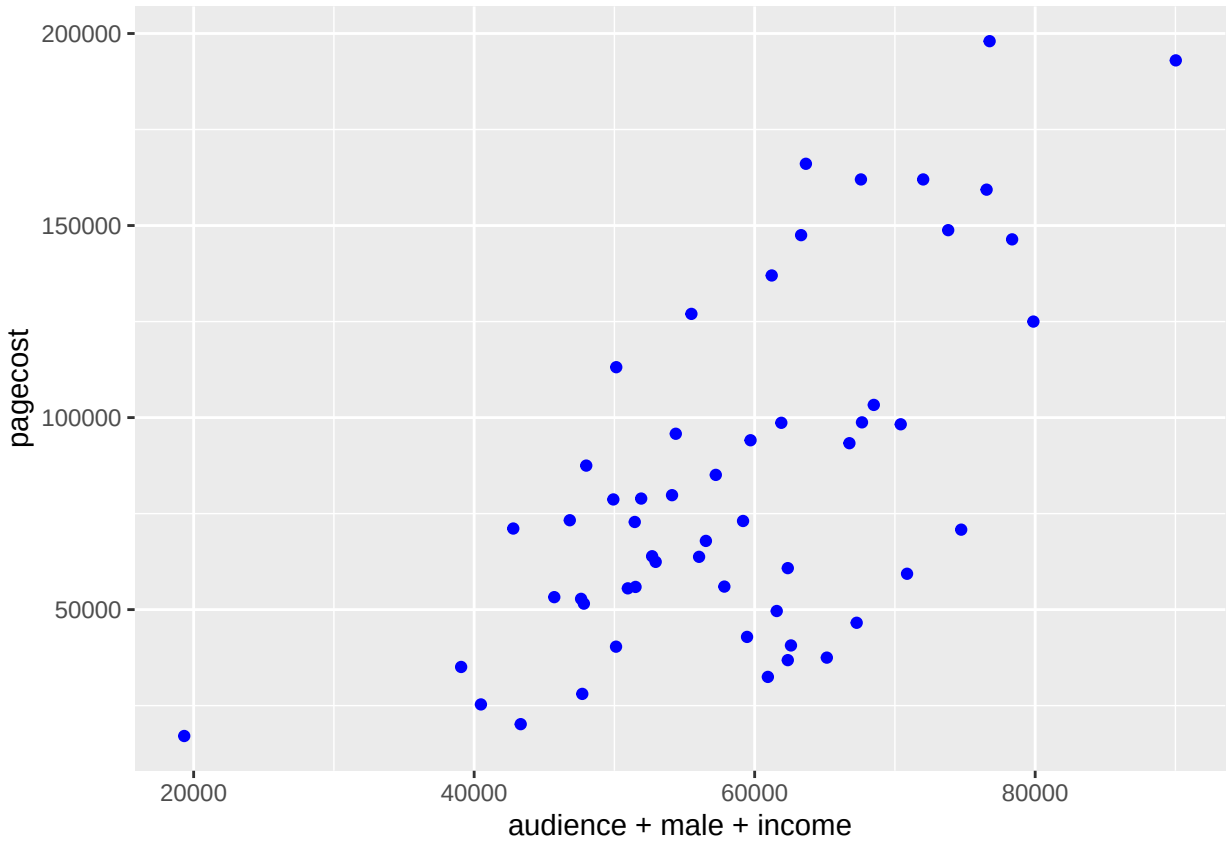
```
## # A tibble: 55 x 5
##    Magazine               pagecost audience  male income
##    <chr>                     <dbl>    <dbl> <dbl>  <dbl>
##  1 Audubon                   25315     1645  51.1  38787
##  2 Better Homes & Gardens   198000    34797  22.1  41933
##  3 Business Week            103300     4760  68.1  63667
##  4 Cosmopolitan              94100    15452  17.3  44237
##  5 Elle                      55540     3735  12.5  47211
##  6 Entrepreneur              40355     2476  60.4  47579
##  7 Esquire                   51559     3037  71.3  44715
##  8 Family Circle            147500    24539  13    38759
##  9 First For Women           28059     3856   3.6  43850
## 10 Forbes                    59340     4191  68.8  66606
## # ... with 45 more rows
```

Part 2: Look at the scatterplot.

To plot the regression line and scatterplot, I have used the library ggplot which we loaded in the beginning:

```
# command for loading the plot and assigning the axis + plotting the points on graph in blue
# y-axis denotes our dependent variable while x-axis denotes all independent variables combined

ggplot(data= AdCost, aes(x= audience+male+income, y= pagecost)) + geom_point(color= 'blue')
```



Part 3: Checking the co-relation

```
# removing the non numeric data from our analysis

all = AdCost[,2:5]

# 'all.obs' is used here as there are no missing data.
# 'pearson' method is used as our data is linear and normally distributed

cor(all, use="all.obs", method="pearson")
```

```
##             pagecost   audience        male     income
## pagecost  1.00000000  0.8722863 -0.08140151 -0.1666626
## audience  0.87228626  1.0000000 -0.13427523 -0.3531596
## male     -0.08140151 -0.1342752  1.00000000  0.5638074
## income   -0.16666264 -0.3531596  0.56380738  1.0000000
```

Part 4:Build the multiple regression model.

```
# multiple regression model stored in the variable mamed "linearRegModel"

MultipleRegModel<- lm(pagecost ~ audience+male+income, data= AdCost)
print(MultipleRegModel)
```

```
##
## Call:
## lm(formula = pagecost ~ audience + male + income, data = AdCost)
##
## Coefficients:
## (Intercept)     audience         male       income
##   4042.7986       3.7880    -123.6343       0.9026
```

Next, we get the summary for our regression model. This function allows us to observe a number of values like R-Squared, t-values and p-values for our independent varialbes.

```
#getting the summary
summary(MultipleRegModel)
```

```
##
## Call:
## lm(formula = pagecost ~ audience + male + income, data = AdCost)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -44049 -13491   -354  17116  44444
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4042.7986 16884.0391   0.239   0.8117
## audience        3.7880     0.2809  13.484   <2e-16 ***
## male         -123.6343   137.8485  -0.897   0.3740
## income          0.9026     0.3696   2.442   0.0181 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 21580 on 51 degrees of freedom
## Multiple R-squared:  0.7871, Adjusted R-squared:  0.7746
## F-statistic: 62.84 on 3 and 51 DF,  p-value: < 2.2e-16
```

Here's how you can obtain the anova output if needed:

```
anova(MultipleRegModel)
```

```
## Analysis of Variance Table
##
## Response: pagecost
##           Df     Sum Sq    Mean Sq  F value  Pr(>F)
## audience   1 8.4858e+10 8.4858e+10 182.2539 < 2e-16 ***
## male       1 1.4495e+08 1.4495e+08   0.3113 0.57931
```

```
## income     1 2.7769e+09 2.7769e+09   5.9642 0.01811 *
## Residuals 51 2.3746e+10 4.6560e+08
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

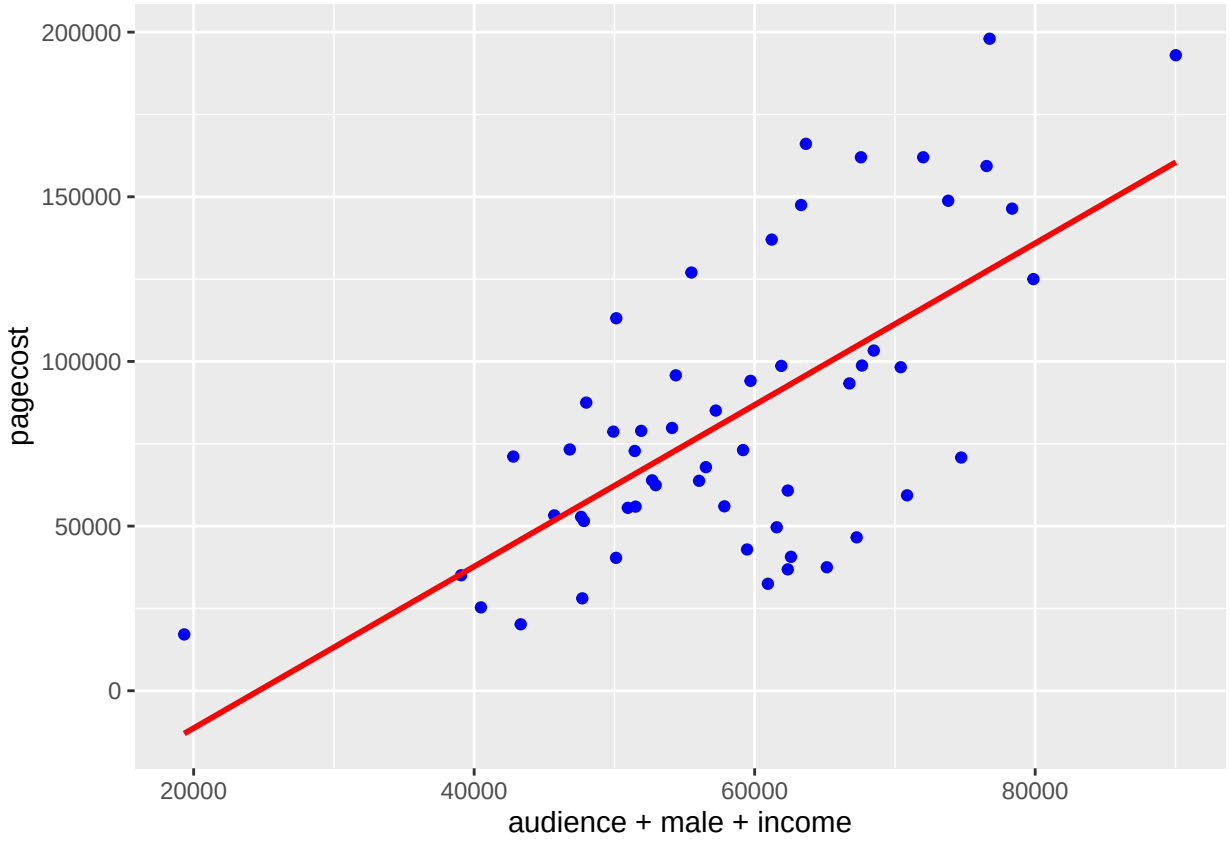
Part 5: Plot the original data and the linear regression line.

We will again use the library ggplot2 to add the regression line

```
# assigning the dataset
data(AdCost)

# deciding on X and Y axis + plotting the points on graph in blue
p1 = ggplot( data= AdCost,aes( x= audience+male+income,y= pagecost)) + geom_point( color= 'blue')

# plotting the regression line through the points
p1 + geom_smooth( method= 'lm', se= F, col= "red")
```



Part 6: Plot the standardized residuals vs. fitted values.

```
# obtaining standard residuals
MultipleRegModel.StdRes <- rstandard(MultipleRegModel)

# obtaining fitted values
MultipleRegModel.Fit <- fitted.values(MultipleRegModel)
```

```r
# deciding on X and Y axis + plotting the points on graph in blue
p3=ggplot(data=AdCost,aes(x=MultipleRegModel.Fit,y=MultipleRegModel.StdRes))+geom_point(color='blue')

# plotting the best fitting line through the points in red
p3 + geom_smooth( method= 'lm', se= F, col= "red")
```