Programming for Data Science with R

Part - II

Mohammad Wasiq

8/2/22

DSM - 1005

Programming for Data Science with R - II
DSM -1005

Table of Contents

I follow the book named **Statistical Inference via Data Science** *A ModernDive into R and the Tidyvers* by **Chester Ismay** and **Albert Y. Kim**

Teacher : **Prof. Athar Ali Khan Sir**

Writer : **Mohammad Wasiq** , $MS(Data \quad Science)$

1 R programing fo Data Science - II

In this Script we learn the R programing for Data Science at intermediate level . We learn the following Topics :

1. **Tidyverse**

- **Data Visualization** Using **ggplot2**
- **Data Wrangling** Using **dplyr**
- **Data Importing** & **Tidy Data**

2. **Data Modelling** with **moderndive**

- **Simple Regression**
- **Multiple Regression**

Note :- We have already discuss the 1st chapter in Part - I

2 Data Modelling with `moderndive`

3 Basic Regression / Linear Regression

- **Linear Models:** Summarising the data in the forms of equation is known as Linear Models.
- **Regression Analysis**: Regression Analysis is a simple method for ivestigation relationship among variables. **Linear regression** is one of the most commonly-used and easy-to-understand approaches to modeling. Linear regression involves a *numerical* outcome variable y and explanatory variables x that are either *numerical* or *categorical*.

3.1 Simple Linear Regression / SLR – Theory

Model :- $\boldsymbol{y = \beta_0 + \beta_1 x_1 + \epsilon}$

where , y is Response variable / Outcome of Study / Dependent Variable
β_0 is Intercept
β_1 is Slope i.e. $\frac{\Delta y}{\Delta x}$
x_1 is Explanatory variable / Predictor / Regressor / Independent Variable ϵ is Error

1. **Response Variable**(y)**:** A response variable measures an outcome of a study.

2. **Explanatory Variable**(x)**:** Explanatory variable explains or cause change in the response variable. Ex- Beer Drinking and Blood Alcohol Level. How does drinking beer affect the level of alcohol in our blood. Model: Blood Alcohol Level(y)=$\beta_0(intercept) + \beta_1(slope) * Beer - Drink(x) + \epsilon$

3. **Slope**(β_1)**:** $\beta_1 = \frac{\Delta y}{\Delta x}$ is slope, the amount by which y changes, when x changes by one unit. The slope is an important numerical description of the relationship b/w two variables.
   Ex- $Weight = \widehat{\beta_0} + \widehat{\beta_1} Age$ ⇒ Weight(kg) =3+0.2 Age(yrs) *Interpretation*- If age changes by one unit(i.e. 1 year) then weight changes by 0.2 kg.

4. **Intercept**(β_0)**:** β_0 is the intercept , the value of y when $x = 0$. Prediction: we can use a regression line to predict the response y for a specific value of the explanatory variable x.

5. **Residual:** Observed(y) - Predict(y) ⇒ $(y - \hat{y})$

6. **Assumption of Linear Model**
   * **Linear in Parameter:** The model (A) is linear in the parameters β_0 & β_1
   * **Random Sampling:** We have a random sample of n observation i.e. we draw samples from the population by simple random sampling method .
   * **Normality:** The error will follow normal distribution with $mean = 0$ & $variance = \sigma^2$ i.e. X~ $N(0, \sigma^2)$ *
   Homoscedasticity: The error has the same variance given any values of the explanatory variables. i.e. Variance is constant at every value x. ⇒ $V(e|x_1, x_2 \ldots . x_n) = \sigma^2$
   * **No Perfect Multicollinearity/No Auto Correlation:** In the Model(A), there is no perfect linear relationship b/w regression.(That's why we call x is independent variable) i.e. $Cov(e_i, e_j) = 0$

7. **Some Other Definition**:-
   * **Error:** Error of the dataset is the difference b/w the observed value and the unobserved value.
   * **Residuals:** Residual is calculated after running the regression model and is the difference b/w observed value and the estimated value.

$e_i = (y_i - \widehat{y_i}) = y_i - (\widehat{\beta_0} + \widehat{\beta_1}x)$

* **Sum of Squares:** Sum of squares is one of the most important output in regression analysis. The general rule is that a smaller sum of squares indicate a better model,as there is less variation in the data.

* **Coefficient of Determination /** $R^2 - Value$ It can be noted that a fitted model can be said to be good model when residuals are small for the measure of Goodness of Model, we use the following formula: $\boldsymbol{R^2} = \frac{\boldsymbol{SSR}}{\boldsymbol{SST}} = \mathbf{1} - \frac{\boldsymbol{SS_{res}}}{\boldsymbol{SST}}$, this is called, the coefficient of determination.

The ratio $\frac{\boldsymbol{SSR}}{\boldsymbol{SST}}$ describe the proportion of variability i.e. explained by the regression in relation to the total varialbility of y .

The ratio $\frac{\boldsymbol{SS_{res}}}{\boldsymbol{SST}}$ describe the proportion of variability that is not explained by the regression.

The value of R^2 lies $\mathbf{0} \leq \boldsymbol{R^2} \leq \mathbf{1}$.

$R^2 = 0$, indicates that poorest fit of the model. $R^2 = 1$, indicates that best fit of the model. $R^2 = 0.95$, indicates that 95% of the variation in y is explained by R^2. In simple words, the model is 95% good .

Drawbacks of R^2- As R^2 always increase with an increase in the no. of explanatory vaiables in the model.The main drawback of this property is that even when the irrelevant explanatory variables. are odded in the model, R^2 still increases. This indicates that the model is getting better, which is not really correct. With a purpose of correction in the overly optimstic picture ,Adjusted R^2, denoted by R^2_{adj} is used , which is defined as: $\boldsymbol{R^2_{adj}} = \mathbf{1} - \frac{\boldsymbol{SS_{res}/(n-k-1)}}{\boldsymbol{SST/(n-1)}}$ **OR**

$\boldsymbol{R^2_{adj}} = \mathbf{1} - \frac{\boldsymbol{SS_{res}}}{\boldsymbol{SST}} \times \frac{\boldsymbol{(n-1)}}{\boldsymbol{(n-k-1)}}$ **OR** $\boldsymbol{R^2_{adj}} = \mathbf{1} - \frac{\boldsymbol{n-1}}{\boldsymbol{n-k-1}}(\mathbf{1} - \boldsymbol{R^2})$

8. **Types of Sum of Square**:-
   (i).*Total Sum of Square(SST)*: $\boldsymbol{\sum_{i=1}^{n}(y_i - \bar{y})^2}$ where y_i=value in a sample and $\bar{y}$=mean value of the sample
   (ii). *Regression Sum of Square(SSR)*: $\boldsymbol{\sum_{i=1}^{n}(\widehat{y_i} - \bar{y})^2}$, where $\widehat{y_i}$=value estimated by regression line. and $\bar{y}$=Mean value of the sample. $SSR \propto \frac{1}{fitting-of-model}$
   (iii). *Residual Sum of Square(SSres)*: $SS_{res} = \sum_{i=1}^{n}(y_i - \widehat{y_i})^2$,where y_i=Observed Value and $\widehat{y_i}$=Estimated by regression line $SS_{res} \propto \frac{1}{Explanation-of-Data}$ $SST = SSR + SS_{res}$

9. Hypothesis of SLR:
   * **Null Hypothesis** $H_0: \beta_0 = \beta_{00}$ **VS** **Alternative Hypothesis** $H_1: \beta_0 \neq \beta_{00}$

Purpose of Data Modeling

- **Modeling for Explanation :** When you want to explicitly describe and quantify the relationship between the outcome variable y and a set of explanatory variables x, determine the significance of any relationships, have measures summarizing these relationships, and possibly identify any causal relationships between the variables.

- **Modeling for prediction:** When you want to predict an outcome variable y based on the information contained in a set of predictor variables x. Unlike modeling for explanation, however, you don't care so much about understanding how all the variables relate and interact with one another, but rather only whether you can make good predictions about y using the information in x.

Exp.- We are interested in an outcome variable y of whether patient develop *lung cancer* and information x on their risk factors , such as *smoking habits* , *age* and *socioeconomics* status. One reason could be that we want to design an intervention to reduce lung cancer incidence in a population, such as targeting smokers of a specific age group with advertising for smoking cessation programs. If we are modeling for prediction, however, we wouldn't care so much about understanding how all the individual risk factors contribute to lung cancer, but rather only whether we can make good predictions of which people will contract lung cancer.

3.2 One Numerical Explanatory Variable :

3.2.1 Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) is an approach to analyze the data using visual techniques. It is used to discover trends, patterns, or ti check assumptions with the help of statistical summary and graphical representations.

A crucial step before doing any kind of analysis or modeling is performing an exploratory data analysis, or EDA for short. EDA gives you a sense of the distributions of the individual variables in your data, whether any potential relationships exist between variables, whether there are outliers and/or missing values, and (most importantly) how to build your model. Here are three common steps in an EDA:

- Most crucially, looking at the raw data values.
- Computing summary statistics, such as means, medians, and interquartile ranges.
- Creating data visualizations.

About Data `evals` The data on the 463 courses at UT Austin can be found in the evals data frame included in the moderndive package.

- **ID:** An identification variable used to distinguish between the 1 through 463 courses in the dataset.
- **score:** A numerical variable of the course instructor's average teaching score, where the average is computed from the evaluation scores from all students in that course. Teaching scores of 1 are lowest and 5 are highest. This is the outcome variable y of interest.
- **bty_avg:** A numerical variable of the course instructor's average "beauty" score, where the average is computed from a separate panel of six students. "Beauty" scores of 1 are lowest and 10 are highest. This is the explanatory variable x of interest.

- Link : *https://www.openintro.org/stat/data/?data=evals*
- **age:** A numerical variable of the course instructor's age. This will be another explanatory variable x that we'll use in the Learning check at the end of this subsection.

I^{st} **step of EDA - Looking the Data** .

```
# Load the require library
library(tidyverse)
library(moderndive)
library(skimr)
library(gapminder)

# Load the evals data
data("evals")

# dimension of data
dim(evals)

## [1] 463  14

# Names of Columns of data evals
names(evals)

##  [1] "ID"           "prof_ID"      "score"
##  [4] "age"          "bty_avg"      "gender"
##  [7] "ethnicity"    "language"     "rank"
## [10] "pic_outfit"   "pic_color"    "cls_did_eval"
## [13] "cls_students" "cls_level"

# View and Head of Data
# View(evals)
head(evals)

## # A tibble: 6 x 14
##      ID prof_ID score   age bty_avg gender ethnicity language
##   <int>   <int> <dbl> <int>   <dbl> <fct>  <fct>     <fct>
## 1     1       1   4.7    36       5 female minority  english
## 2     2       1   4.1    36       5 female minority  english
## 3     3       1   3.9    36       5 female minority  english
## 4     4       1   4.8    36       5 female minority  english
## 5     5       2   4.6    59       3 male   not mino~ english
## 6     6       2   4.3    59       3 male   not mino~ english
## # ... with 6 more variables: rank <fct>, pic_outfit <fct>,
## #   pic_color <fct>, cls_did_eval <int>, cls_students <int>,
## #   cls_level <fct>
```

Task : Select the columns *ID* , *score* , *bty_avg* , *age* from the data *evals* and named *evals_ch5)*.

```
evals_ch5 <- evals %>%
  select(ID , score , bty_avg , age)

# View(evals_ch5)
glimpse(evals_ch5)
```

```
## Rows: 463
## Columns: 4
## $ ID      <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, ~
## $ score   <dbl> 4.7, 4.1, 3.9, 4.8, 4.6, 4.3, 2.8, 4.1, 3.4~
## $ bty_avg <dbl> 5.000, 5.000, 5.000, 5.000, 3.000, 3.000, 3~
## $ age     <int> 36, 36, 36, 36, 59, 59, 59, 51, 51, 40, 40,~
```

Sample of size 5 from evals_ch5

```
evals_ch5 %>%
  sample_n(size = 5)
```

```
## # A tibble: 5 x 4
##        ID score bty_avg   age
##     <int> <dbl>   <dbl> <int>
## 1   277   4.9    6.5     38
## 2    41   4.3    4       51
## 3   431   4.5    5.83    33
## 4    90   4.8    2.5     56
## 5   316   3.7    6       52
```

II^{nd} **Step of EDA - Statistical Summary**

Task : Find the *mean* and *median* of *bty_avg* and *score* variables of *evals_ch5* subdata .

```
evals_ch5 %>%
  summarise(
    mean_bty_avg = mean(bty_avg) ,
    median_bty_avg = median(bty_avg) ,
    mean_score = mean(score) ,
    median_score = median(score))
```

```
## # A tibble: 1 x 4
##   mean_bty_avg median_bty_avg mean_score median_score
##          <dbl>          <dbl>      <dbl>        <dbl>
## 1         4.42           4.33       4.17          4.3
```

In above summary we get only *mean* and *median* only , not all the summary . If we want to get all summary , then our code is very long and time consuming . Instead to write a long code we use **skim()** command of *skimr* package .

```
evals_ch5 %>%
  select(bty_avg , score) %>%
  skim()
```

Data summary

Name	Piped data
Number of rows	463
Number of columns	2
_______________________	
Column type frequency:	
numeric	2
________________________	
Group variables	None

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
bty_avg	0	1	4.42	1.53	1.67	3.17	4.33	5.5	8.17	▃▇▇▃▂
score	0	1	4.17	0.54	2.30	3.80	4.30	4.6	5.00	▁▁▅▇▇

Note : The *skim()* function only returns what are known as univariate summary statistics: functions that take a single variable and return some numerical summary of that variable .

A *correlation coefficient* is a quantitative expression of the *strength* of the linear relationship between two numerical variables. It lies between -1 to 1. Value closer to 0 means weak linearity and closer to -1 or 1 means strong linearity.

Task : Find correlation coefficient between score and bty_avg .

```
evals_ch5 %>%
  get_correlation(formula = score ~ bty_avg) # get_correlation is from
moderndive pkg.
```

```
## # A tibble: 1 x 1
##     cor
##   <dbl>
## 1 0.187
```

Another way to find correlation coeff.

```
round(evals_ch5 %>%
  summarise(correlation = cor(score , bty_avg)) , 2)
```

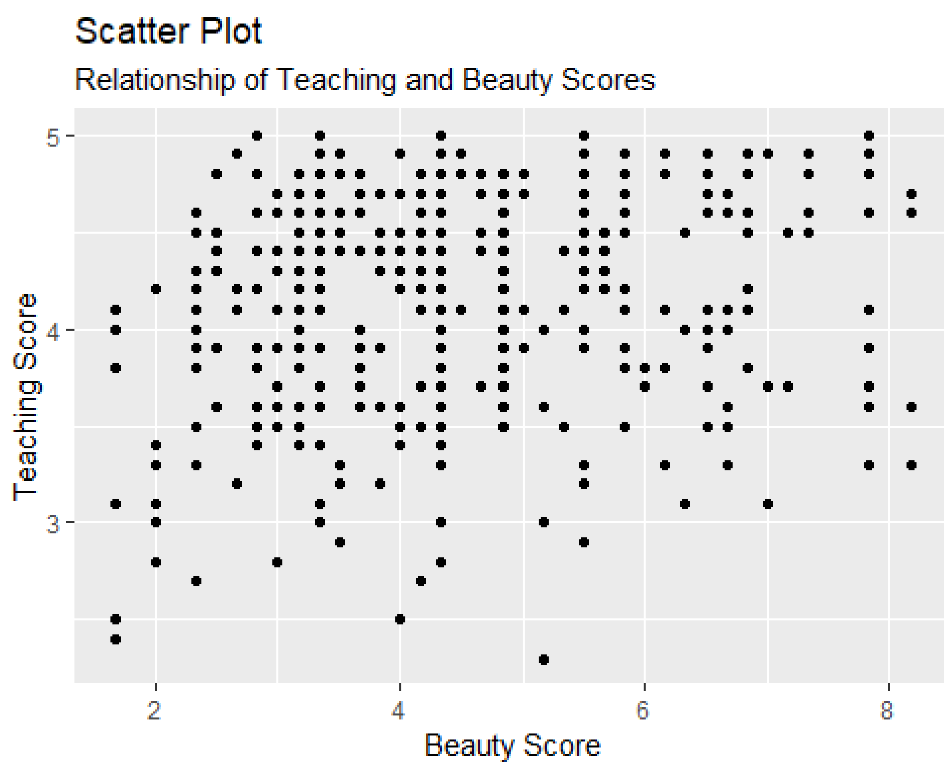
```
## # A tibble: 1 x 1
##   correlation
##         <dbl>
## 1        0.19
```

In our case, the correlation coefficient **0.187** indicates that the relationship between teaching evaluation score and "beauty" average is *"weakly positive"* .

III^{rd} **Step of EDA - Graphically Presentation**

Task : Make a *scatter plot*

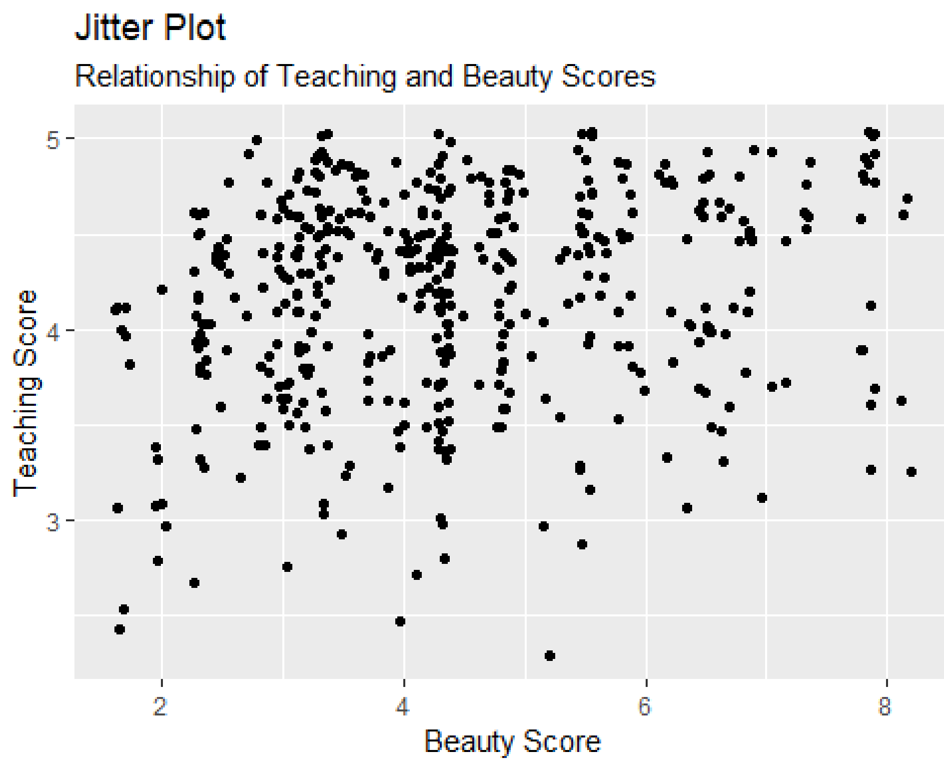
```
ggplot(evals_ch5, aes(bty_avg , score)) +
  geom_point() +
  labs(x = "Beauty Score" , y = "Teaching Score" , title = "Scatter Plot" ,
subtitle = "Relationship of Teaching and Beauty Scores")
```



The relationship between "Teaching Score" and "Beauty Score" is "weakly positive." This is consistent with our earlier computed correlation coefficient of **0.187** . This plot suffers from overplotting.

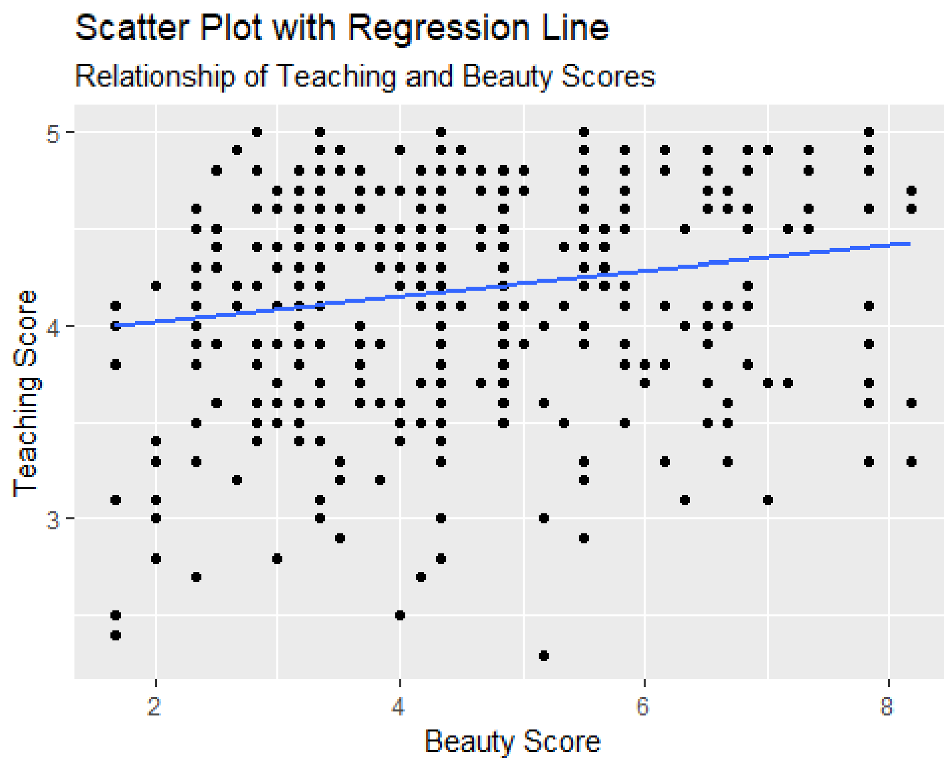
Task : To avoid *Overplotting* , make a **Jitter plot** .

```
ggplot(evals_ch5, aes(bty_avg , score)) +
  geom_jitter() +
  labs(x = "Beauty Score" , y = "Teaching Score" , title = "Jitter Plot" ,
subtitle = "Relationship of Teaching and Beauty Scores")
```



Task : Make a **Scatter plot** with *Regression line* .

```
ggplot(evals_ch5, aes(bty_avg , score)) +
  geom_point() + geom_smooth(method = "lm" , se = F) +
  labs(x = "Beauty Score" , y = "Teaching Score" , title = "Scatter Plot with
Regression Line" , subtitle = "Relationship of Teaching and Beauty Scores")

## `geom_smooth()` using formula 'y ~ x'
```



The *regression line* is a visual summary of the relationship between two numerical variables. A regression line is *"best-fitting"* in that it minimizes some mathematical criteria.

3.2.2 Simple Linear Regression

Our Model : $score = \beta_0 + \beta_1(bty\_avg) + \epsilon$

- We first "fit" the linear regression model using the **lm()** function and save it in score_model.
- We get the regression table by applying the **get_regression_table()** function from the *moderndive* package to score_model.

```
# Fit Regression Model :
score_model <- lm(score ~ bty_avg , data = evals_ch5)

# Regression Table :
get_regression_table(score_model)

## # A tibble: 2 x 7
##   term      estimate std_error statistic p_value lower_ci
##   <chr>        <dbl>     <dbl>     <dbl>   <dbl>    <dbl>
## 1 intercept    3.88      0.076     51.0        0    3.73
## 2 bty_avg      0.067     0.016      4.09       0    0.035
## # ... with 1 more variable: upper_ci <dbl>
```

Estimated Model: $\widehat{score} = \beta_0 + \beta_{bty\_avg}(bty\_avg)$ **Fitted Model :** $\widehat{score} = 3.88 + 0.067(bty\_avg)$ **Interpretation :** For every increase of 1 unit in bty_avg, there is an associated increase of, on average, **0.067** units of score .

To get the Summary of our Model

```
# Summary of Regression Model :
get_regression_summaries(score_model)

## # A tibble: 1 x 9
##   r_squared adj_r_squared   mse  rmse sigma statistic p_value
##       <dbl>         <dbl> <dbl> <dbl> <dbl>     <dbl>   <dbl>
## 1     0.035         0.033 0.285 0.534 0.535      16.7       0
## # ... with 2 more variables: df <dbl>, nobs <dbl>
```

The value of $R^2 = 0.035$ that means only **3.5%** of variability is explained . **OR** Ou Model Is only **3.5%** Good .

3.2.3 Observed / Fitted Values and Residuals

Residuals: Residual is calculated after running the regression model and is the difference b/w observed value and the estimated value. $e_i = (y_i - \widehat{y_i}) = y_i - \left(\widehat{\beta_0} + \widehat{\beta_1}x\right)$

We get the residuals using **get_regression_points(model)**

```
residual <- get_regression_points(score_model)

residual[c(1:5,24,200) ,]

## # A tibble: 7 x 5
##      ID score bty_avg score_hat residual
##   <int> <dbl>   <dbl>     <dbl>    <dbl>
## 1     1   4.7    5         4.21    0.486
## 2     2   4.1    5         4.21   -0.114
## 3     3   3.9    5         4.21   -0.314
## 4     4   4.8    5         4.21    0.586
## 5     5   4.6    3         4.08    0.52
## 6    24   4.4    5.5       4.25    0.153
## 7   200   4      2.33      4.04   -0.036
```

- The *score* column represents the observed outcome variable y. This is the y-position of the 463 black points.
- The *bty_avg* column represents the values of the explanatory variable x. This is the x-position of the 463 black points.
- The *score_hat* column represents the fitted values $\hat{y}$. This is the corresponding value on the regression line for the $463xx$ values.
- The *residual* column represents the residuals $y - \hat{y}$. This is the 463 vertical distances between the 463 black points and the regression line.

Now we talk about 24th value .

- **score = 4.4** is the observed teaching score y for this course's instructor.
- **bty_avg = 5.50** is the value of the explanatory variable bty_avg x for this course's instructor.
- **score_hat = 4.25 = 3.88 + 0.067 ⯑ 5.5** is the fitted valu e$\hat{y}$ on the regression line for this course's instructor.
- **residual = 0.153 = 4.4 - 4.25** is the value of the residual for this instructor. In other words, the model's fitted value was off by 0.153 teaching score units for this course's instructor.

3.2.4 EDA with Age & Score

Task LC(5.1 : 5:3) -: Conduct a new *Exploratory Data Analysis* with the same outcome variable y being **score** but with **age** as the new explanatory variable .

EDA I - Looking the Data

```
# Load the require library
library(tidyverse)
library(moderndive)
library(skimr)
library(gapminder)

# Load the evals data
data("evals")
```

Select the Require Data

```
ev_age <- evals %>%
  select(ID , score , age)
# View(ev_age)

glimpse(ev_age)

## Rows: 463
## Columns: 3
## $ ID    <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14~
## $ score <dbl> 4.7, 4.1, 3.9, 4.8, 4.6, 4.3, 2.8, 4.1, 3.4, ~
## $ age   <int> 36, 36, 36, 36, 59, 59, 59, 51, 51, 40, 40, 4~
```

Sample of size 5 from ev_age

```
ev_age %>%
  sample_n(size = 5)

## # A tibble: 5 x 3
##       ID score   age
##    <int> <dbl> <int>
## 1   116   3.4    57
## 2    94   4      48
## 3   262   4.3    52
```

```
## 4   235   4.6    61
## 5   456   4.5    32
```

EDA II - Statistical Summary

```
ev_age %>%
  select(score , age) %>%
  skim()
```

Data summary

Name	Piped data
Number of rows	463
Number of columns	2
_______________________	
Column type frequency:	
numeric	2
________________________	
Group variables	None

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
score	0	1	4.17	0.54	2.3	3.8	4.3	4.6	5	▁▁▅▇▆
age	0	1	48.37	9.80	29.00	42.00	48.00	57.00	73	▆▆▇▆▁

Correlation Coefficient

```
round(ev_age %>%
  get_correlation(formula = score ~ age) , 2)

## # A tibble: 1 x 1
##     cor
##   <dbl>
## 1 -0.11
```

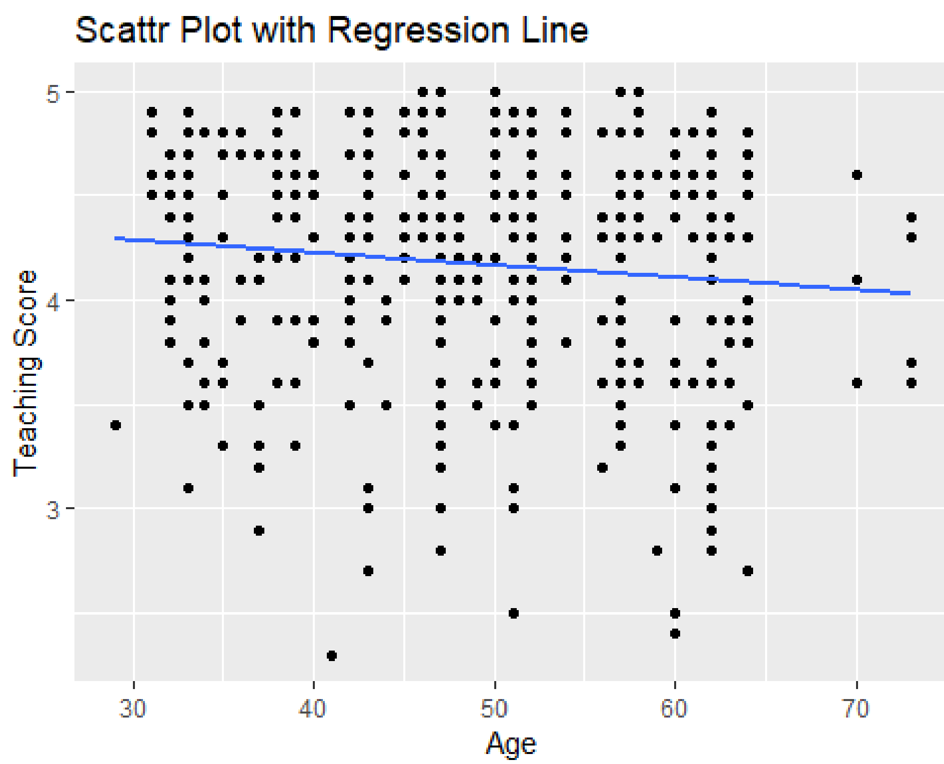
Another way to find correlation coeff.

```
round(ev_age %>%
  summarise(Correlation = cor(score , age)) , 2)

## # A tibble: 1 x 1
##   Correlation
##         <dbl>
## 1       -0.11
```

In our case , the correlation coefficient **-0.11** indicates that the relationship b/w *Score* and *Age* is *weakly negative* .

EDA III - Graphically Presentation

```
ggplot(ev_age , aes(age , score)) +
  geom_point() +
  geom_smooth(method = "lm" , se = F) +
  labs(x = "Age" , y = "Teaching Score" , title = "Scattr Plot with
Regression Line")

## `geom_smooth()` using formula 'y ~ x'
```



The relationship between "Teaching Score" and "Age" is "weakly positive." This is consistent with our earlier computed correlation coefficient of **0.187** .

Simple Linear Regression

Our Model : $Score = \beta_0 + \beta_{Age}(Age) + \epsilon$

```
# Fit Regression Model :
age_model <- lm(score ~ age , data = ev_age)

# Regression Table :
get_regression_table(age_model)

## # A tibble: 2 x 7
##   term      estimate std_error statistic p_value lower_ci
```

```
##   <chr>         <dbl>     <dbl>     <dbl>   <dbl>    <dbl>
## 1 intercept    4.46      0.127     35.2    0        4.21
## 2 age         -0.006     0.003     -2.31   0.021   -0.011
## # ... with 1 more variable: upper_ci <dbl>
```

Estimated Model : $\widehat{score} = \beta_0 + \beta_{Age}(Age)$

Fitted Model : $\widehat{Score} = 4.46 - 0.006(Age)$

Interpretation : For every increase of **1** unit in *Age*, there is an associated decrease of, on average, **0.006** units of score .

Summary of Our Model

```
get_regression_summaries(age_model)

## # A tibble: 1 x 9
##   r_squared adj_r_squared   mse  rmse sigma statistic p_value
##       <dbl>         <dbl> <dbl> <dbl> <dbl>     <dbl>   <dbl>
## 1     0.011         0.009 0.292 0.540 0.541      5.34   0.021
## # ... with 2 more variables: df <dbl>, nobs <dbl>
```

The value of $R^2 = 0.01$, that means only **1%** of variability is explained . **OR** Our Model is only **1%** Good.

Observed / Fitted Values & Residuals

```
residual <- get_regression_points(age_model)

residual[c(1:4 , 25, 220) , ]

## # A tibble: 6 x 5
##      ID score   age score_hat residual
##   <int> <dbl> <int>     <dbl>    <dbl>
## 1     1   4.7    36      4.25    0.452
## 2     2   4.1    36      4.25   -0.148
## 3     3   3.9    36      4.25   -0.348
## 4     4   4.8    36      4.25    0.552
## 5    25   4.6    62      4.09    0.506
## 6   220   4.9    42      4.21    0.687
```

3.3 One Categorical Explanatory Variable

About Gapminder Data The data on the 142 countries can be found in the gapminder data frame included in the gapminder package. 1. A numerical outcome variable y (a country's life expectancy) 2. A single categorical explanatory variable x (the continent that the country is a part of).

3.3.1 Exploratory Data Analysis

Task : Filter *countary , lifeExp , Continent , gdpPercap* of year *2007* from *gapmider* data.

```
library(tidyverse)
library(gapminder)
data("gapminder")
dim(gapminder)

## [1] 1704    6

names(gapminder)

## [1] "country"   "continent" "year"      "lifeExp"
## [5] "pop"       "gdpPercap"

gap7 <- gapminder %>%
  filter(year == 2007) %>%
  select(country , lifeExp , continent , gdpPercap)

# View(gap7)
glimpse(gap7)

## Rows: 142
## Columns: 4
## $ country   <fct> "Afghanistan", "Albania", "Algeria", "Ang~
## $ lifeExp   <dbl> 43.828, 76.423, 72.301, 42.731, 75.320, 8~
## $ continent <fct> Asia, Europe, Africa, Africa, Americas, O~
## $ gdpPercap <dbl> 974.5803, 5937.0295, 6223.3675, 4797.2313~
```

Sample of size 5 from gap7

```
gap7 %>%
  sample_n(size = 5)

## # A tibble: 5 x 4
##   country      lifeExp continent gdpPercap
##   <fct>          <dbl> <fct>         <dbl>
## 1 Morocco         71.2 Africa        3820.
## 2 Zimbabwe        43.5 Africa         470.
## 3 Egypt           71.3 Africa        5581.
## 4 South Africa    49.3 Africa        9270.
## 5 Greece          79.5 Europe       27538.
```

Task : *Get the summary of lifeExp and continent*

Summary of gap7

```
gap7 %>%
  select(lifeExp , continent) %>%
  skim()
```

Data summary

Name	Piped data
Number of rows	142

Number of columns	2
______________________	
Column type frequency:	
factor	1
numeric	1
________________________	
Group variables	None

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
continent	0	1	FALSE	5	Afr: 52, Asi: 33, Eur: 30, Ame: 25

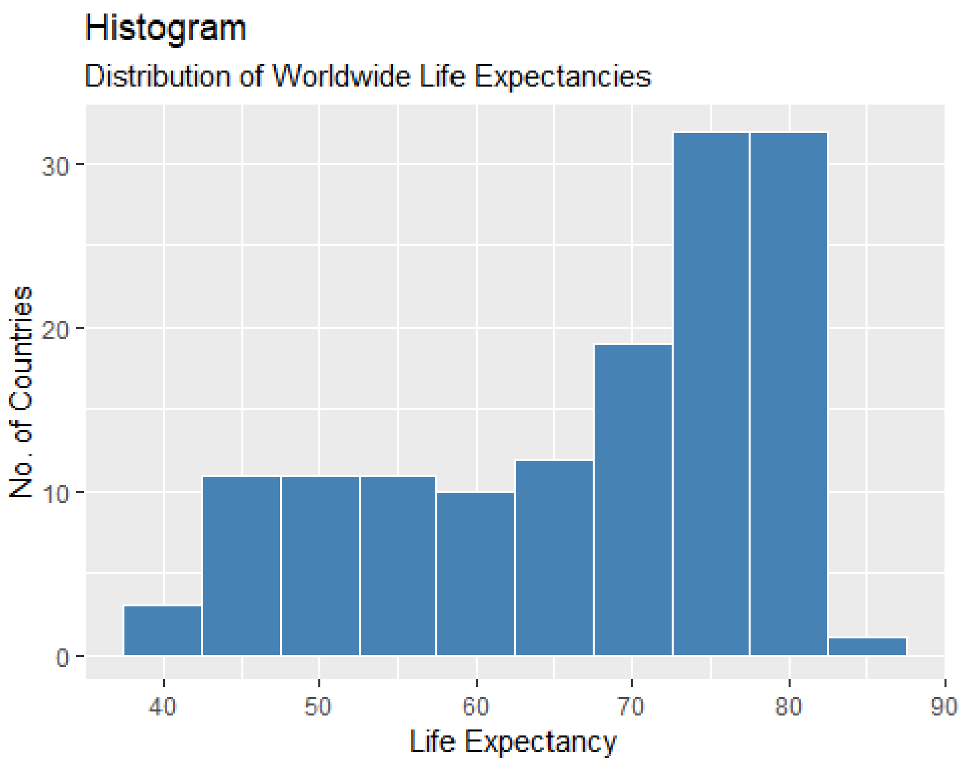
Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
lifeExp	0	1	67.01	12.07	39.61	57.16	71.94	76.41	82.6	▁▂▂▇▇

Graphically Presentation of gap7

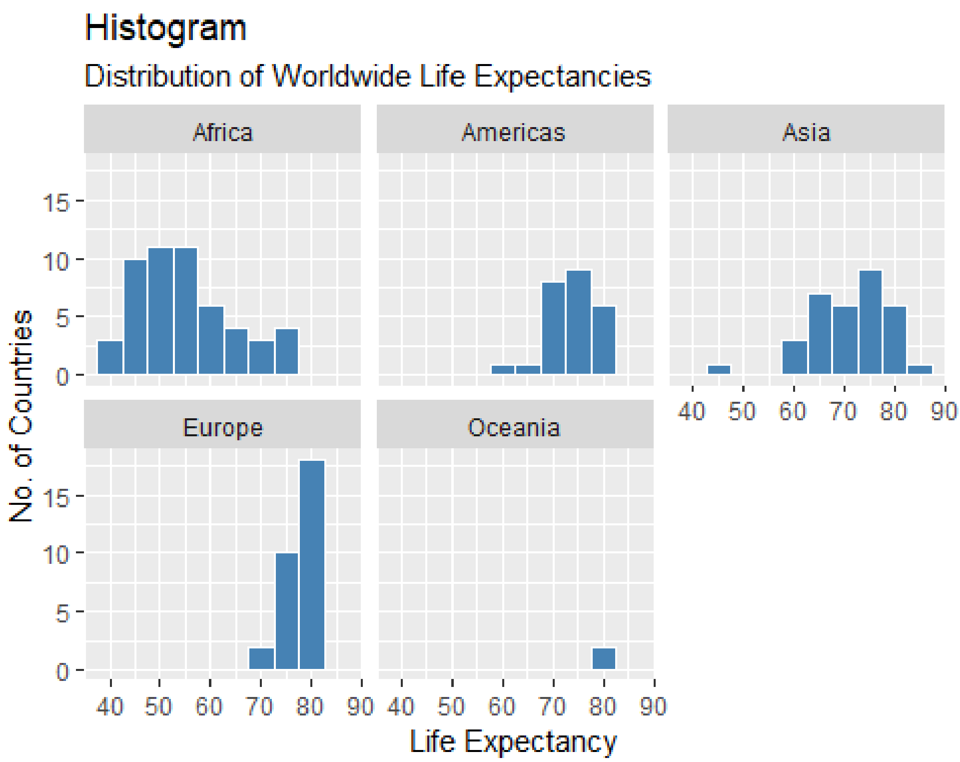
Task : Make a **Histogram**

```
ggplot(gap7 , aes(x = lifeExp)) +
  geom_histogram(binwidth = 5 , col = "white" , fill = "steelblue") +
  labs(x = "Life Expectancy" , y = "No. of Countries" , title = "Histogram" ,
subtitle = "Distribution of Worldwide Life Expectancies")
```



We see that this data is left-skewed, also known as negatively skewed: there are a few countries with low life expectancy that are bringing down the mean life expectancy. However, the median is less sensitive to the effects of such outliers; hence, the median is greater than the mean in this case. i.e. $M_e < M_d$

```
ggplot(gap7 , aes(x = lifeExp)) +
  geom_histogram(binwidth = 5 , col = "white" , fill = "steelblue") +
  facet_wrap(~ continent , nrow = 2) +
  labs(x = "Life Expectancy" , y = "No. of Countries" , title = "Histogram" ,
subtitle = "Distribution of Worldwide Life Expectancies")
```

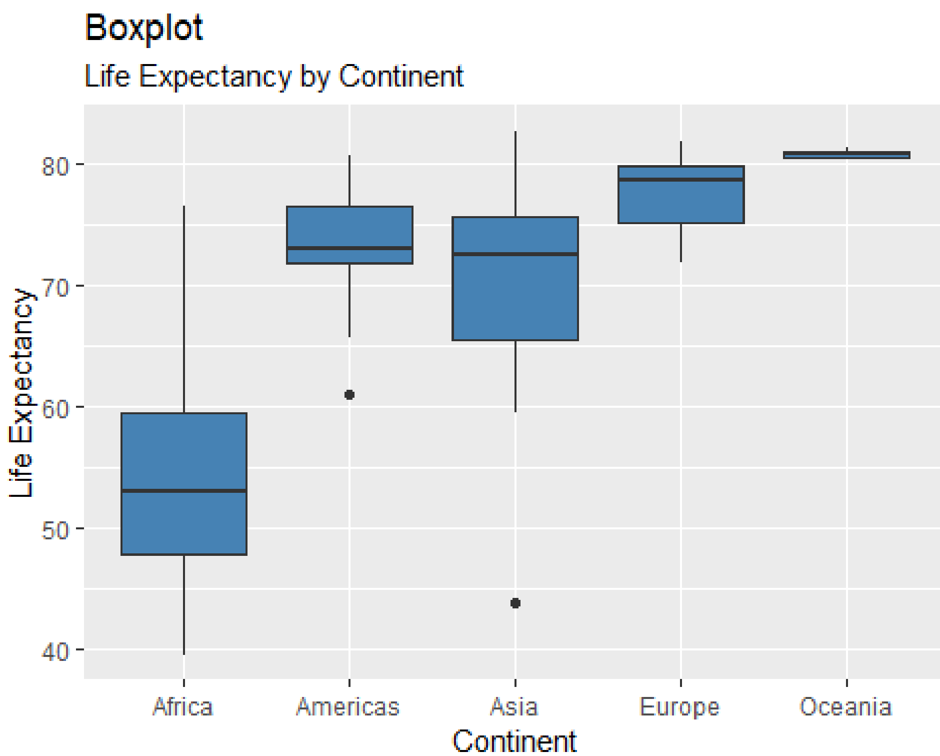


Observe that unfortunately the distribution of African life expectancies is much lower than the other continents, while in Europe life expectancies tend to be higher and furthermore do not vary as much. On the other hand, both Asia and Africa have the most variation in life expectancies. There is the least variation in Oceania, but keep in mind that there are only two countries in Oceania: Australia and New Zealand.

Some people prefer comparing the distributions of a numerical variable between different levels of a categorical variable using a boxplot instead of a faceted histogram. This is because we can make quick comparisons between the categorical variable's levels with imaginary horizontal lines.

Task : Make **Boxplot**

```
ggplot(gap7 , aes(x = continent , y = lifeExp)) +
  geom_boxplot(fill = "steelblue") +
  labs(x = "Continent" , y = "Life Expectancy" , title = "Boxplot" , subtitle
= "Life Expectancy by Continent")
```



Task : Compute the median and mean life expectancy for each continent .

```
gap7 %>%
  group_by(continent) %>%
  summarise(
    Median = median(lifeExp) ,
    Mean = mean(lifeExp)
            )

## # A tibble: 5 x 3
##   continent Median  Mean
##   <fct>      <dbl> <dbl>
## 1 Africa      52.9  54.8
## 2 Americas    72.9  73.6
## 3 Asia        72.4  70.7
## 4 Europe      78.6  77.6
## 5 Oceania     80.7  80.7
```

3.3.2 Simple Linear Regression

Our Model : $Life\_Exp. = \beta_0 + \beta_{cont.}(Continent) + \epsilon$

```
# Fit the Model :
le_model <- lm(lifeExp ~ continent , data = gap7)

# Regression Table :
get_regression_table(le_model)
```

```
## # A tibble: 5 x 7
##   term           estimate std_error statistic p_value lower_ci
##   <chr>             <dbl>     <dbl>     <dbl>   <dbl>    <dbl>
## 1 intercept          54.8      1.02     53.4        0     52.8
## 2 continent: A~      18.8      1.8      10.4        0     15.2
## 3 continent: A~      15.9      1.65      9.68       0     12.7
## 4 continent: E~      22.8      1.70     13.5        0     19.5
## 5 continent: O~      25.9      5.33      4.86       0     15.4
## # ... with 1 more variable: upper_ci <dbl>
```

Fitted Model :

$$\widehat{Life\_Exp.} = 54.8 + 18.8(Americas) + 15.9(Asia) + 22.8(Europe) + 25.9(Oceania)$$

Interpretation :

- For every increase of 1 unit in *Americas* there is an associated increase of, on average, **18.8** units of *Life Expectation* .
- For every increase of 1 unit in *Asia* there is an associated increase of, on average, **15.9** units of *Life Expectation* .
- For every increase of 1 unit in *Europe* there is an associated increase of, on average, **22.8** units of *Life Expectation* .
- For every increase of 1 unit in *Oceania* there is an associated increase of, on average, **25.9** units of *Life Expectation* .

3.3.3 Observed / Fitted Values & Residuals

```
rp <- get_regression_points(le_model, ID = "country")

View(rp)
```

LC(5.6) : Iidentify the five countries with the five smallest (most negative) residuals .

```
rp %>%
  top_n(n = -5 , wt = residual)

## # A tibble: 5 x 5
##   country     lifeExp continent lifeExp_hat residual
##   <fct>         <dbl> <fct>           <dbl>    <dbl>
## 1 Afghanistan    43.8 Asia             70.7    -26.9
## 2 Haiti          60.9 Americas         73.6    -12.7
## 3 Mozambique     42.1 Africa           54.8    -12.7
## 4 Swaziland      39.6 Africa           54.8    -15.2
## 5 Zambia         42.4 Africa           54.8    -12.4

# Arrange in Ascending Order because above command arrange by country

rp %>%
```

```
  top_n(n = -5 , wt = residual)%>%
  arrange(residual)

## # A tibble: 5 x 5
##   country     lifeExp continent lifeExp_hat residual
##   <fct>         <dbl> <fct>           <dbl>    <dbl>
## 1 Afghanistan    43.8 Asia             70.7    -26.9
## 2 Swaziland      39.6 Africa           54.8    -15.2
## 3 Mozambique     42.1 Africa           54.8    -12.7
## 4 Haiti          60.9 Americas         73.6    -12.7
## 5 Zambia         42.4 Africa           54.8    -12.4
```

The residual for Afghanistan is −26.900 and it is the smallest residual. This means that the average life expectancy of Afghanistan is 26.900 years lower than the average life expectancy of its continent, Asia.

LC(5.7) : Identify the five countries with the five largest (most posiitve) residuals .

```
rp %>%
  top_n(n = 5 , wt = residual)

## # A tibble: 5 x 5
##   country   lifeExp continent lifeExp_hat residual
##   <fct>       <dbl> <fct>           <dbl>    <dbl>
## 1 Algeria      72.3 Africa           54.8     17.5
## 2 Libya        74.0 Africa           54.8     19.1
## 3 Mauritius    72.8 Africa           54.8     18.0
## 4 Reunion      76.4 Africa           54.8     21.6
## 5 Tunisia      73.9 Africa           54.8     19.1

# Arrange Residuals in Descending Order because above command arrange by
country

rp %>%
  top_n(n = 5 , wt = residual) %>%
  arrange(desc(residual))

## # A tibble: 5 x 5
##   country   lifeExp continent lifeExp_hat residual
##   <fct>       <dbl> <fct>           <dbl>    <dbl>
## 1 Reunion      76.4 Africa           54.8     21.6
## 2 Libya        74.0 Africa           54.8     19.1
## 3 Tunisia      73.9 Africa           54.8     19.1
## 4 Mauritius    72.8 Africa           54.8     18.0
## 5 Algeria      72.3 Africa           54.8     17.5
```

The residual for Reunion is 21.636 and it is the largest residual. This means that the average life expectancy of Reunion is 21.636 years higher than the average life expectancy of its continent, Africa.

3.3.4 EDA with Continent & GDP

Conduct exploratory data analysis as done above with *x=continent, y=gdpPercap* from the same dataset **gapmider**.

EDA I - Select the Data

Task : Filter countary , lifeExp , Continent , gdpPercap of year 2007 from gapmider data

```
# Load require libraries
library(tidyverse)
library(moderndive)
library(skimr)
library(gapminder)

# Load the require dataset "gapminder"
data("gapminder")

# Filtering and Selecting the Data
gap7 <- gapminder %>%
  filter(year == 2007) %>%
  select(country , continent , gdpPercap)

# Sample of size n from above Data
gap7 %>%
  sample_n(size = 5)

## # A tibble: 5 x 3
##   country                continent gdpPercap
##   <fct>                  <fct>         <dbl>
## 1 Czech Republic         Europe       22833.
## 2 Angola                 Africa        4797.
## 3 Bosnia and Herzegovina Europe        7446.
## 4 Kenya                  Africa        1463.
## 5 Tanzania               Africa        1107.
```

EDA II - Summary of Data

Task : Find the summary of gdpPerCap

```
gap7 %>%
  select(continent ,gdpPercap) %>%
  skim()
```

Data summary

Name	Piped data
Number of rows	142
Number of columns	2
_______________________	

Column type frequency:	
factor	1
numeric	1
________________________	
Group variables	None

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
continent	0	1	FALSE	5	Afr: 52, Asi: 33, Eur: 30, Ame: 25

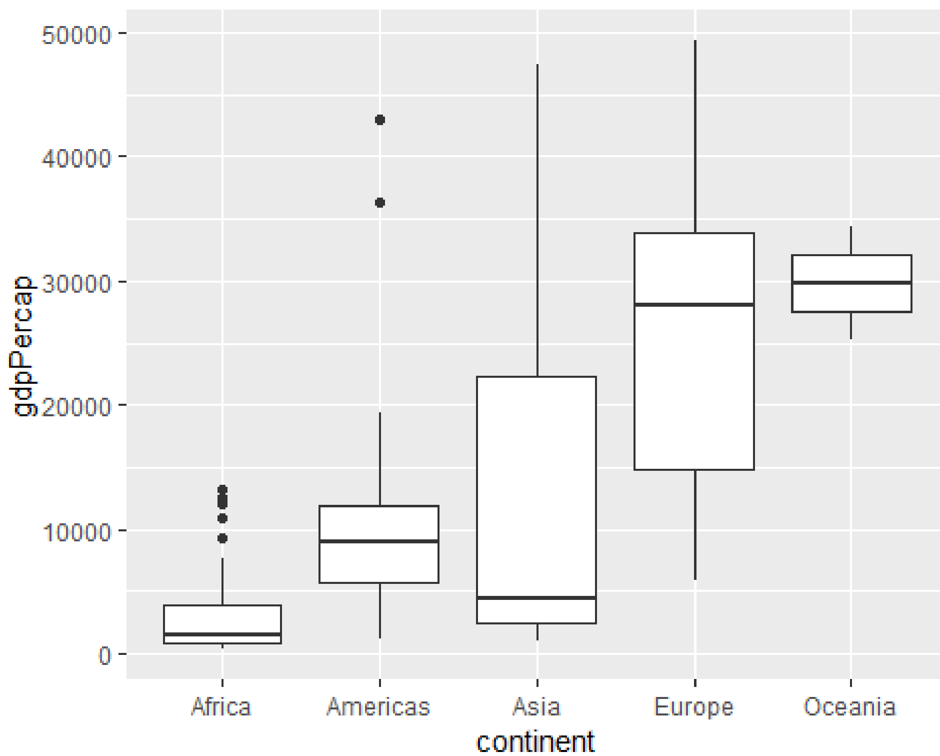
Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
gdpPercap	0	1	11680.07	12859.94	277.55	1624.84	6124.37	18008.84	49357.19	▇▂▁▂▁

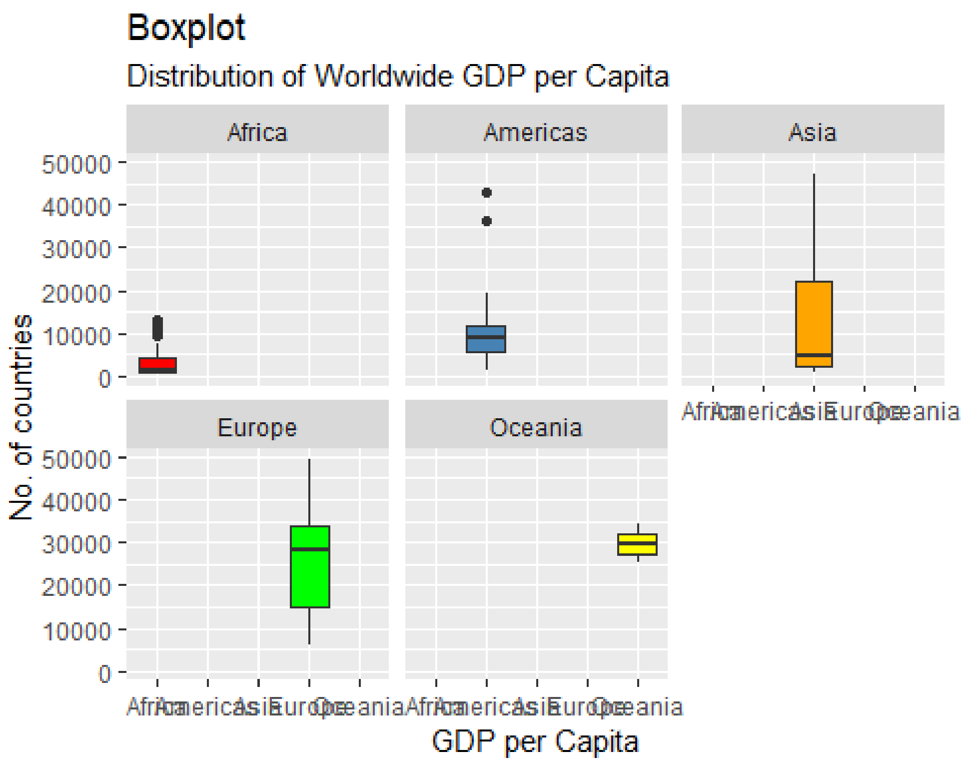
EDA III - Graphically Representation of Data

Task : Make Boxplot

```
ggplot(gap7, aes(x = continent , y = gdpPercap)) +
  geom_boxplot()
```




```
# Customized Boxplot
ggplot(gap7, aes(x = continent , y = gdpPercap)) +
  geom_boxplot(fill = c("red" , "steelblue ", "orange" , "green" , "yellow"))
+
  facet_wrap(~ continent) +
  labs(x = "GDP per Capita" , y = "No. of countries" , title = "Boxplot" ,
subtitle = "Distribution of Worldwide GDP per Capita")
```



Task : Fit the Simple Linear Model**

Model : $$GDP = \beta_0 + \beta_1(Continent) + \epsilon$$

```
gdp_model <- lm(gdpPercap ~ continent , data = gap7)
```

```
get_regression_table(gdp_model)
```

```
## # A tibble: 5 x 7
##   term           estimate std_error statistic p_value lower_ci
##   <chr>             <dbl>     <dbl>     <dbl>   <dbl>    <dbl>
## 1 intercept         3089.     1373.      2.25   0.026     375.
## 2 continent: A~     7914.     2409.      3.28   0.001    3150.
## 3 continent: A~     9384.     2203.      4.26   0        5027.
## 4 continent: E~    21965.     2270.      9.68   0       17478.
## 5 continent: O~    26721.     7133.      3.75   0       12616.
## # ... with 1 more variable: upper_ci <dbl>
```

Fitted Model :

$$\widehat{GDP} = 3089.03 + 7913.14(Americas) + 9383.99(Asia) + 21965.45(Europe) + 26721.16(Oceania)$$

Interpretation

- For every increase of 1 unit in *Americas* there is an associated increase of, on average, **7913.14** units of *GDP per Capita* .

- For every increase of 1 unit in *Asia* there is an associated increase of, on average, **9383.99** units of *GDP per Capita* .
- For every increase of 1 unit in *Europe* there is an associated increase of, on average, **21965.45** units of *GDP per Capita* .
- For every increase of 1 unit in *Oceania* there is an associated increase of, on average, **26721.16** units of *GDP per Capita* .

Summary of Model :

```
get_regression_summaries(gdp_model)

## # A tibble: 1 x 9
##   r_squared adj_r_squared       mse  rmse sigma statistic
##       <dbl>         <dbl>     <dbl> <dbl> <dbl>     <dbl>
## 1     0.424         0.407 94538944. 9723. 9899.      25.2
## # ... with 3 more variables: p_value <dbl>, df <dbl>,
## #   nobs <dbl>
```

The value of $R^2 = 0.42$, which indicates that ou model is **42%** good .

Observed and Residuals of our Model :

```
rp <- get_regression_points(gdp_model) ; head(rp)

## # A tibble: 6 x 5
##      ID gdpPercap continent gdpPercap_hat residual
##   <int>     <dbl> <fct>             <dbl>    <dbl>
## 1     1      975. Asia              12473.  -11498.
## 2     2     5937. Europe            25054.  -19117.
## 3     3     6223. Africa             3089.    3134.
## 4     4     4797. Africa             3089.    1708.
## 5     5    12779. Americas          11003.    1776.
## 6     6    34435. Oceania           29810.    4625.
```

Task : Iidentify the five countries with the five smallest (most negative) residuals .

```
rp %>%
  top_n(n = -5 , wt = residual)%>%
  arrange(residual)

## # A tibble: 5 x 5
##      ID gdpPercap continent gdpPercap_hat residual
##   <int>     <dbl> <fct>             <dbl>    <dbl>
## 1     2     5937. Europe            25054.  -19117.
## 2    13     7446. Europe            25054.  -17608.
## 3   132     8458. Europe            25054.  -16596.
## 4    85     9254. Europe            25054.  -15801.
## 5   112     9787. Europe            25054.  -15268.
```

Task : Iidentify the five countries with the five largest (most positive) residuals .

```
rp %>%
  top_n(n = 5 , wt = residual)%>%
  arrange(desc(residual))

## # A tibble: 5 x 5
##      ID gdpPercap continent gdpPercap_hat residual
##   <int>     <dbl> <fct>             <dbl>    <dbl>
## 1    72    47307. Asia              12473.   34834.
## 2   114    47143. Asia              12473.   34670.
## 3   135    42952. Americas          11003.   31949.
## 4    56    39725. Asia              12473.   27252.
## 5    21    36319. Americas          11003.   25316.
```

4 Multiple Regression

4.1 Multiple Linear Regression / MLR - Theory

(Multiple Linear Regression Analysis) The basic difference between simple and multiple regression is that in simple there is only one predictor x, whereas in multiple regression it must be 2 or $more$. We shall write a function to implement multiple regression analysis with 2 regressors or covariates.

1. **Model:** The Multiple Linear Regression Model is denoted as: $y_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \cdots \beta_i x_{ip} + \epsilon$ where, y is the response vaiable , $\beta_1 + \beta_2 + \cdots + \beta_i$ is regression coefficient and $x_1 + x_2 + \cdots + x_{ip}$ are predictors.

2. **Regressiom Coefficient:** Change in response y per unit change in regressor x.

3. **Formulas for Calculation**

$$(y, X, \beta, \sigma^2, I)$$

It is to be noted that y is the vector of responses, X is termed as model matrix and β iis known as vector of regression coefficients.However, σ^2 is known as residual variance, I stands for indentity matrix of order $n \times n$.

The method of least square is used to estimate β . This method states that we will close that value of β which will minimize error sum of squares defined as : $errorSS = e^T e = (y - X\beta)^T (y - X\beta)$ and the result is solution normal equations defined as: $(X^T X)\hat{\beta} = X^T y$ alternatively least square estimate of β is defined as: $\hat{\beta} = (X^T X)^{-1}(X^T y)$

This implies that variance covariance matrix of $\hat{\beta}$ is : $Var(\hat{\beta}) = \sigma^2 (X^T X)^{-1}$ and its estimate is $\widehat{Var(\hat{\beta})} = \widehat{\sigma^2} (X^T X)^{-1}$

The diagonal elements of this matrix are variances and non-diagonal elements are co-variances, Thus standard error of β is $SE(\hat{\beta}) = \sqrt{diag\left(\widehat{Var(\hat{\beta})}\right)}$ where $\widehat{\sigma^2} = \frac{ResidSS}{n-(p+1)} = MSresidual$ where, $ResidSS = (y - X\hat{\beta})^T (y - X\hat{\beta})$

4. **Sum of Squares -**
   * ***Total Sum of Square:*** $SST = Y^TY - n\bar{Y}^2$ with degree of freedom $n-1$
   ****Regression Sum of Square:*** $SS_{res} = \hat{\beta}^T X^T Y - n\bar{X}^2$ with degree of freedom k
   * ***Residual Sum of Square:*** $SSR = Y^TY\hat{\beta}^T X^T Y$ with degree of freedom n-k-1

5. **Hypothesis of SLR:**

Null Hypothesis $H_0: \beta_1 = \beta_2 = \cdots = \beta_i = \cdots = \beta_k = 0$

Alternative Hypothesis H_1: At least one $\beta_i's \neq 0 \quad ; i = 1,2,\ldots,k$

4.2 One Numerical & One Categorical Explanatory Variable

Here we are discussing about the data *evals* .

4.2.1 EDA

```
# Load the require library
library(tidyverse)
library(moderndive)
library(skimr)
library(ISLR)

# Data
data("evals")
```

Task : Select the columns *ID, score, age , gender* from the data **evals**

I^{st} step **Look at the Data** .

```
# Select te require data
evals_ch6 <- evals %>%
  select(ID , score , age , gender)

# Take the sample of size 5 from above data
evals_ch6 %>%
  sample_n(size = 5)

## # A tibble: 5 x 4
##      ID score   age gender
##   <int> <dbl> <int> <fct>
## 1   243   3.9    56 female
## 2   409   3.3    47 female
## 3   308   3.7    35 male
## 4   278   4.8    38 female
## 5    89   4.4    56 female
```

II^{nd} Step – **Summarizing the Data**

```
evals_ch6 %>%
  select(score , age , gender) %>%
  skim()
```

Data summary

Name	Piped data
Number of rows	463
Number of columns	3
_______________________	
Column type frequency:	
factor	1
numeric	2
________________________	
Group variables	None

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
gender	0	1	FALSE	2	mal: 268, fem: 195

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
score	0	1	4.17	0.54	2.3	3.8	4.3	4.6	5	▁▁▅▇▇
age	0	1	48.37	9.80	29.0	42.0	48.0	57.0	73	▅▇▇▆▁

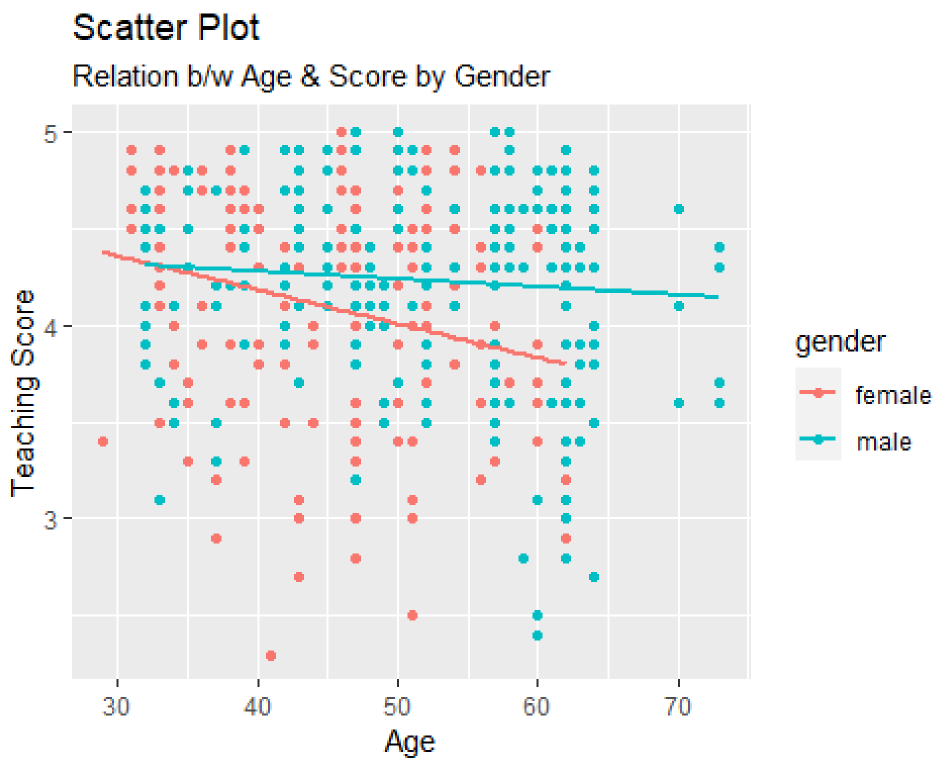
Task : Correlation Coefficient between the *score* and *age* .

```
evals_ch6 %>%
  get_correlation(formula = score ~ age)

## # A tibble: 1 x 1
##      cor
##    <dbl>
## 1 -0.107
```

III^{rd} Step – **Graphically Representation**

Task : Make a **Scatter Plot** between *score* and score and fill by *gender* .

```
ggplot(evals_ch6 , aes(age , score , col = gender)) +
  geom_point() +
  geom_smooth(method = "lm" , se = F) +
  labs(x = "Age" , y = "Teaching Score" , title = "Scatter Plot" , subtitle =
"Relation b/w Age & Score by Gender")

## `geom_smooth()` using formula 'y ~ x'
```



Female instructors are paying a harsher penalty for advanced age than the male instructors .

4.2.2 Regression Model

$$Score = \beta_0 + \beta_1(Age * Gender) + \epsilon$$

$$\Rightarrow \quad Score = \beta_0 + \beta_{Age}(Age) + \beta_{Male}(M) + \beta_{GM}(Age\_Male) + \epsilon$$

```
# Fit the Model :
score_model <- lm(score ~ age*gender , data = evals_ch6)

# Regression Table :
get_regression_table(score_model)
```

```
## # A tibble: 4 x 7
##   term           estimate std_error statistic p_value lower_ci
##   <chr>             <dbl>     <dbl>     <dbl>   <dbl>    <dbl>
## 1 intercept         4.88      0.205     23.8    0         4.48
## 2 age              -0.018     0.004     -3.92   0        -0.026
## 3 gender: male     -0.446     0.265     -1.68   0.094    -0.968
## 4 age:gendermale    0.014     0.006      2.45   0.015     0.003
## # ... with 1 more variable: upper_ci <dbl>
```

Our Model : $\widehat{Score} = 4.88 - 0.018(Age) - 0.446(Male) + 0.014(Age\_Male)$

Task : Accuracy of Model

```
get_regression_summaries(score_model)
```

```
## # A tibble: 1 x 9
##   r_squared adj_r_squared   mse  rmse sigma statistic p_value
##       <dbl>         <dbl> <dbl> <dbl> <dbl>     <dbl>   <dbl>
## 1     0.051         0.045 0.280 0.529 0.531      8.29       0
## # ... with 2 more variables: df <dbl>, nobs <dbl>
```

The value of $R^2 = 0.051$, that means that our Model is only *5%* Good .

Task : Observed / Fitted Values / Residuals

```
get_regression_points(score_model) -> rp ; head(rp)
```

```
## # A tibble: 6 x 6
##      ID score   age gender score_hat residual
##   <int> <dbl> <int> <fct>      <dbl>    <dbl>
## 1     1   4.7    36 female      4.25    0.448
## 2     2   4.1    36 female      4.25   -0.152
## 3     3   3.9    36 female      4.25   -0.352
## 4     4   4.8    36 female      4.25    0.548
## 5     5   4.6    59 male        4.20    0.399
## 6     6   4.3    59 male        4.20    0.099
```

Task : Top 5 +ve Residuals

```
rp %>%
  top_n(5 , residual) %>%
  arrange(desc(residual))
```

```
## # A tibble: 5 x 6
##      ID score   age gender score_hat residual
##   <int> <dbl> <int> <fct>      <dbl>    <dbl>
## 1   415   4.9    54 female      3.94    0.963
## 2   445   4.9    52 female      3.97    0.928
## 3   103   5      46 female      4.08    0.923
## 4   108   5      46 female      4.08    0.923
## 5    90   4.8    56 female      3.90    0.898
```

Task : Top 5 -ve Residuals

```
rp %>%
  top_n(-5 , residual) %>%
  arrange(residual)

## # A tibble: 5 x 6
##      ID score   age gender score_hat residual
##   <int> <dbl> <int> <fct>      <dbl>    <dbl>
## 1   162   2.3    41 female      4.16    -1.86
## 2   335   2.4    60 male        4.20    -1.80
## 3   337   2.5    60 male        4.20    -1.70
## 4    40   2.5    51 female      3.99    -1.49
## 5   329   2.7    64 male        4.18    -1.48
```

(LC6.1) : Compute the observed values, fitted values, and residuals not for the interaction model as we just did, but rather for the parallel slopes model we saved in score_model_parallel_slopes.

Model : $Slope = \beta_0 + \beta_{Age}(Age) + \beta_{Gender}(Gender) + \epsilon$

```
score_model <- lm(score ~ age + gender , data = evals_ch6)
get_regression_table(score_model)

## # A tibble: 3 x 7
##   term         estimate std_error statistic p_value lower_ci
##   <chr>           <dbl>     <dbl>     <dbl>   <dbl>    <dbl>
## 1 intercept       4.48      0.125     35.8    0         4.24
## 2 age            -0.009     0.003     -3.28   0.001    -0.014
## 3 gender: male    0.191     0.052      3.63   0         0.087
## # ... with 1 more variable: upper_ci <dbl>

regression_points_parallel <- get_regression_points(score_model)

head(regression_points_parallel)

## # A tibble: 6 x 6
##      ID score   age gender score_hat residual
##   <int> <dbl> <int> <fct>      <dbl>    <dbl>
## 1     1   4.7    36 female      4.17    0.528
## 2     2   4.1    36 female      4.17   -0.072
## 3     3   3.9    36 female      4.17   -0.272
## 4     4   4.8    36 female      4.17    0.628
## 5     5   4.6    59 male        4.16    0.437
## 6     6   4.3    59 male        4.16    0.137
```

4.3 Two Numerical Explanatory Variable

Here we use **Credit** dataset from **ISLR** package.

```
# Call the Data
data("Credit")
dim(Credit)

## [1] 400  12

names(Credit)

##  [1] "ID"        "Income"    "Limit"     "Rating"
##  [5] "Cards"     "Age"       "Education" "Gender"
##  [9] "Student"   "Married"   "Ethnicity" "Balance"

# View(Credit)
```

EDA

Task : Select the columns named *ID* , *Balance* , *Limit* , *Income* , *Rating* , *Age* and assign them new names as *ID* , *debt* , *credit_limit* , *income* , *credit_rating* , *age* repesctively .

I^{st} Step – **Looking at Data**

```
credit_ch6 <- Credit %>%
  as_tibble() %>%
  select(ID , debt = Balance , credit_limit = Limit , income = Income ,
credit_rating = Rating , age = Age)

glimpse(credit_ch6)

## Rows: 400
## Columns: 6
## $ ID            <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12~
## $ debt          <int> 333, 903, 580, 964, 331, 1151, 203, 8~
## $ credit_limit  <int> 3606, 6645, 7075, 9504, 4897, 8047, 3~
## $ income        <dbl> 14.891, 106.025, 104.593, 148.924, 55~
## $ credit_rating <int> 283, 483, 514, 681, 357, 569, 259, 51~
## $ age           <int> 34, 82, 71, 36, 68, 77, 37, 87, 66, 4~

# Sample of 5 obs.
credit_ch6 %>%
  sample_n(5)

## # A tibble: 5 x 6
##      ID  debt credit_limit income credit_rating   age
##   <int> <int>        <int>  <dbl>         <int> <int>
```

```
## 1    25     0         1757   10.7            156    57
## 2   291   159         3235   26.4            268    78
## 3   368   216         3615   23.8            263    70
## 4   286     0         1626   19.0            156    41
## 5   396   560         4100   12.1            307    32
```

II^{nd} Step – **Summary of debt , credit_limit , income**

```
credit_ch6 %>%
  select(debt , credit_limit , income) %>%
  skim()
```

Data summary

Name	Piped data
Number of rows	400
Number of columns	3
______________________	
Column type frequency:	
numeric	3
________________________	
Group variables	None

Variable type: numeric

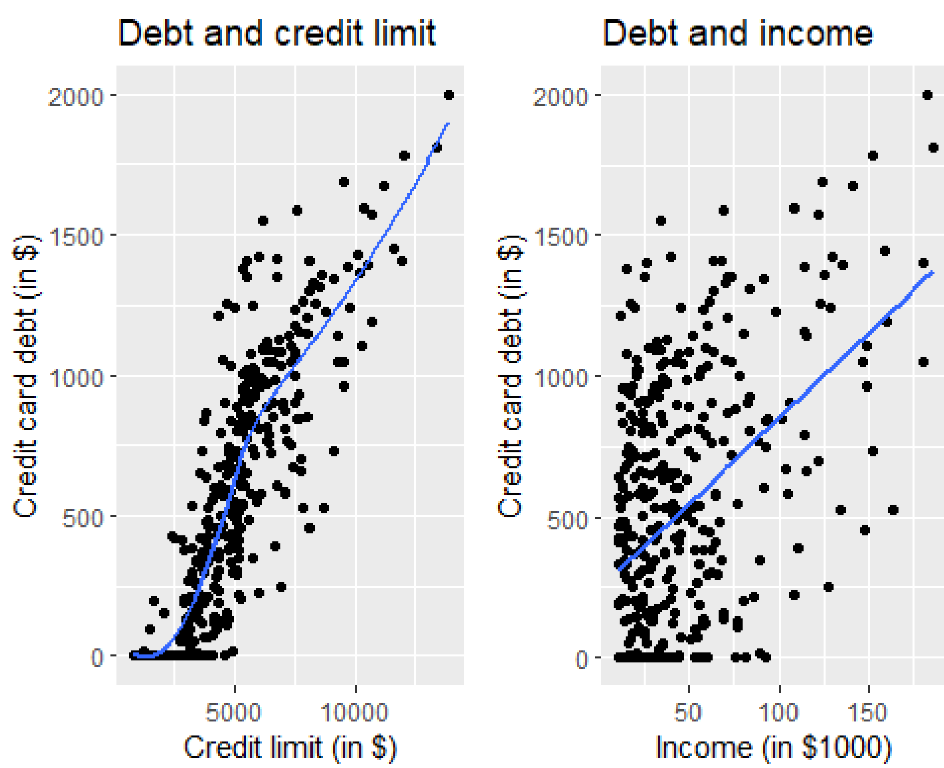
skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
debt	0	1	520.02	459.76	0.00	68.75	459.50	863.00	1999.00	▇▅▃▂▁
credit_limit	0	1	4735.60	2308.20	855.00	3088.00	4622.50	5872.75	13913.00	▅▇▇▂▁
income	0	1	45.22	35.24	10.35	21.01	33.12	57.47	186.63	▇▂▁▁▁

**Correlation* b/w *debt , credit_limit , income*

```
credit_ch6 %>%
  select(debt , credit_limit , income) %>%
  cor()

##                   debt credit_limit    income
## debt         1.0000000    0.8616973 0.4636565
## credit_limit 0.8616973    1.0000000 0.7920883
## income       0.4636565    0.7920883 1.0000000
```

Graphical Presentation

```r
# Divide the screen into 2 parts
library(cowplot)

# Make a Scatterplot for credit limit and debt
p1 <- ggplot(credit_ch6 , aes(credit_limit , debt)) +
  geom_point() +
  labs(x = "Credit limit (in $)", y = "Credit card debt (in $)",
title = "Debt and credit limit")+
  geom_smooth(methos = "lm" , se = F)

# Make a Scatter plot for income and debt
p2 <- ggplot(credit_ch6, aes(x = income, y = debt)) +
  geom_point() +
  labs(x = "Income (in $1000)", y = "Credit card debt (in $)",
title = "Debt and income") +
  geom_smooth(method = "lm", se = FALSE)

# Plotting the Both Graphs in a row
plot_grid(p1, p2)

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'

## `geom_smooth()` using formula 'y ~ x'
```



Regression Models

Model : $Debt = \beta_0 + \beta_{CL}(Credit\_Limit) + \beta_{Income}(Income) + \epsilon$

```r
# Fit regression model:
debt_model <- lm(debt ~ credit_limit + income, data = credit_ch6)

# Get regression table:
get_regression_table(debt_model)
```

```
## # A tibble: 3 x 7
##   term         estimate std_error statistic p_value lower_ci
##   <chr>           <dbl>     <dbl>     <dbl>   <dbl>    <dbl>
## 1 intercept    -385.       19.5       -19.8       0 -423.
## 2 credit_limit    0.264     0.006      45.0       0    0.253
## 3 income         -7.66      0.385     -19.9       0   -8.42
## # ... with 1 more variable: upper_ci <dbl>
```

$$\widehat{Debt} = -385.18 + 0.26(Credit\_Limit) - 7.66(Income)$$

Accuracy :

```r
debt_model %>%
  get_regression_summaries()
```

```
## # A tibble: 1 x 9
##   r_squared adj_r_squared    mse  rmse sigma statistic
##       <dbl>         <dbl>  <dbl> <dbl> <dbl>     <dbl>
## 1     0.871          0.87 27177.  165.  165.     1342.
## # ... with 3 more variables: p_value <dbl>, df <dbl>,
## #   nobs <dbl>
```

The value of $R^2 = 0.871$, which means that Our Model is **87%** is Good/Fitted.

Observed / Fitted Values & Residuals

```r
rp <- debt_model %>%
  get_regression_points()
head(rp)
```

```
## # A tibble: 6 x 6
##      ID  debt credit_limit income debt_hat residual
##   <int> <int>        <int>  <dbl>    <dbl>    <dbl>
## 1     1   333         3606   14.9     454.   -121.
## 2     2   903         6645  106.      559.    344.
## 3     3   580         7075  105.      683.   -103.
## 4     4   964         9504  149.      986.    -21.7
## 5     5   331         4897   55.9     481.   -150.
## 6     6  1151         8047   80.2    1127.     23.6
```

Top 5 +ve Residuals

```r
rp %>%
  top_n(5 , residual) %>%
  arrange(desc(residual))
```

```
## # A tibble: 5 x 6
##      ID  debt credit_limit income debt_hat residual
##   <int> <int>        <int>  <dbl>    <dbl>    <dbl>
## 1   223  1549         6207   33.4     999.     550.
## 2   127  1404         5533   26.4     875.     529.
## 3   204  1411         6784   68.2     885.     526.
## 4   274  1255         4706   16.8     730.     525.
## 5   208  1216         4391   10.8     692.     524.
```

4.3.1 EDA for Credit_Rating & Age

(LC 6.2) : Conduct a new exploratory data analysis with the same outcome variable y being *debt* but with *credit_rating* and *age* as the new explanatory variables x_1 and x_2 .

Looking at Data :

```
credit_ch6 %>%
  select(debt, credit_rating, age) %>%
  head()
```

```
## # A tibble: 6 x 3
##    debt credit_rating   age
##   <int>         <int> <int>
## 1   333           283    34
## 2   903           483    82
## 3   580           514    71
## 4   964           681    36
## 5   331           357    68
## 6  1151           569    77
```

Summary of Data

```
credit_ch6 %>%
  select(debt, credit_rating, age) %>%
  skim()
```

Data summary

Name	Piped data
Number of rows	400
Number of columns	3
_______________________	
Column type frequency:	
numeric	3
________________________	
Group variables	None

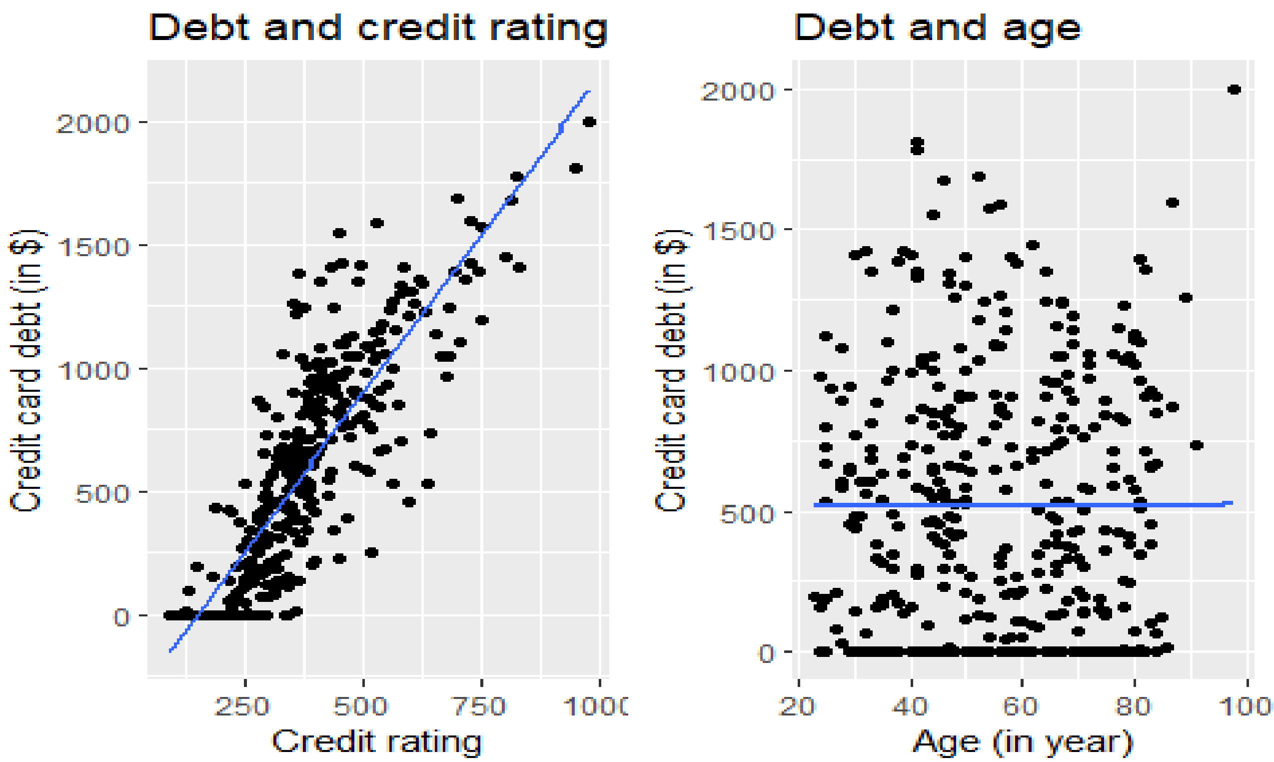
Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
debt	0	1	520.02	459.76	0	68.75	459.5	863.00	1999	▇▅▃▃▁
credit_rating	0	1	354.94	154.72	93	247.25	344.0	437.25	982	▆▇▃▁▁
age	0	1	55.67	17.25	23	41.75	56.0	70.00	98	▇▇▇▇▁

```
credit_ch6 %>%
  select(debt , credit_limit , income) %>%
  cor()

##                  debt credit_limit    income
## debt         1.0000000    0.8616973 0.4636565
## credit_limit 0.8616973    1.0000000 0.7920883
## income       0.4636565    0.7920883 1.0000000
```

Graphically Representation

```
# Scatter Plot for Credit Rating and Debt
p1 <- ggplot(credit_ch6, aes(x = credit_rating, y = debt)) +
  geom_point() +
  labs(x = "Credit rating", y = "Credit card debt (in $)",
    title = "Debt and credit rating") +
  geom_smooth(method = "lm", se = FALSE)

# Scatter Plot for Age and Debt
p2 <- ggplot(credit_ch6, aes(x = age, y = debt)) +
  geom_point() +
  labs(x = "Age (in year)", y = "Credit card debt (in $)",
    title = "Debt and age") +
  geom_smooth(method = "lm", se = FALSE)

plot_grid(p1 , p2)

## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
```



Regression Analysis Model : $Debt = \beta_0 + \beta_{CR}(Credit\_Rating) + \beta_{Age}(Age) + \epsilon$

```
# Fit regression model:
debt_model_2 <- lm(debt ~ credit_rating + age, data = credit_ch6)

# Get regression table:
get_regression_table(debt_model_2)

## # A tibble: 3 x 7
##   term          estimate std_error statistic p_value lower_ci
##   <chr>            <dbl>     <dbl>     <dbl>   <dbl>    <dbl>
## 1 intercept      -270.      44.8       -6.02       0   -358.
## 2 credit_rating     2.59     0.074     34.8        0      2.45
## 3 age              -2.35     0.668     -3.52       0     -3.66
## # ... with 1 more variable: upper_ci <dbl>
```

Fitted Model : $Debt = -269.6 + 2.6(Credit\_Rating) - 2.4(Age)$

Accuracy of Model :

```
debt_model_2 %>%
  get_regression_summaries()

## # A tibble: 1 x 9
##   r_squared adj_r_squared    mse  rmse sigma statistic
##       <dbl>         <dbl>  <dbl> <dbl> <dbl>     <dbl>
## 1     0.754         0.752 51965.  228.  229.      607.
```

```
## # ... with 3 more variables: p_value <dbl>, df <dbl>,
## #   nobs <dbl>
```

The value of $R^2 = 0.754$, which means that Our Model is **75%** is Good/Fitted.

Observed / Fitted Values & Residuals

```
debt_model_2 %>%
  get_regression_points() %>%
  head()

## # A tibble: 6 x 6
##      ID  debt credit_rating   age debt_hat residual
##   <int> <int>         <int> <int>    <dbl>    <dbl>
## 1     1   333           283    34     384.    -51.4
## 2     2   903           483    82     790.    113.
## 3     3   580           514    71     896.   -316.
## 4     4   964           681    36    1412.   -448.
## 5     5   331           357    68     496.   -165.
## 6     6  1151           569    77    1025.    126.
```

Top 5 -ve Residuals

```
debt_model_2 %>%
  get_regression_points() %>%
  top_n(-5 , residual) %>%
  arrange(residual)

## # A tibble: 5 x 6
##      ID  debt credit_rating   age debt_hat residual
##   <int> <int>         <int> <int>    <dbl>    <dbl>
## 1   276   529           636    50    1262.    -733.
## 2   279   250           518    78     890.    -640.
## 3   122   454           599    83    1089.    -635.
## 4    33   526           563    48    1078.    -552.
## 5   154     0           344    32     547.    -547.
```

4.3.2 EDA for MA_school Data

Looking at Data :

```
data("MA_schools")
dim(MA_schools)

## [1] 332   4

names(MA_schools)

## [1] "school_name"      "average_sat_math" "perc_disadvan"
## [4] "size"

MA_schools %>%
  head()
```

```
## # A tibble: 6 x 4
##   school_name    average_sat_math perc_disadvan size
##   <chr>                     <dbl>         <dbl> <fct>
## 1 Abington High               516          21.5 medium
## 2 Agawam High                 514          22.7 large
## 3 Amesbury High               534          14.6 large
## 4 Andover High                581           6.3 large
## 5 Arlington High              592          10.3 large
## 6 Ashland High                576          10.3 large
```

Summary of Data :

```
MA_schools %>%
  skim()
```

Data summary

Name	Piped data
Number of rows	332
Number of columns	4
_______________________	
Column type frequency:	
character	1
factor	1
numeric	2
________________________	
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
school_name	0	1	9	73	0	332	0

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
size	0	1	FALSE	3	lar: 235, med: 69, sma: 28

Variable type: numeric

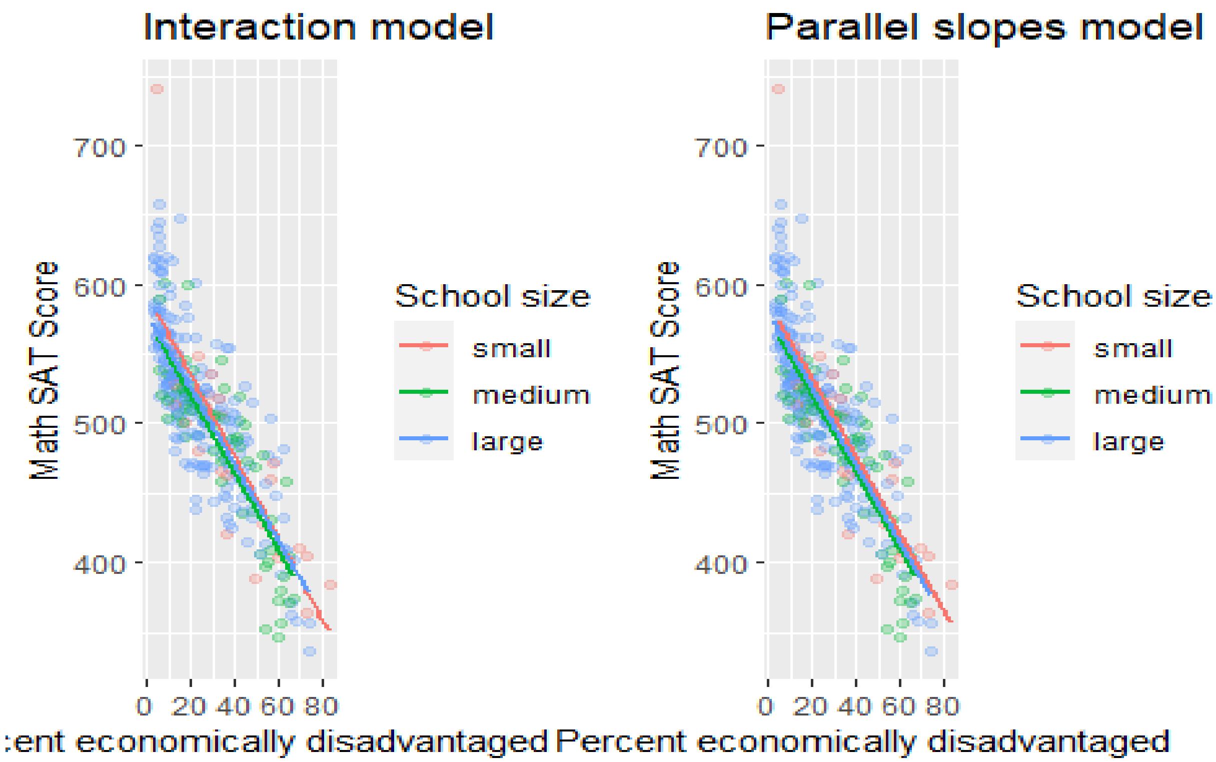
skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
average_sat_math	0	1	507.06	60.76	336.0	473.00	514	540.0	741.0	▂▃▇▂▁

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
perc_disadvan	0	1	26.70	18.24	3.1	11.78	22	38.4	83.3	▇▅▃▂▁

```
# Correlation of Data
MA_schools %>%
  select(average_sat_math , perc_disadvan) %>%
  cor()

##                   average_sat_math perc_disadvan
## average_sat_math        1.0000000    -0.8343829
## perc_disadvan          -0.8343829     1.0000000
```

Graphs

```
# Interaction model
p1 <- ggplot(MA_schools, aes(x = perc_disadvan, y = average_sat_math, color =
size)) +
  geom_point(alpha = 0.25) +
  geom_smooth(method = "lm", se = FALSE) +
  labs(x = "Percent economically disadvantaged", y = "Math SAT Score", color
= "School size", title = "Interaction model")

# Parallel slopes model
p2 <- ggplot(MA_schools, aes(x = perc_disadvan, y = average_sat_math, color =
size)) +
  geom_point(alpha = 0.25) +
  geom_parallel_slopes(se = FALSE) +
  labs(x = "Percent economically disadvantaged", y = "Math SAT Score", color
= "School size", title = "Parallel slopes model")

# Plot Both Graphs
plot_grid(p1 , p2)

## `geom_smooth()` using formula 'y ~ x'
```



Regression Analysis

Model : $ASM = \beta_0 + beta_1(PD * Size) + \epsilon$

```
# Fit the Model :
model_2_interaction <- lm(average_sat_math ~ perc_disadvan * size,
data = MA_schools)

# Model Table :
get_regression_table(model_2_interaction)

## # A tibble: 6 x 7
##   term           estimate std_error statistic p_value lower_ci
##   <chr>             <dbl>     <dbl>     <dbl>   <dbl>    <dbl>
## 1 intercept       594.       13.3      44.7     0       568.
## 2 perc_disadvan    -2.93      0.294    -9.96    0        -3.51
## 3 size: medium    -17.8      15.8      -1.12    0.263   -48.9
## 4 size: large     -13.3      13.8      -0.962   0.337   -40.5
## 5 perc_disadva~     0.146     0.371     0.393   0.694    -0.585
## 6 perc_disadva~     0.189     0.323     0.586   0.559    -0.446
## # ... with 1 more variable: upper_ci <dbl>
```

Accuracy of Model

```
model_2_interaction %>%
  get_regression_summaries()

## # A tibble: 1 x 9
##   r_squared adj_r_squared   mse  rmse sigma statistic p_value
##       <dbl>         <dbl> <dbl> <dbl> <dbl>     <dbl>   <dbl>
## 1     0.699         0.694 1107.  33.3  33.6      151.       0
## # ... with 2 more variables: df <dbl>, nobs <dbl>
```

The value of $R^2 = 6.99$, means that our model is **70%** Good / Fitted .

Observed / Fitted Values and Residuals

```
model_2_interaction %>%
  get_regression_points() %>%
  head()

## # A tibble: 6 x 6
##      ID average_sat_math perc_disadvan size   average_sat_mat~
##   <int>            <dbl>         <dbl> <fct>             <dbl>
## 1     1              516          21.5 medium             517.
## 2     2              514          22.7 large              519.
## 3     3              534          14.6 large              541.
## 4     4              581           6.3 large              564.
## 5     5              592          10.3 large              553.
## 6     6              576          10.3 large              553.
## # ... with 1 more variable: residual <dbl>
```

Top 5 +ve Residuals

```
model_2_interaction %>%
  get_regression_points() %>%
  top_n(5 , residual) %>%
  arrange(desc(residual))

## # A tibble: 5 x 6
##      ID average_sat_math perc_disadvan size  average_sat_mat~
##   <int>            <dbl>         <dbl> <fct>            <dbl>
## 1   168              741           5.2 small             579.
## 2    21              647          15.7 large             538.
## 3    77              658           6.1 large             564.
## 4    78              601          21.9 large             521.
## 5   156              645           5.6 large             566.
## # ... with 1 more variable: residual <dbl>
```

Note :-

Next Part is on Another File