Basic to Advanced Data Analytics in R-Studio

Tutor: Lumumba Wandera Victor

2023-07-06

Contents

MATRIX

Matrix Creation

```
A = matrix(c(5,6,2,8,9,2,4,5,1),ncol = 3, nrow = 3, byrow = T)
A
```

```
     [,1] [,2] [,3]
[1,]    5    6    2
[2,]    8    9    2
[3,]    4    5    1
```

Getting the determinant of a matrix

```
det(A)
```

```
[1] 3
```

Inverse

```
solve(A)
```

```
           [,1]       [,2] [,3]
[1,] -0.3333333  1.3333333   -2
[2,]  0.0000000 -1.0000000    2
[3,]  1.3333333 -0.3333333   -1
```

Matrix Operation

```
B <- matrix(c(3,4,6,2,3,4,5,2,5), ncol = 3, nrow = 3, byrow = T)
C <- matrix(c(8,5,3,2,3,5,9,3,3), ncol = 3, nrow = 3, byrow = T)
```

View the matrix

```
B
```

```
     [,1] [,2] [,3]
[1,]    3    4    6
[2,]    2    3    4
[3,]    5    2    5
```

```
C
```

```
     [,1] [,2] [,3]
[1,]    8    5    3
[2,]    2    3    5
[3,]    9    3    3
```

Matrix Addition

```
B+C
```

```
     [,1] [,2] [,3]
[1,]   11    9    9
[2,]    4    6    9
[3,]   14    5    8
```

Matrix Subtraction

```
B-C
```

```
     [,1] [,2] [,3]
[1,]   -5   -1    3
[2,]    0    0   -1
[3,]   -4   -1    2
```

Matrix Division

```
B/C
```

```
          [,1]      [,2]     [,3]
[1,] 0.3750000 0.8000000 2.000000
[2,] 1.0000000 1.0000000 0.800000
[3,] 0.5555556 0.6666667 1.666667
```

Matrix Multiplication

```
B*C
```

```
     [,1] [,2] [,3]
[1,]   24   20   18
[2,]    4    9   20
[3,]   45    6   15
```

```
B%*%C
```

```
     [,1] [,2] [,3]
[1,]   86   45   47
[2,]   58   31   33
[3,]   89   46   40
```

Getting the Identity Matrix

```
zapsmall(solve(A)%*%A)
```

```
     [,1] [,2] [,3]
[1,]    1    0    0
[2,]    0    1    0
[3,]    0    0    1
```

Mathematical Operations

```
y = 45+65
y
```

Addition

```
[1] 110
```

```
x = 563-546
x
```

Subtraction

```
[1] 17
```

Division

```
m = 563/87
m
```

```
[1] 6.471264
```

Multiplication

```
t = 56*56
t
```

```
[1] 3136
```

```r
sqrt(81)
```

Squares and Square roots

```
[1] 9
```

```r
sqrt(225)
```

```
[1] 15
```

```r
225^0.5
```

```
[1] 15
```

```r
5^2
```

```
[1] 25
```

Exponentials and Logarithmic

```r
log10(100)
```

```
[1] 2
```

To be checked!!!!!!

```r
exp(2)
```

```
[1] 7.389056
```

Data Importation (Comma Seperated Values, csv)

```r
data <- read.csv("Gapminder.csv")
head(data,5)
```

```
      country year population continent life_exp  gdp_cap ln_population
1 Afghanistan 1952    8425333      Asia   28.801 779.4453      6.925587
2 Afghanistan 1957    9240934      Asia   30.332 820.8530      6.965716
3 Afghanistan 1962   10267083      Asia   31.997 853.1007      7.011447
4 Afghanistan 1967   11537966      Asia   34.020 836.1971      7.062129
5 Afghanistan 1972   13079460      Asia   36.088 739.9811      7.116590
  ln_life_exp ln_gdpPercap
1    1.459408     6.658583
2    1.481901     6.710344
3    1.505109     6.748878
4    1.531734     6.728864
5    1.557363     6.606625
```

```
tail(data,5)
```

```
      country year population continent life_exp  gdp_cap ln_population
1700 Zimbabwe 1987    9216418    Africa   62.351 706.1573      6.964562
1701 Zimbabwe 1992   10704340    Africa   60.377 693.4208      7.029560
1702 Zimbabwe 1997   11404948    Africa   46.809 792.4500      7.057093
1703 Zimbabwe 2002   11926563    Africa   39.989 672.0386      7.076515
1704 Zimbabwe 2007   12311143    Africa   43.487 469.7093      7.090298
     ln_life_exp ln_gdpPercap
1700    1.794843     6.559838
1701    1.780872     6.541637
1702    1.670329     6.675129
1703    1.601941     6.510316
1704    1.638359     6.152114
```

Check the structure of the data

```
str(data)
```

```
'data.frame':   1704 obs. of  9 variables:
 $ country      : chr  "Afghanistan" "Afghanistan" "Afghanistan" "Afghanistan" ...
 $ year         : int  1952 1957 1962 1967 1972 1977 1982 1987 1992 1997 ...
 $ population   : int  8425333 9240934 10267083 11537966 13079460 14880372 12881816 13867957 16317921 2
 $ continent    : chr  "Asia" "Asia" "Asia" "Asia" ...
 $ life_exp     : num  28.8 30.3 32 34 36.1 ...
 $ gdp_cap      : num  779 821 853 836 740 ...
 $ ln_population: num  6.93 6.97 7.01 7.06 7.12 ...
 $ ln_life_exp  : num  1.46 1.48 1.51 1.53 1.56 ...
 $ ln_gdpPercap : num  6.66 6.71 6.75 6.73 6.61 ...
```

Manual Data Entry

```
age <- c(45,65,34,32,23,25,56,76,45,22,21,45,34,56,54)
age
```

```
 [1] 45 65 34 32 23 25 56 76 45 22 21 45 34 56 54
```

```
height <- c(122,134,144,165,155,133,123,132,145,154,166,134,121,154,165)
height
```

```
 [1] 122 134 144 165 155 133 123 132 145 154 166 134 121 154 165
```

Column Binding

```
height_age <- cbind(age, height)
height_age
```

```
      age height
 [1,]  45    122
 [2,]  65    134
 [3,]  34    144
 [4,]  32    165
 [5,]  23    155
 [6,]  25    133
 [7,]  56    123
 [8,]  76    132
 [9,]  45    145
[10,]  22    154
[11,]  21    166
[12,]  45    134
[13,]  34    121
[14,]  56    154
[15,]  54    165
```

Data Framing

```
mydata <- data.frame(age, height)
head(mydata,5)
```

```
  age height
1  45    122
2  65    134
3  34    144
4  32    165
5  23    155
```

Descriptive Statistics

```
library(stargazer)
library(gtsummary)
stargazer(data[,-2], type = "text")
```

```
=============================================================================
Statistic       N        Mean          St. Dev.       Min        Max
-----------------------------------------------------------------------------
population    1,704 29,601,212.000 106,157,897.000 60,011  1,318,683,096
life_exp      1,704     59.474         12.917      23.599     82.603
gdp_cap       1,704   7,215.327       9,857.455    241.166  113,523.100
ln_population 1,704     6.847           0.697       4.778      9.120
ln_life_exp   1,704     1.763           0.101       1.373      1.917
ln_gdpPercap  1,704     8.159           1.241       5.485     11.640
-----------------------------------------------------------------------------
```

Additional Way of Displaying Summary Statistics.

```r
### Load the libraries
library(ggplot2)
library(devtools)
library(predict3d)
library(psych)
library(dplyr)
library(gtsummary)
library(DescTools)
library(nortest)
library(lmtest)
library(sandwich)
```

Display the Summary Statistics

```r
knitr::kable(
  describeBy(data[,-1]) %>% round(2)
)
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
year	1	1704	1979.50	17.27	1979.50	1979.50	22.24	1952.00	2.007000e+03	5.500000e+01	0.00	-1.22	0.42
population	2	1704	29601212.32	106157896.74	7023595.50	11399459.45	7841473.62	60011.00	1.318683e+09	1.318623e+09	8.33	77.62	2571683.45
continent*	3	1704	2.33	1.21	2.00	2.27	1.48	1.00	5.000000e+00	4.000000e+00	0.25	-1.34	0.03
life_exp	4	1704	59.47	12.92	60.71	59.92	16.10	23.60	8.260000e+01	5.910000e+01	-0.25	-1.13	0.31
gdp_cap	5	1704	7215.33	9857.45	3531.85	5221.44	4007.61	241.17	1.135231e+05	1.132820e+05	3.84	27.40	238.80
ln_population	6	1704	6.85	0.70	6.85	6.85	0.62	4.78	9.120000e+00	4.340000e+00	0.08	0.47	0.02
ln_life_exp	7	1704	1.76	0.10	1.78	1.77	0.11	1.37	1.920000e+00	5.400000e-01	-0.57	-0.66	0.00
ln_gdpPercap	8	1704	8.16	1.24	8.17	8.14	1.51	5.49	1.164000e+01	6.150000e+00	0.11	-0.95	0.03

Correlationa and Covariances

```r
data22 <- read.csv("german_credit__data.csv")
attach(data22)
head(data22,5)
```

```
  S.no Age    Sex Job Housing Saving.accounts Checking.account Credit.amount
1    0  67   male   2     own            <NA>           little          1169
2    1  22 female   2     own          little         moderate          5951
3    2  49   male   1     own          little             <NA>          2096
4    3  45   male   2    free          little           little          7882
5    4  53   male   2    free          little           little          4870
```

```
  Duration             Purpose
1        6            radio/TV
2       48            radio/TV
3       12           education
4       42 furniture/equipment
5       24                 car
```

How many observations do we have in our data set

```
str(data22)
```

```
'data.frame':   1000 obs. of  10 variables:
 $ S.no            : int  0 1 2 3 4 5 6 7 8 9 ...
 $ Age             : int  67 22 49 45 53 35 53 35 61 28 ...
 $ Sex             : chr  "male" "female" "male" "male" ...
 $ Job             : int  2 2 1 2 2 1 2 3 1 3 ...
 $ Housing         : chr  "own" "own" "own" "free" ...
 $ Saving.accounts : chr  NA "little" "little" "little" ...
 $ Checking.account: chr  "little" "moderate" NA "little" ...
 $ Credit.amount   : int  1169 5951 2096 7882 4870 9055 2835 6948 3059 5234 ...
 $ Duration        : int  6 48 12 42 24 36 24 36 12 30 ...
 $ Purpose         : chr  "radio/TV" "radio/TV" "education" "furniture/equipment" ...
```

Identify the missing observations for each variables

```
library(mlr)
library(tidyverse)
map_dbl(data22, ~sum(is.na(.)))
```

```
            S.no              Age              Sex              Job
               0                0                0                0
         Housing  Saving.accounts Checking.account    Credit.amount
               0              183              394                0
        Duration          Purpose
               0                0
```

Eliminate Missing Observations

```
data22<- na.omit(data22)
```

Confirm the Remaining Observations

```
str(data22)
```

```
'data.frame':   522 obs. of  10 variables:
 $ S.no            : int  1 3 4 7 9 10 11 12 13 14 ...
 $ Age             : int  22 45 53 35 28 25 24 22 60 28 ...
 $ Sex             : chr  "female" "male" "male" "male" ...
 $ Job             : int  2 2 2 3 3 2 2 2 1 2 ...
 $ Housing         : chr  "own" "free" "free" "rent" ...
 $ Saving.accounts : chr  "little" "little" "little" "little" ...
 $ Checking.account: chr  "moderate" "little" "little" "moderate" ...
 $ Credit.amount   : int  5951 7882 4870 6948 5234 1295 4308 1567 1199 1403 ...
 $ Duration        : int  48 42 24 36 30 12 48 12 24 15 ...
 $ Purpose         : chr  "radio/TV" "furniture/equipment" "car" "car" ...
 - attr(*, "na.action")= 'omit' Named int [1:478] 1 3 6 7 9 17 18 20 21 25 ...
  ..- attr(*, "names")= chr [1:478] "1" "3" "6" "7" ...
```

Confirm if the data set still has missing observations

```
map_dbl(data22, ~sum(is.na(.)))
```

```
          S.no            Age            Sex            Job 
             0              0              0              0 
       Housing Saving.accounts Checking.account  Credit.amount 
             0              0              0              0 
      Duration        Purpose 
             0              0 
```

Correlation Matrix

```
COR <- data.frame(Age, Credit.amount,Duration)
head(COR,5)
```

```
  Age Credit.amount Duration
1  67          1169        6
2  22          5951       48
3  49          2096       12
4  45          7882       42
5  53          4870       24
```

```
cor(COR)
```

```
                    Age Credit.amount    Duration
Age           1.00000000    0.03271642 -0.03613637
Credit.amount 0.03271642    1.00000000  0.62498420
Duration      -0.03613637    0.62498420  1.00000000
```

Visualize the Results in Stargazer

```
stargazer(cor(COR),type = "text")
```

```
==========================================
               Age   Credit.amount Duration
------------------------------------------
Age              1        0.033      -0.036
Credit.amount 0.033         1        0.625
Duration      -0.036     0.625        1
------------------------------------------
```

The Covariance Matrix

```
stargazer(cov(COR), type = "text")
```

```
================================================
                 Age    Credit.amount  Duration
------------------------------------------------
Age             129.401    1,050.523     -4.957
Credit.amount 1,050.523 7,967,843.000 21,273.750
Duration       -4.957    21,273.750    145.415
------------------------------------------------
```

Basic R Commands

- head
- tail
- str
- list
- attach
- Renaming data set
- View

Example of how these commands are used

```
list(data22)
```

```
[[1]]
     S.no Age    Sex Job Housing Saving.accounts Checking.account Credit.amount
2       1  22 female   2     own          little         moderate          5951
4       3  45   male   2    free          little           little          7882
5       4  53   male   2    free          little           little          4870
8       7  35   male   3    rent          little         moderate          6948
10      9  28   male   3     own          little         moderate          5234
11     10  25 female   2    rent          little         moderate          1295
12     11  24 female   2    rent          little           little          4308
```

13	12	22	female	2	own	little	moderate	1567
14	13	60	male	1	own	little	little	1199
15	14	28	female	2	rent	little	little	1403
16	15	32	female	1	own	moderate	little	1282
19	18	44	female	3	free	little	moderate	12579
22	21	44	male	2	rent	quite rich	little	2647
23	22	48	male	1	rent	little	little	2241
24	23	44	male	2	own	moderate	moderate	1804
26	25	36	male	1	own	little	little	1374
28	27	42	female	2	rent	rich	rich	409
29	28	34	male	2	own	little	moderate	2415
30	29	63	male	2	own	little	little	6836
31	30	36	male	2	own	rich	moderate	1913
32	31	27	male	2	own	little	little	4020
33	32	30	male	2	own	moderate	moderate	5866
35	34	33	female	3	own	little	rich	1474
36	35	25	male	1	own	little	moderate	4746
38	37	37	male	2	own	little	rich	2100
39	38	37	male	2	own	little	rich	1225
40	39	24	male	2	own	little	moderate	458
42	41	26	male	2	own	quite rich	moderate	1158
43	42	44	male	1	own	little	moderate	6204
44	43	24	male	2	rent	moderate	little	6187
45	44	58	female	1	free	little	little	6143
48	47	23	female	0	rent	quite rich	little	1352
52	51	30	male	3	own	little	moderate	5965
55	54	57	male	2	free	little	moderate	2225
59	58	23	female	3	own	little	rich	1961
60	59	23	female	1	rent	little	little	6229
61	60	27	male	2	own	little	moderate	1391
63	62	61	male	3	free	little	moderate	1953
64	63	25	male	2	own	little	moderate	14421
68	67	22	male	2	own	rich	moderate	1007
73	72	51	male	3	free	little	little	1164
74	73	41	female	1	own	little	moderate	5954
76	75	66	male	3	free	little	little	1526
77	76	34	male	2	own	little	little	3965
78	77	51	male	2	own	little	moderate	4771
80	79	22	male	2	own	little	moderate	3832
84	83	58	female	1	own	little	little	1755
85	84	52	male	1	own	little	little	2315
87	86	27	female	2	own	little	moderate	1295
88	87	47	male	2	free	moderate	moderate	12612
89	88	30	male	3	own	moderate	little	2249
90	89	28	male	2	own	little	little	1108
92	91	54	male	2	own	little	little	1409
95	94	54	male	2	own	rich	moderate	1318
96	95	58	male	2	rent	little	moderate	15945
98	97	34	male	2	own	moderate	moderate	2622
99	98	36	male	2	own	little	moderate	2337
102	101	24	male	2	rent	little	moderate	2323
104	103	35	male	2	rent	little	moderate	1919
106	105	39	male	3	own	little	moderate	11938
108	107	32	male	2	own	little	moderate	6078

110	109	35	male	2	own	quite rich	moderate	1410
111	110	31	male	2	own	moderate	moderate	1449
112	111	23	female	2	rent	little	rich	392
113	112	28	male	1	rent	little	moderate	6260
115	114	35	male	2	own	quite rich	little	1680
119	118	23	female	2	own	quite rich	little	4281
120	119	36	male	3	own	quite rich	moderate	2366
121	120	25	female	2	own	little	little	1835
124	123	63	male	2	free	little	rich	781
126	125	30	male	2	own	little	little	2121
127	126	40	male	1	own	little	little	701
128	127	30	male	2	own	little	moderate	639
129	128	34	male	3	own	little	moderate	1860
130	129	29	female	2	own	little	little	3499
132	131	29	male	2	own	little	little	6887
138	137	66	male	1	own	quite rich	moderate	766
140	139	44	female	1	rent	little	rich	1881
141	140	27	male	0	own	rich	rich	709
142	141	30	female	3	own	little	moderate	4795
143	142	27	male	3	own	little	little	3416
144	143	22	male	2	own	little	little	2462
146	145	30	male	2	own	moderate	moderate	3566
147	146	39	female	2	own	little	little	860
149	148	28	male	2	own	little	little	5371
153	152	24	male	2	own	little	rich	5848
154	153	29	female	2	rent	rich	moderate	7758
155	154	36	male	3	rent	moderate	moderate	6967
156	155	20	female	2	rent	little	little	1282
157	156	48	male	2	own	moderate	little	1288
158	157	45	male	1	own	little	little	339
159	158	38	male	2	own	moderate	moderate	3512
164	163	70	male	3	free	little	moderate	7308
167	166	33	female	2	own	little	little	1131
168	167	20	female	2	own	rich	moderate	1577
170	169	31	male	2	own	little	moderate	1935
171	170	33	male	2	rent	little	little	950
173	172	34	female	3	own	little	moderate	2064
174	173	33	male	2	own	little	moderate	1414
175	174	26	male	2	own	little	little	3414
177	176	42	male	2	own	little	little	2577
178	177	52	male	2	own	quite rich	little	338
180	179	65	male	2	own	little	little	571
182	181	30	male	3	own	little	moderate	4455
185	184	36	male	2	own	little	moderate	884
187	186	74	female	3	free	little	moderate	5129
188	187	68	male	0	free	little	moderate	1175
189	188	20	male	2	own	moderate	little	674
190	189	33	female	2	own	little	moderate	3244
192	191	34	male	1	free	moderate	moderate	3844
193	192	36	male	2	own	little	moderate	3915
195	194	21	male	2	rent	moderate	moderate	3031
196	195	34	female	3	own	little	moderate	1501
198	197	27	female	2	rent	moderate	moderate	951
200	199	40	male	3	own	little	moderate	4297

202	201	27	male	1	own	little	little	1168
204	203	21	male	2	rent	little	little	902
206	205	38	male	3	free	little	little	10623
208	207	26	male	2	own	little	moderate	1424
209	208	21	male	1	own	little	little	6568
213	212	50	male	2	own	little	little	5293
214	213	66	male	3	own	little	rich	1908
217	216	31	male	2	own	little	little	3104
218	217	23	male	2	own	little	rich	3913
219	218	24	male	1	rent	little	little	3021
221	220	26	male	1	own	little	moderate	625
227	226	27	male	2	own	rich	moderate	10961
228	227	53	male	3	free	little	little	7865
230	229	22	male	2	free	little	little	3149
231	230	26	male	2	own	little	rich	4210
234	233	25	male	1	own	little	moderate	866
236	235	30	male	3	own	little	little	1823
238	237	61	male	1	rent	moderate	moderate	2767
240	239	39	male	2	own	little	little	2522
243	242	24	male	2	free	little	little	4605
249	248	26	male	2	own	little	rich	1925
251	250	39	female	1	own	rich	little	666
252	251	46	female	1	own	little	rich	2251
253	252	24	female	2	own	little	moderate	2150
258	257	29	male	2	free	little	little	2149
261	260	27	male	2	own	little	little	1657
262	261	55	female	2	own	little	little	1603
263	262	36	male	3	free	little	little	5302
266	265	37	male	2	own	little	moderate	802
269	268	45	male	3	own	little	little	8978
274	273	28	male	2	own	little	moderate	3060
275	274	34	male	1	own	little	little	11998
285	284	37	male	2	own	moderate	moderate	3878
286	285	35	female	1	own	little	little	10722
287	286	26	male	2	own	little	little	4788
288	287	31	male	3	free	moderate	moderate	7582
289	288	49	female	2	own	little	moderate	1092
290	289	48	male	2	own	little	little	1024
292	291	28	male	3	rent	little	moderate	9398
293	292	44	female	3	free	little	little	6419
294	293	56	male	2	free	little	rich	4796
296	295	26	female	2	own	little	moderate	9960
300	299	32	male	2	own	rich	moderate	2745
302	301	42	female	2	own	little	moderate	3804
304	303	49	male	2	own	little	little	1038
308	307	33	male	1	own	moderate	little	727
309	308	24	female	2	own	little	moderate	1237
310	309	22	male	1	rent	little	moderate	276
313	312	26	female	2	own	little	rich	3749
314	313	25	male	1	own	little	moderate	685
316	315	31	male	2	own	little	little	2746
317	316	38	male	1	own	little	little	708
320	319	27	female	1	own	little	little	3643
321	320	28	male	3	own	little	moderate	4249

322	321	32	male	2	own	little	little	1938
323	322	34	male	3	free	little	little	2910
324	323	28	male	2	own	rich	little	2659
326	325	39	male	1	own	little	little	3398
329	328	31	male	2	own	little	rich	4473
330	329	28	male	2	own	little	moderate	1068
331	330	75	male	3	free	little	little	6615
333	332	24	female	3	own	moderate	moderate	7408
335	334	23	male	2	rent	little	little	4110
336	335	44	male	3	rent	little	little	3384
337	336	23	female	1	own	little	moderate	2101
339	338	28	male	2	own	little	little	4169
340	339	31	male	1	own	little	moderate	1521
341	340	24	female	2	free	little	moderate	5743
342	341	26	female	1	rent	little	little	3599
343	342	25	male	2	rent	quite rich	moderate	3213
344	343	33	male	3	own	little	moderate	4439
345	344	37	male	1	own	little	rich	3949
347	346	23	male	2	own	little	moderate	882
348	347	23	female	0	rent	quite rich	moderate	3758
350	349	32	male	2	free	rich	moderate	1136
352	351	29	female	2	own	little	moderate	959
354	353	28	male	2	rent	little	little	6199
356	355	23	male	1	own	little	moderate	1246
360	359	23	female	2	rent	little	little	2406
363	362	36	female	2	own	little	rich	2247
365	364	25	male	0	own	little	little	2473
368	367	22	female	2	rent	little	little	3650
369	368	42	male	2	own	little	little	3446
370	369	40	female	2	rent	little	moderate	3001
375	374	60	female	3	free	moderate	moderate	14782
376	375	37	female	2	rent	little	little	7685
379	378	57	male	3	free	little	moderate	14318
382	381	38	female	3	free	little	moderate	12976
384	383	26	male	2	own	little	rich	1330
388	387	40	male	3	own	little	moderate	7374
389	388	27	male	2	own	quite rich	moderate	2326
392	391	19	female	1	rent	rich	moderate	983
393	392	39	male	3	free	little	little	3249
394	393	31	female	2	own	little	little	1957
396	395	32	male	2	rent	moderate	moderate	11760
397	396	55	female	3	free	little	little	2578
398	397	46	male	2	own	little	little	2348
399	398	46	male	2	rent	little	moderate	1223
406	405	22	male	2	own	little	moderate	2039
408	407	27	male	2	own	little	little	1053
410	409	28	male	2	own	quite rich	rich	939
411	410	20	female	2	own	little	moderate	1967
417	416	33	male	1	own	little	little	2579
423	422	47	male	1	own	little	moderate	958
426	425	21	male	2	rent	little	moderate	2779
430	429	55	female	0	free	little	little	1190
432	431	29	male	3	own	little	moderate	11328
433	432	36	male	3	free	little	little	1872

435	434	25	male	2	own	little	little	2136
439	438	65	male	0	own	little	little	3394
440	439	26	female	0	own	little	rich	609
442	441	30	female	2	own	little	little	1620
443	442	29	male	2	own	little	moderate	2629
445	444	30	female	3	own	little	moderate	5096
447	446	34	female	2	own	little	little	1842
448	447	35	male	2	own	little	moderate	2576
450	449	61	male	2	own	rich	moderate	1512
455	454	31	male	2	own	little	little	4817
457	456	36	male	2	rent	little	little	3905
458	457	35	male	2	free	little	little	3386
459	458	27	female	2	own	little	little	343
461	460	37	male	2	own	little	little	3620
462	461	36	male	2	own	little	little	1721
463	462	34	female	3	rent	little	moderate	3017
466	465	63	male	2	own	little	little	2924
467	466	29	female	1	rent	little	little	1659
471	470	22	male	2	rent	moderate	moderate	3092
472	471	23	female	2	own	little	little	448
473	472	28	male	1	own	little	little	654
475	474	33	male	2	own	little	moderate	1245
476	475	26	female	2	rent	little	little	3114
478	477	25	male	2	own	little	rich	5152
479	478	39	male	1	own	moderate	moderate	1037
480	479	44	male	2	own	little	little	1478
481	480	23	female	1	own	little	moderate	3573
482	481	26	male	2	own	little	moderate	1201
483	482	57	female	2	rent	rich	little	3622
486	485	47	male	3	own	little	moderate	1209
492	491	42	female	3	free	little	moderate	8318
495	494	39	male	1	rent	little	little	2122
497	496	29	male	3	rent	moderate	moderate	9034
499	498	32	male	1	own	little	moderate	1301
500	499	28	male	2	own	moderate	rich	1323
501	500	27	female	2	own	little	little	3123
502	501	42	male	2	free	little	little	5493
503	502	49	male	2	own	moderate	rich	1126
504	503	38	male	2	own	moderate	moderate	1216
505	504	24	female	2	rent	little	little	1207
507	506	36	male	2	own	quite rich	rich	2360
508	507	34	male	3	own	moderate	moderate	6850
511	510	26	male	2	own	little	little	759
513	512	26	male	2	rent	little	rich	2687
514	513	20	male	2	rent	little	moderate	585
516	515	37	female	2	own	little	little	609
517	516	40	male	1	own	little	little	1361
519	518	43	male	2	own	moderate	little	1203
522	521	24	female	2	own	little	little	3190
523	522	53	male	2	free	little	little	7119
525	524	26	female	1	own	little	moderate	1113
526	525	30	male	2	own	little	moderate	7966
529	528	31	male	2	rent	little	little	2302
530	529	41	male	1	own	little	little	662

531	530	32	male	2	own	little	moderate	2273
532	531	28	female	2	rent	moderate	moderate	2631
536	535	33	male	2	rent	little	rich	2319
538	537	37	female	2	own	little	moderate	3612
539	538	42	male	3	free	little	little	7763
540	539	45	female	1	own	little	rich	3049
541	540	23	male	2	rent	little	moderate	1534
544	543	34	male	1	own	little	rich	2864
546	545	43	male	2	free	little	little	1333
549	548	24	female	1	own	little	little	626
553	552	34	male	2	own	little	little	6999
554	553	27	male	2	own	moderate	moderate	1995
555	554	67	female	3	own	little	moderate	1199
556	555	22	male	2	own	little	moderate	1331
557	556	28	female	2	own	moderate	moderate	2278
559	558	27	male	2	own	little	little	3552
560	559	31	male	1	own	little	moderate	1928
562	561	24	male	1	rent	little	little	1546
563	562	29	female	2	own	little	rich	683
566	565	23	female	2	rent	moderate	moderate	1553
567	566	36	male	2	own	little	little	1372
570	569	31	female	2	own	little	little	6758
571	570	23	female	1	rent	little	little	3234
574	573	22	female	1	own	little	little	806
575	574	27	male	1	own	little	moderate	1082
577	576	27	female	2	own	little	moderate	2930
579	578	27	male	2	own	little	moderate	2820
581	580	30	male	2	own	little	moderate	1056
582	581	49	male	1	own	little	moderate	3124
584	583	33	male	1	rent	little	moderate	2384
586	585	20	female	2	rent	little	little	2039
587	586	36	male	2	rent	little	little	2799
588	587	21	male	1	own	little	little	1289
589	588	47	male	1	own	little	little	1217
590	589	60	male	2	own	little	little	2246
591	590	58	female	1	own	little	little	385
594	593	20	female	1	rent	little	moderate	2718
596	595	32	female	1	own	moderate	moderate	931
597	596	23	female	2	rent	little	little	1442
598	597	36	male	1	own	little	moderate	4241
601	600	45	female	2	own	little	moderate	2329
602	601	30	female	2	own	little	moderate	918
603	602	34	female	1	free	little	moderate	1837
605	604	23	female	2	own	little	rich	1275
606	605	22	male	2	own	quite rich	little	2828
608	607	50	female	2	free	moderate	moderate	2671
611	610	22	female	2	own	moderate	little	741
612	611	48	female	1	free	moderate	rich	1240
613	612	29	female	2	own	rich	little	3357
614	613	22	female	2	rent	little	little	3632
618	617	37	male	2	rent	little	little	3676
619	618	21	female	2	rent	moderate	moderate	3441
621	620	27	male	2	own	little	moderate	3652
624	623	22	female	2	rent	little	little	1858

625	624	65	male	2	free	little	little	2600
627	626	41	male	2	own	little	rich	2116
628	627	29	male	2	own	moderate	moderate	1437
631	630	28	female	2	own	little	little	3660
632	631	44	male	2	own	little	little	1553
635	634	25	female	1	own	little	moderate	1355
640	639	26	male	2	own	little	little	4370
641	640	27	female	0	own	little	little	750
642	641	38	male	1	own	little	moderate	1308
645	644	32	male	3	own	little	little	1880
647	646	32	male	2	own	little	little	4583
649	648	38	male	2	free	little	rich	947
650	649	40	male	1	rent	little	little	684
651	650	50	male	3	free	little	little	7476
652	651	37	male	1	own	little	moderate	1922
653	652	45	male	2	own	little	little	2303
654	653	42	male	3	own	moderate	moderate	8086
656	655	22	male	2	free	little	little	3973
657	656	41	male	1	own	little	moderate	888
659	658	28	female	2	own	little	moderate	4221
660	659	41	male	2	own	little	moderate	6361
661	660	23	male	2	rent	little	rich	1297
664	663	35	male	3	own	little	moderate	1050
665	664	50	female	1	own	little	rich	1047
667	666	34	male	2	own	rich	moderate	3496
669	668	43	male	2	rent	little	little	4843
670	669	47	male	2	own	little	rich	3017
678	677	24	male	2	own	moderate	moderate	5595
679	678	64	male	1	rent	little	little	2384
685	684	31	male	1	own	moderate	moderate	9857
688	687	30	male	2	free	moderate	moderate	2862
690	689	31	male	2	own	rich	little	3651
691	690	25	male	2	own	little	little	975
692	691	25	female	1	own	moderate	moderate	2631
693	692	29	male	2	own	moderate	moderate	2896
697	696	29	male	2	own	little	moderate	1103
700	699	40	male	3	rent	little	rich	1905
702	701	46	male	2	free	little	little	6331
703	702	47	female	2	free	moderate	rich	1377
704	703	41	male	2	own	moderate	moderate	2503
705	704	32	female	2	own	little	moderate	2528
707	706	24	male	2	own	moderate	moderate	6560
708	707	25	female	2	rent	little	moderate	2969
709	708	25	female	2	own	little	moderate	1206
710	709	37	male	1	own	little	moderate	2118
712	711	35	female	2	free	little	little	1198
714	713	25	male	1	own	little	little	1138
715	714	27	male	3	own	little	moderate	14027
720	719	31	male	2	own	moderate	moderate	6148
721	720	34	male	3	own	little	rich	1337
722	721	24	female	2	rent	rich	moderate	433
723	722	24	female	1	own	little	little	1228
724	723	66	female	1	own	quite rich	moderate	790
728	727	25	female	2	rent	little	little	1882

729	728	59	female	2	rent	little	moderate	6416
730	729	36	male	2	own	rich	rich	1275
731	730	33	male	2	own	little	moderate	6403
732	731	21	male	1	rent	little	little	1987
733	732	44	female	1	own	little	moderate	760
737	736	23	female	3	rent	little	moderate	11560
738	737	35	male	1	own	moderate	little	4380
740	739	26	female	1	rent	moderate	moderate	4280
741	740	32	male	2	own	moderate	little	2325
742	741	23	male	1	own	little	moderate	1048
744	743	22	male	2	own	quite rich	little	2483
746	745	28	male	1	own	little	little	1797
747	746	23	female	2	rent	little	little	2511
748	747	37	female	1	own	little	little	1274
751	750	49	female	2	own	little	little	428
752	751	23	female	1	own	little	little	976
753	752	23	female	1	rent	moderate	moderate	841
757	756	74	male	0	own	little	rich	1299
760	759	35	male	2	own	little	little	691
762	761	24	female	2	rent	little	little	2124
763	762	24	male	1	own	little	little	2214
766	765	40	male	1	own	little	moderate	1155
767	766	31	male	1	own	little	little	3108
769	768	28	male	2	rent	little	moderate	3617
772	771	25	female	3	own	little	little	8065
775	774	66	male	0	free	quite rich	rich	1480
778	777	25	female	2	own	little	little	3509
780	779	67	female	2	own	little	moderate	3872
781	780	25	male	2	own	little	moderate	4933
783	782	31	male	1	own	little	moderate	1410
784	783	23	female	1	own	moderate	moderate	836
786	785	35	male	1	own	rich	moderate	1941
789	788	50	male	2	free	little	moderate	6224
790	789	27	male	2	own	little	little	5998
791	790	39	female	2	own	little	moderate	1188
794	793	51	male	2	free	little	rich	2892
802	801	48	female	1	rent	little	moderate	1795
803	802	24	female	2	own	little	little	4272
806	805	24	male	2	own	little	little	9271
807	806	26	male	1	own	little	moderate	590
809	808	55	male	3	free	little	moderate	9283
810	809	26	female	0	rent	little	moderate	1778
811	810	26	male	2	own	little	moderate	907
812	811	28	male	1	own	little	moderate	484
813	812	24	male	2	own	little	little	9629
814	813	54	male	2	own	little	little	3051
815	814	46	male	2	free	little	little	3931
816	815	54	female	2	rent	little	moderate	7432
819	818	43	male	3	own	little	little	15857
820	819	26	male	2	own	little	little	1345
822	821	24	male	2	own	little	rich	3016
823	822	41	male	2	own	little	little	2712
824	823	47	male	1	own	little	little	731
826	825	30	male	2	own	little	little	1602

827	826	33	female	2	rent	little	little	3966
832	831	23	female	2	rent	little	little	1216
833	832	29	male	2	rent	little	little	11816
835	834	25	female	1	own	little	rich	2327
836	835	48	male	2	own	little	little	1082
839	838	63	male	2	own	little	little	2957
841	840	29	male	2	own	little	little	5179
849	848	59	male	2	own	little	little	1364
850	849	57	male	1	own	little	little	709
851	850	33	male	2	rent	little	little	2235
854	853	32	male	1	free	little	little	1442
859	858	29	female	2	own	little	little	3959
863	862	35	female	2	own	little	little	2439
867	866	27	female	2	own	little	little	2389
870	869	24	female	2	rent	little	little	652
872	871	46	male	2	own	little	rich	1343
873	872	26	male	2	own	moderate	little	1382
875	874	29	male	1	own	little	little	3590
876	875	40	female	2	own	rich	moderate	1322
877	876	36	male	3	free	little	little	1940
879	878	27	male	3	free	little	little	1422
885	884	43	male	2	own	little	moderate	4057
886	885	53	female	2	own	little	little	795
888	887	23	male	2	own	little	moderate	15672
891	890	43	male	3	own	little	little	2442
893	892	38	male	1	own	little	little	2171
894	893	34	male	2	own	little	moderate	5800
897	896	28	female	3	rent	little	little	2606
900	899	42	male	2	own	little	little	4153
901	900	43	male	2	rent	little	little	2625
906	905	20	male	3	rent	little	little	1107
912	911	25	female	1	own	little	moderate	4736
915	914	31	male	2	rent	little	little	3161
916	915	32	female	3	own	little	moderate	18424
918	917	68	male	3	own	little	little	14896
919	918	33	male	2	own	moderate	little	2359
920	919	39	male	3	rent	little	little	3345
923	922	22	female	2	rent	little	little	1366
924	923	30	male	2	rent	little	moderate	2002
925	924	55	male	2	own	little	little	6872
926	925	46	male	2	own	little	little	697
927	926	21	female	2	rent	little	little	1049
928	927	39	male	2	free	little	little	10297
930	929	43	male	1	own	little	little	1344
931	930	24	male	1	own	little	little	1747
932	931	22	female	2	own	little	moderate	1670
935	934	23	female	2	own	little	little	1498
936	935	30	male	3	own	moderate	moderate	1919
937	936	28	female	1	own	little	rich	745
938	937	30	male	3	rent	little	moderate	2063
939	938	42	male	2	free	little	moderate	6288
945	944	46	female	2	rent	little	little	1845
946	945	30	female	2	own	quite rich	moderate	8358
947	946	30	male	2	free	quite rich	little	3349

951	950	40	male	0	own	little	moderate	3590
952	951	24	male	2	own	little	little	2145
953	952	28	female	2	rent	quite rich	moderate	4113
955	954	29	female	2	own	little	little	1893
956	955	57	female	3	rent	rich	little	1231
958	957	37	male	1	own	little	moderate	1154
959	958	45	male	1	own	little	little	4006
960	959	30	male	2	free	moderate	moderate	3069
962	961	47	male	2	own	little	moderate	2353
965	964	22	male	1	own	little	moderate	454
967	966	23	male	1	own	quite rich	moderate	2520
970	969	40	male	1	own	little	little	3939
971	970	22	male	2	own	moderate	moderate	1514
973	972	29	female	0	rent	little	little	1193
974	973	36	male	2	rent	little	little	7297
976	975	57	female	1	own	quite rich	rich	1258
977	976	64	female	2	own	little	moderate	753
980	979	25	male	2	rent	moderate	moderate	1264
981	980	49	male	2	own	little	moderate	8386
983	982	28	female	3	own	moderate	rich	2923
984	983	26	male	2	own	little	little	8229
986	985	25	female	2	rent	little	little	1433
987	986	33	male	2	own	little	rich	6289
989	988	29	male	3	free	little	little	6579
990	989	48	male	1	own	little	moderate	1743
994	993	30	male	3	own	little	little	3959
997	996	40	male	3	own	little	little	3857
999	998	23	male	2	free	little	little	1845
1000	999	27	male	2	own	moderate	moderate	4576

	Duration	Purpose
2	48	radio/TV
4	42	furniture/equipment
5	24	car
8	36	car
10	30	car
11	12	car
12	48	business
13	12	radio/TV
14	24	car
15	15	car
16	24	radio/TV
19	24	car
22	6	radio/TV
23	10	car
24	12	car
26	6	furniture/equipment
28	12	radio/TV
29	7	radio/TV
30	60	business
31	18	business
32	24	furniture/equipment
33	18	car
35	12	furniture/equipment
36	45	radio/TV

```
38         18            radio/TV
39         10 domestic appliances
40          9            radio/TV
42         12            radio/TV
43         18             repairs
44         30                 car
45         48                 car
48          6                 car
52         27                 car
55         36                 car
59         18                 car
60         36 furniture/equipment
61          9            business
63         36            business
64         48            business
68         12                 car
73          8     vacation/others
74         42            business
76         12                 car
77         42            radio/TV
78         11            radio/TV
80         30 furniture/equipment
84         24     vacation/others
85         10            radio/TV
87         18 furniture/equipment
88         36           education
89         18                 car
90         12             repairs
92         12                 car
95         12                 car
96         54            business
98         18            business
99         36            radio/TV
102        36            radio/TV
104         9 furniture/equipment
106        24     vacation/others
108        12                 car
110        14            business
111         6            business
112        15           education
113        18                 car
115        12            radio/TV
119        33 furniture/equipment
120        12                 car
121        21            radio/TV
124        10                 car
126        12                 car
127        12            radio/TV
128        12             repairs
129        12                 car
130        12                 car
132        36           education
138        12            radio/TV
140        12            radio/TV
```

141	6	car
142	36	radio/TV
143	27	radio/TV
144	18	furniture/equipment
146	48	business
147	6	car
149	36	furniture/equipment
153	36	radio/TV
154	24	car
155	24	business
156	12	furniture/equipment
157	9	repairs
158	12	education
159	24	car
164	10	car
167	18	furniture/equipment
168	11	furniture/equipment
170	24	business
171	15	car
173	24	furniture/equipment
174	8	radio/TV
175	21	education
177	12	furniture/equipment
178	6	radio/TV
180	21	car
182	36	business
185	18	car
187	9	car
188	16	car
189	12	radio/TV
190	18	furniture/equipment
192	48	business
193	27	business
195	45	radio/TV
196	9	education
198	12	furniture/equipment
200	18	furniture/equipment
202	12	car
204	12	education
206	30	car
208	12	domestic appliances
209	24	business
213	27	business
214	30	business
217	18	business
218	36	radio/TV
219	24	furniture/equipment
221	12	radio/TV
227	48	radio/TV
228	12	furniture/equipment
230	24	furniture/equipment
231	36	radio/TV
234	18	radio/TV
236	24	radio/TV

238	21	business
240	30	radio/TV
243	48	car
249	24	furniture/equipment
251	6	car
252	12	furniture/equipment
253	30	car
258	12	radio/TV
261	12	furniture/equipment
262	24	radio/TV
263	18	car
266	15	radio/TV
269	14	car
274	48	radio/TV
275	30	repairs
285	24	car
286	47	car
287	48	car
288	48	vacation/others
289	12	radio/TV
290	24	radio/TV
292	36	car
293	24	car
294	42	car
296	48	furniture/equipment
300	21	furniture/equipment
302	36	radio/TV
304	10	car
308	12	radio/TV
309	8	furniture/equipment
310	9	car
313	24	furniture/equipment
314	12	car
316	36	furniture/equipment
317	12	furniture/equipment
320	15	furniture/equipment
321	30	car
322	24	radio/TV
323	24	car
324	18	furniture/equipment
326	8	car
329	36	radio/TV
330	6	radio/TV
331	24	car
333	60	car
335	24	furniture/equipment
336	6	furniture/equipment
337	13	radio/TV
339	24	furniture/equipment
340	10	furniture/equipment
341	24	education
342	21	furniture/equipment
343	18	radio/TV
344	18	business

345	10	car
347	13	radio/TV
348	24	radio/TV
350	9	education
352	9	furniture/equipment
354	12	radio/TV
356	24	car
360	30	furniture/equipment
363	12	car
365	18	furniture/equipment
368	18	furniture/equipment
369	36	furniture/equipment
370	18	furniture/equipment
375	60	vacation/others
376	48	business
379	36	car
382	18	car
384	12	car
388	18	furniture/equipment
389	15	business
392	12	furniture/equipment
393	36	car
394	6	radio/TV
396	39	education
397	12	furniture/equipment
398	36	furniture/equipment
399	12	car
406	24	radio/TV
408	15	radio/TV
410	12	car
411	24	radio/TV
417	12	car
423	12	car
426	18	car
430	18	repairs
432	24	vacation/others
433	6	furniture/equipment
435	9	furniture/equipment
439	42	repairs
440	12	business
442	12	furniture/equipment
443	20	vacation/others
445	48	furniture/equipment
447	36	car
448	7	radio/TV
450	15	repairs
455	24	car
457	11	car
458	12	car
459	6	domestic appliances
461	36	furniture/equipment
462	15	car
463	12	furniture/equipment
466	24	car

467	24	radio/TV
471	24	radio/TV
472	6	education
473	9	car
475	18	radio/TV
476	18	furniture/equipment
478	24	radio/TV
479	12	business
480	15	furniture/equipment
481	12	radio/TV
482	24	car
483	30	furniture/equipment
486	6	car
492	27	business
495	12	car
497	36	furniture/equipment
499	18	radio/TV
500	6	car
501	24	car
502	36	car
503	9	radio/TV
504	24	radio/TV
505	24	car
507	15	car
508	15	car
511	12	car
513	15	business
514	12	radio/TV
516	6	car
517	6	car
519	6	car
522	18	radio/TV
523	48	furniture/equipment
525	18	radio/TV
526	26	car
529	36	radio/TV
530	6	car
531	36	education
532	15	car
536	21	education
538	18	furniture/equipment
539	48	car
540	18	furniture/equipment
541	12	radio/TV
544	18	furniture/equipment
546	24	car
549	12	radio/TV
553	48	radio/TV
554	12	car
555	9	education
556	12	radio/TV
557	18	car
559	24	furniture/equipment
560	18	furniture/equipment

562	24	radio/TV
563	6	radio/TV
566	24	radio/TV
567	12	car
570	48	radio/TV
571	24	furniture/equipment
574	15	business
575	9	radio/TV
577	12	radio/TV
579	36	car
581	18	car
582	12	car
584	36	repairs
586	18	furniture/equipment
587	9	car
588	12	furniture/equipment
589	18	domestic appliances
590	12	furniture/equipment
591	12	radio/TV
594	24	car
596	6	car
597	24	car
598	24	business
601	7	radio/TV
602	9	furniture/equipment
603	24	education
605	10	furniture/equipment
606	24	furniture/equipment
608	36	radio/TV
611	12	domestic appliances
612	10	car
613	21	radio/TV
614	24	car
618	6	car
619	30	furniture/equipment
621	21	business
624	12	furniture/equipment
625	18	radio/TV
627	6	furniture/equipment
628	9	car
631	24	radio/TV
632	18	furniture/equipment
635	24	car
640	42	radio/TV
641	18	education
642	15	repairs
645	18	radio/TV
647	30	furniture/equipment
649	24	car
650	12	education
651	48	education
652	12	furniture/equipment
653	24	car
654	36	car

656	14	car
657	12	car
659	30	business
660	18	furniture/equipment
661	12	radio/TV
664	6	furniture/equipment
665	6	education
667	30	furniture/equipment
669	12	car
670	30	radio/TV
678	72	radio/TV
679	24	radio/TV
685	36	business
688	36	car
690	12	car
691	15	furniture/equipment
692	15	repairs
693	24	radio/TV
697	12	radio/TV
700	15	education
702	48	car
703	24	radio/TV
704	30	business
705	27	business
707	48	car
708	12	furniture/equipment
709	9	radio/TV
710	9	radio/TV
712	6	education
714	9	radio/TV
715	60	car
720	20	car
721	9	radio/TV
722	6	education
723	12	car
724	9	radio/TV
728	18	radio/TV
729	48	business
730	24	business
731	24	radio/TV
732	24	radio/TV
733	8	radio/TV
737	24	car
738	18	car
740	30	business
741	24	car
742	10	radio/TV
744	24	furniture/equipment
746	13	business
747	15	car
748	12	car
751	6	furniture/equipment
752	18	car
753	12	business

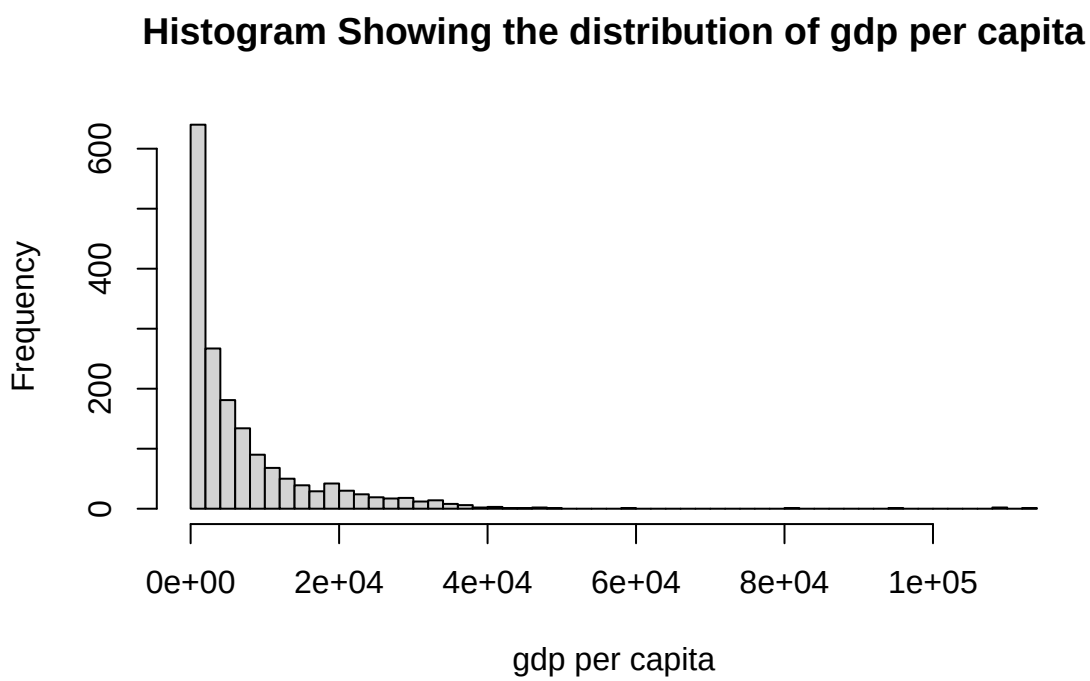
757	6	car
760	12	car
762	18	furniture/equipment
763	12	radio/TV
766	12	radio/TV
767	30	furniture/equipment
769	12	furniture/equipment
772	36	education
775	12	car
778	18	radio/TV
780	18	repairs
781	39	radio/TV
783	12	education
784	12	car
786	18	business
789	48	education
790	40	education
791	21	business
794	24	furniture/equipment
802	18	radio/TV
803	20	furniture/equipment
806	36	car
807	6	radio/TV
809	42	car
810	15	car
811	8	business
812	6	radio/TV
813	36	car
814	48	domestic appliances
815	48	car
816	36	car
819	36	vacation/others
820	18	radio/TV
822	12	radio/TV
823	36	furniture/equipment
824	8	car
826	21	car
827	18	car
832	18	car
833	45	business
835	15	radio/TV
836	12	car
839	24	car
841	36	furniture/equipment
849	9	radio/TV
850	12	radio/TV
851	20	car
854	18	car
859	15	car
863	24	radio/TV
867	18	radio/TV
870	12	furniture/equipment
872	6	car
873	24	business

875	12	furniture/equipment
876	11	car
877	18	radio/TV
879	9	car
885	24	furniture/equipment
886	12	education
888	48	business
891	27	business
893	12	car
894	36	car
897	21	radio/TV
900	18	furniture/equipment
901	16	car
906	12	radio/TV
912	24	furniture/equipment
915	24	business
916	48	vacation/others
918	6	car
919	24	furniture/equipment
920	24	furniture/equipment
923	9	radio/TV
924	12	car
925	24	furniture/equipment
926	12	car
927	18	furniture/equipment
928	48	car
930	12	car
931	24	furniture/equipment
932	9	radio/TV
935	12	radio/TV
936	30	radio/TV
937	9	radio/TV
938	6	radio/TV
939	60	education
945	15	furniture/equipment
946	48	car
947	24	furniture/equipment
951	18	business
952	36	business
953	24	car
955	12	car
956	24	radio/TV
958	9	radio/TV
959	28	car
960	24	furniture/equipment
962	21	car
965	6	repairs
967	27	radio/TV
970	11	car
971	15	repairs
973	24	car
974	60	business
976	24	radio/TV
977	6	radio/TV

```
980        15                  car
981        30 furniture/equipment
983        21                  car
984        36                  car
986        15 furniture/equipment
987        42             business
989        24                  car
990        24             radio/TV
994        36 furniture/equipment
997        30                  car
999        45             radio/TV
1000       45                  car
```

Data Visualization

```
hist(data$gdp_cap,
     breaks = 45,
     xlab = "gdp per capita",
     ylab = "Frequency",
     main = "Histogram Showing the distribution of gdp per capita")
```



Histogram

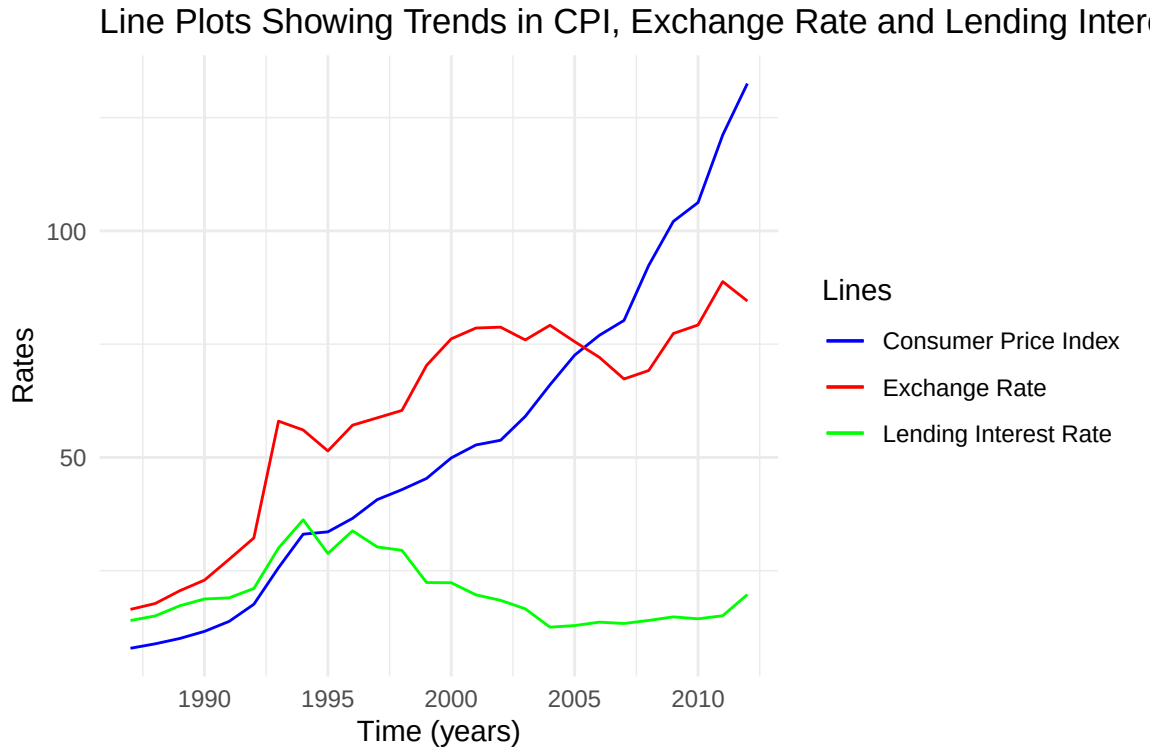
Line Chart

```r
### Import the data
line_plot <- read.csv("training model.csv")
attach(line_plot)
head(line_plot,5)
```

```
  year       CPI Exch.Rate Lend.Int.Rates
1 1987  7.872727  16.45499        14.0000
2 1988  8.848083  17.74710        15.0000
3 1989 10.035029  20.57247        17.2500
4 1990 11.602322  22.91477        18.7500
5 1991 13.805882  27.50870        18.9975
```

Multiple Line plot

```r
library(ggplot2)
ggplot(data = line_plot, aes(x = year)) +
  geom_line(aes(y = CPI, color = "Consumer Price Index")) +
  geom_line(aes(y = Exch.Rate, color = "Exchange Rate")) +
  geom_line(aes(y = Lend.Int.Rates, color = "Lending Interest Rate")) +
  labs(x = "Time (years)", y = "Rates", color = "Lines") +
  scale_color_manual(values = c("blue", "red", "green")) +
  ggtitle("Line Plots Showing Trends in CPI, Exchange Rate and Lending Interest Rates") +
  theme_minimal()
```



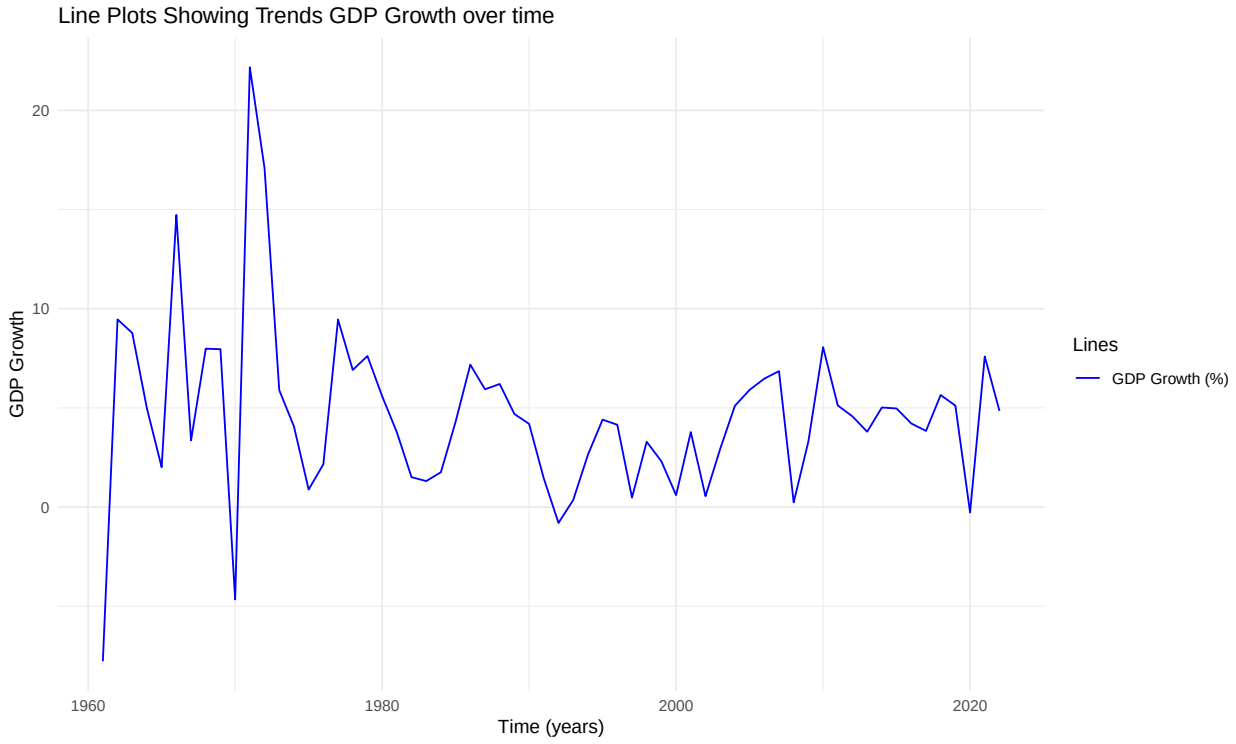
```r
gdp_growth <- read.csv("gdp_growth.csv")
attach(gdp_growth)
head(gdp_growth,5)
```

```
  year GDP.growth..annual...
1 1961             -7.774635
2 1962              9.457359
3 1963              8.778340
4 1964              4.964467
5 1965              2.009094
```

```r
tail(gdp_growth,5)
```

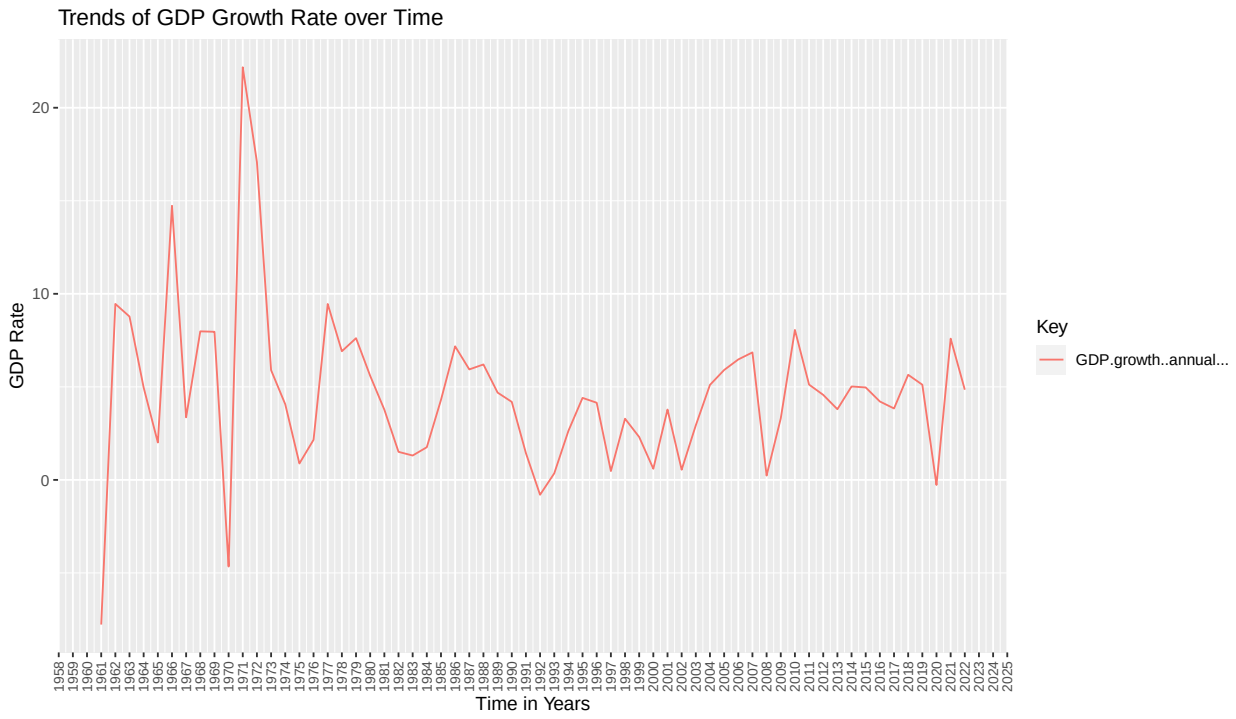
```
   year GDP.growth..annual...
58 2018             5.6479464
59 2019             5.1141589
60 2020            -0.2727663
61 2021             7.5904895
62 2022             4.8466349
```

```r
ggplot(data = gdp_growth, aes(x = year)) +
  geom_line(aes(y = GDP.growth..annual..., color = "GDP Growth (%)")) +
  labs(x = "Time (years)", y = "GDP Growth", color = "Lines") +
  scale_color_manual(values = c("blue")) +
  ggtitle("Line Plots Showing Trends GDP Growth over time") +
  theme_minimal()
```



Alternative Good Looking Plot

```
date<-seq(as.Date("1961-01-01"),by="1 year",length.out=length(gdp_growth$year))
ggplot(data=gdp_growth,aes(x=date))+
  geom_line(aes(y=GDP.growth..annual...,colour="GDP.growth..annual..."))+
  labs(title="Trends of GDP Growth Rate over Time",
       caption="", y="GDP Rate", x="Time in Years", color="Key")+
  scale_x_date( date_labels = "%Y", breaks = "1 year")+
  theme(axis.text.x = element_text(angle = 90, vjust=0.5, size = 8))
```



Additional Chart

```
data <- read.csv("Gapminder.csv")
head(data,5)
```

```
      country year population continent life_exp  gdp_cap ln_population
1 Afghanistan 1952    8425333      Asia   28.801 779.4453      6.925587
2 Afghanistan 1957    9240934      Asia   30.332 820.8530      6.965716
3 Afghanistan 1962   10267083      Asia   31.997 853.1007      7.011447
4 Afghanistan 1967   11537966      Asia   34.020 836.1971      7.062129
5 Afghanistan 1972   13079460      Asia   36.088 739.9811      7.116590
  ln_life_exp ln_gdpPercap
1    1.459408     6.658583
2    1.481901     6.710344
3    1.505109     6.748878
4    1.531734     6.728864
5    1.557363     6.606625
```

```
attach(data)
```

Leaf and Stem Plot

```
stem(life_exp)
```

```
  The decimal point is 1 digit(s) to the right of the |

  2 | 4
  2 | 9
  3 | 000011222223333344444444444
  3 | 55555555566666666666677777777777777777788888888888888888999999999999
  4 | 00000000000000000000000000000011111111111111111111111111111112222222+90
  4 | 55555555555555555555555555555555555555555555566666666666666666666666+110
  5 | 00000000000000000000000000000000000000011111111111111111111111111111+94
  5 | 55555555555555555555555555566666666666666666666666666666666666667777+87
  6 | 00000000000000000000000000000000000000011111111111111111111111111112+79
  6 | 55555555555555555555555555555566666666666666666666666666666666666677+123
  7 | 00000000000000000000000000000000000000000000000000000000000000000000+251
  7 | 55555555555555555555555555555555555555555555566666666666666666666666+92
  8 | 00000000000011111111111122223
```

This kind of a chart is not appropriate for a lage data set. Consider the chart below.

```
ages <- rnorm(200, mean = 45, sd = 14)
stem(ages)
```

```
  The decimal point is 1 digit(s) to the right of the |

  1 | 113
  1 | 7
  2 | 111124
  2 | 566688
  3 | 0000011112222333334
  3 | 555556667788889999
  4 | 00000011112222222233333344444444
  4 | 55555556666666677777888899999999999
  5 | 00011111222233334444
  5 | 55555666667777888888899999999
  6 | 0111112223334
  6 | 667777889999
  7 | 033
  7 | 5
```

A scatter plot with Regression Equation

```r
income <- read.csv("income.csv")
attach(income)
head(income,5)
```

```
  Year Income Consumption
1 1950  791.8       733.2
2 1951  819.0       748.7
3 1952  844.3       771.4
4 1953  880.0       802.5
5 1954  894.0       822.7
```

```r
library(ggpmisc)
library(ggplot2)
ggplot(data = income, aes(x = Income, y = Consumption)) +
  stat_poly_line() +
  stat_poly_eq(eq.with.lhs = "italic(hat(y))~`=`~",
               use_label(c("eq", "R2"))) +
  ggtitle("A scatter plot of Income and Consumption") +
  geom_point()
```

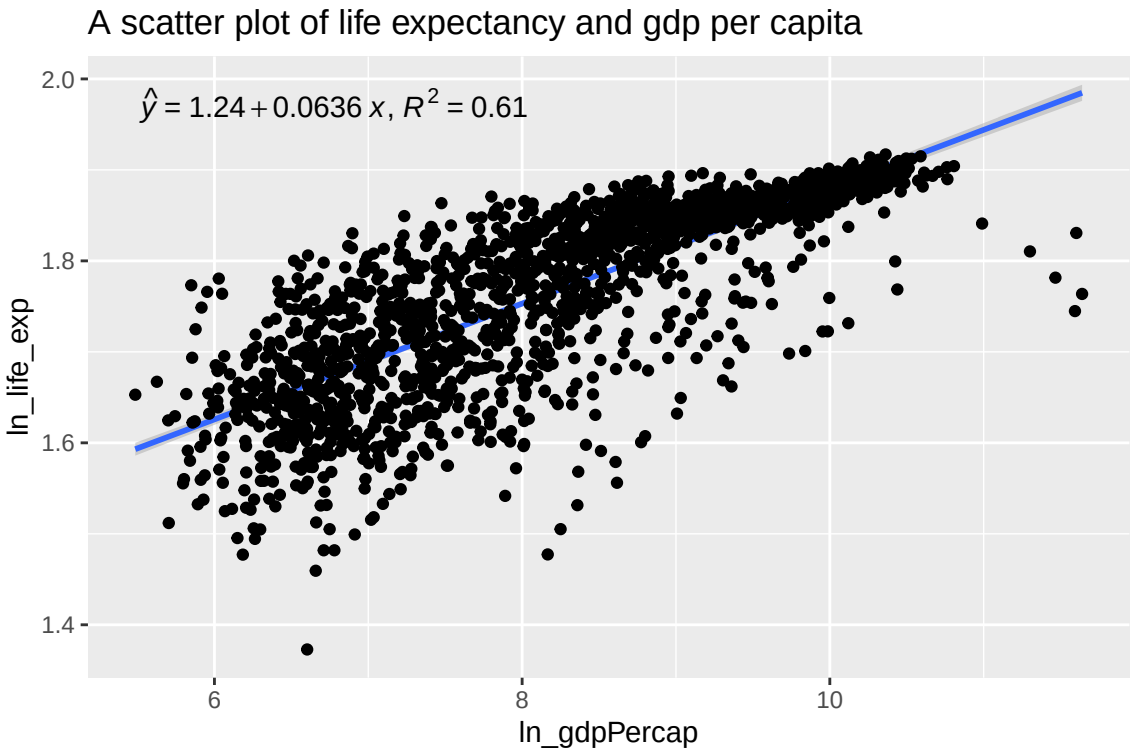
Additional Scatter Plot

```r
data <- read.csv("Gapminder.csv")
head(data,5)
```

```
      country year population continent life_exp  gdp_cap ln_population
1 Afghanistan 1952    8425333      Asia   28.801 779.4453      6.925587
2 Afghanistan 1957    9240934      Asia   30.332 820.8530      6.965716
3 Afghanistan 1962   10267083      Asia   31.997 853.1007      7.011447
4 Afghanistan 1967   11537966      Asia   34.020 836.1971      7.062129
5 Afghanistan 1972   13079460      Asia   36.088 739.9811      7.116590
  ln_life_exp ln_gdpPercap
1    1.459408     6.658583
2    1.481901     6.710344
3    1.505109     6.748878
4    1.531734     6.728864
5    1.557363     6.606625
```

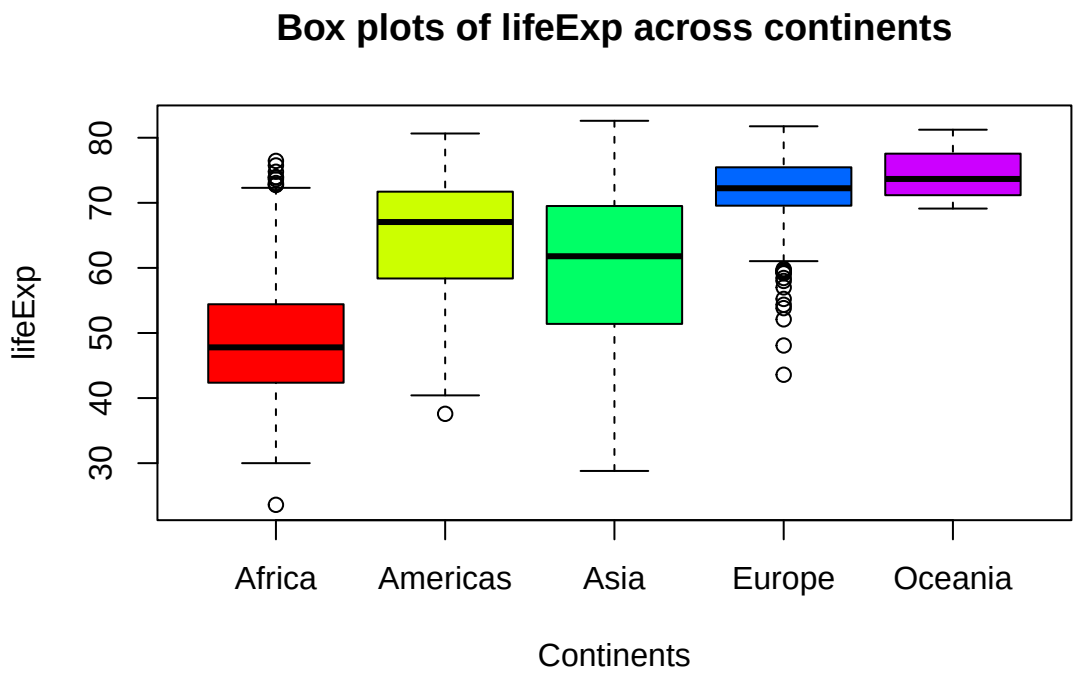
```
attach(data)
```

```
ggplot(data = data, aes(x = ln_gdpPercap, y = ln_life_exp)) +
  stat_poly_line() +
  stat_poly_eq(eq.with.lhs = "italic(hat(y))~`=`~",
               use_label(c("eq", "R2"))) +
  ggtitle("A scatter plot of life expectancy and gdp per capita") +
  geom_point()
```



Box Plot

```
boxplot(life_exp ~ continent, main ="Box plots of lifeExp across continents",
        xlab="Continents",ylab="lifeExp",
        col=rainbow(5))
```



Bar Graph

```
BAR <- read.csv("superstore.csv")
attach(BAR)
head(BAR,5)
```

Load the data

```
  Row_ID       Order_ID Order_Date  Ship_Date      Ship_Mode Customer_ID
1      1 CA-2016-152156 08/11/2016 11/11/2016   Second Class    CG-12520
2      2 CA-2016-152156 08/11/2016 11/11/2016   Second Class    CG-12520
3      3 CA-2016-138688 12/06/2016 16/06/2016   Second Class    DV-13045
4      4 US-2015-108966 11/10/2015 18/10/2015 Standard Class    SO-20335
5      5 US-2015-108966 11/10/2015 18/10/2015 Standard Class    SO-20335
    Customer_Name   Segment       Country            City      State
1     Claire Gute  Consumer United States       Henderson   Kentucky
2     Claire Gute  Consumer United States       Henderson   Kentucky
3 Darrin Van Huff Corporate United States     Los Angeles California
4  Sean O'Donnell  Consumer United States Fort Lauderdale    Florida
5  Sean O'Donnell  Consumer United States Fort Lauderdale    Florida
  Postal_Code Region      Product_ID        Category Sub_Category    Sales
1       42420  South FUR-BO-10001798       Furniture    Bookcases 261.9600
2       42420  South FUR-CH-10000454       Furniture       Chairs 731.9400
3       90036   West OFF-LA-10000240 Office Supplies       Labels  14.6200
4       33311  South FUR-TA-10000577       Furniture       Tables 957.5775
5       33311  South OFF-ST-10000760 Office Supplies      Storage  22.3680
```

```
  Quantity Discount    Profit
1        2     0.00   41.9136
2        3     0.00  219.5820
3        2     0.00    6.8714
4        5     0.45 -383.0310
5        2     0.20    2.5164
```

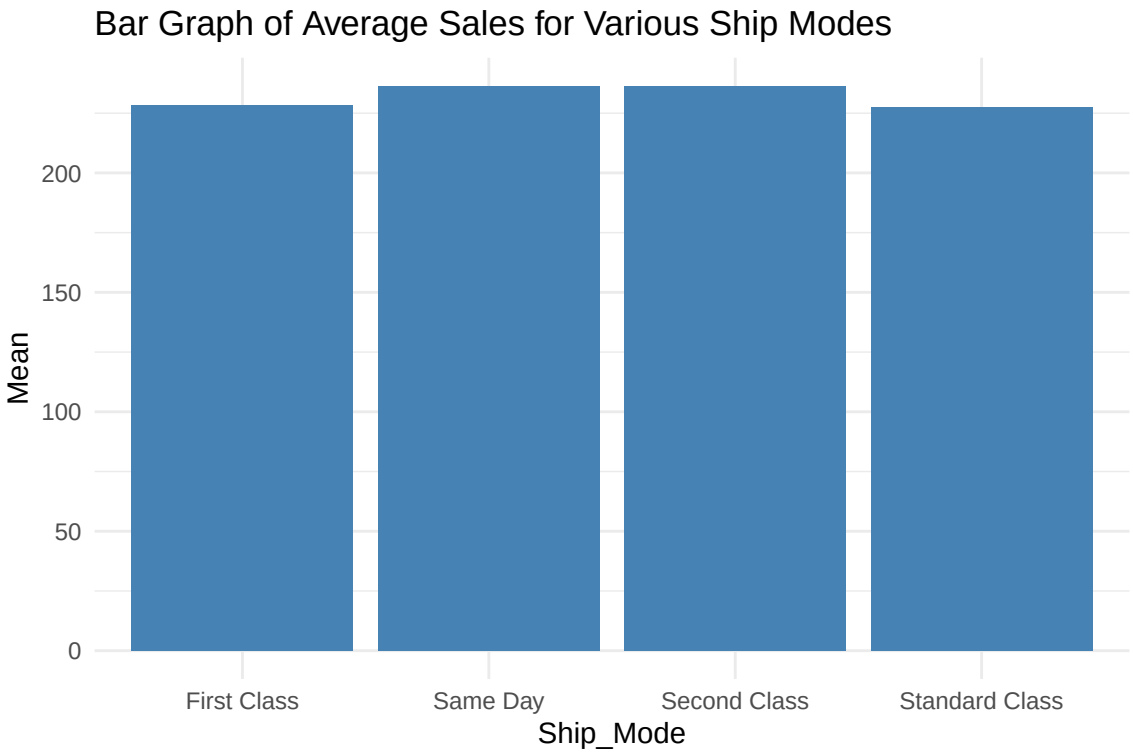
Create Grouped Summaries

```
library(tidyverse)
library(ggpubr)
library(rstatix)

SUM <- BAR %>%
  group_by(Ship_Mode) %>%
  get_summary_stats(Sales, type = "mean_sd")
SUM
```

```
# A tibble: 4 x 5
  Ship_Mode      variable     n  mean    sd
  <chr>          <fct>    <dbl> <dbl> <dbl>
1 First Class    Sales     1538  228.  630.
2 Same Day       Sales      543  236.  555.
3 Second Class   Sales     1945  236.  559.
4 Standard Class Sales     5968  228.  647.
```

Create the Bar Graph

```
ggplot(data = SUM, aes(x = Ship_Mode, y = mean)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(x = "Ship_Mode", y = "Mean") +
  ggtitle("Bar Graph of Average Sales for Various Ship Modes") +
  theme_minimal()
```



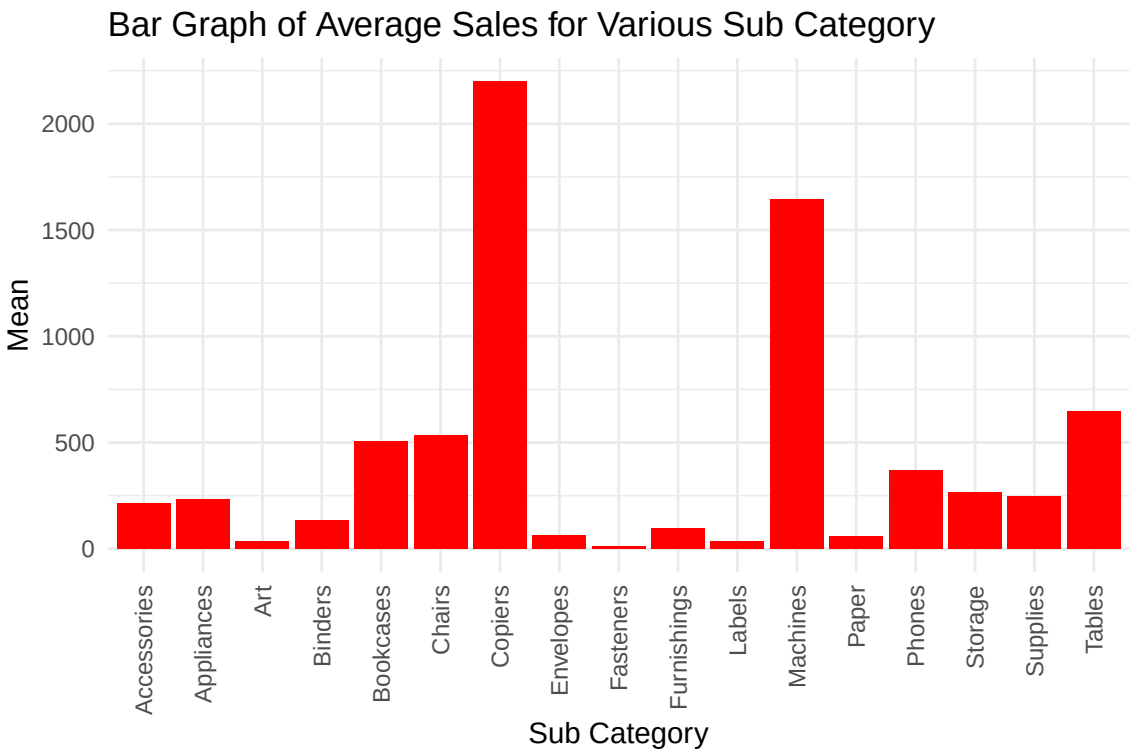
Additional Bar Graph

```
SUM2 <- BAR %>%
  group_by(Sub_Category) %>%
  get_summary_stats(Sales, type = "mean_sd")
SUM2
```

```
# A tibble: 17 x 5
   Sub_Category variable     n   mean     sd
   <chr>        <fct>    <dbl>  <dbl>  <dbl>
 1 Accessories  Sales      775  216.   335.
 2 Appliances   Sales      466  231.   389.
 3 Art          Sales      796   34.1   60.1
 4 Binders      Sales     1523  134.   563.
 5 Bookcases    Sales      228  504.   639.
 6 Chairs       Sales      617  532.   550.
 7 Copiers      Sales       68 2199.  3176.
 8 Envelopes    Sales      254   64.9   84.4
 9 Fasteners    Sales      217   13.9   12.4
10 Furnishings  Sales      957   95.8  148.
11 Labels       Sales      364   34.3   74.1
12 Machines     Sales      115 1646.  2765.
13 Paper        Sales     1370   57.3   78.2
14 Phones       Sales      889  371.   491.
15 Storage      Sales      846  265.   355.
16 Supplies     Sales      190  246.   924.
17 Tables       Sales      319  649.   616.
```

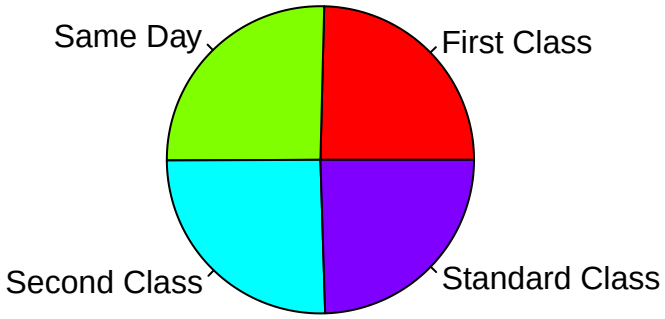
Run the Command below to Create the bar graph of mean sales across Sub categories

```
ggplot(data = SUM2, aes(x = Sub_Category, y = mean)) +
  geom_bar(stat = "identity", fill = "red") +
  labs(x = "Sub Category", y = "Mean") +
  ggtitle("Bar Graph of Average Sales for Various Sub Category") +
  theme_minimal()+
    theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
```



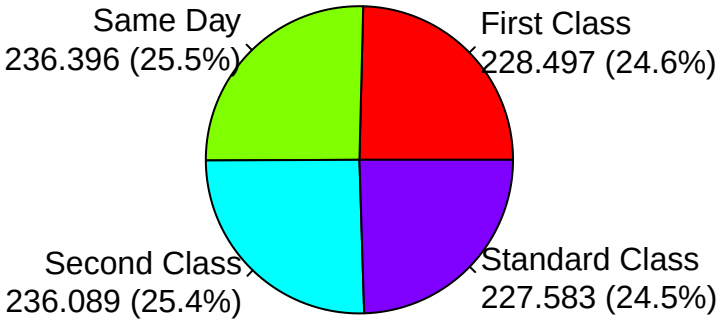
Pie Chart

```
pie(SUM$mean, labels = SUM$Ship_Mode, col = rainbow(length(SUM$mean)))
```



Create pie chart with values and percentages

```
# Calculate the percentages
percentages <- SUM$mean / sum(SUM$mean) * 100

# Create the pie chart with values and percentages
pie(SUM$mean, labels = paste(SUM$Ship_Mode, "\n", SUM$mean, " (", round(percentages, 1), "%)", sep = ""
```



Testing the Normality of the Data

Shapiro wilks statistics

```
data <- read.csv("Gapminder.csv")
head(data,5)
```

```
      country year population continent life_exp  gdp_cap ln_population
1 Afghanistan 1952    8425333      Asia   28.801 779.4453      6.925587
2 Afghanistan 1957    9240934      Asia   30.332 820.8530      6.965716
3 Afghanistan 1962   10267083      Asia   31.997 853.1007      7.011447
4 Afghanistan 1967   11537966      Asia   34.020 836.1971      7.062129
5 Afghanistan 1972   13079460      Asia   36.088 739.9811      7.116590
  ln_life_exp ln_gdpPercap
1    1.459408     6.658583
2    1.481901     6.710344
3    1.505109     6.748878
4    1.531734     6.728864
5    1.557363     6.606625
```

```
attach(data)
```

Perform the test

```r
shapiro.test(gdp_cap)
```

```
    Shapiro-Wilk normality test

data:  gdp_cap
W = 0.6522, p-value < 2.2e-16
```

The Shapiro-Wilk test was performed on the variable gdp_cap. The test result shows a test statistic (W) of 0.6522 and an extremely small p-value (p-value < 2.2e-16). Since the p-value is less than 0.05 (assuming a common significance level), we reject the null hypothesis that the data follows a normal distribution. The extremely small p-value indicates strong evidence against the normality assumption for the gdp_cap dataset.

Run the test for normally distributed data

```r
scores <- rnorm(200, mean = 45, sd = 14)
scores
```

```
  [1] 44.773831 76.014737 42.800109 43.028250 57.532969 28.213615 55.592696
  [8] 53.490589 26.397558 37.824664 45.982286 54.155798 51.183025 20.604243
 [15] 27.859735 46.937388 46.994639 32.007853 17.748024 47.613488 44.971213
 [22] 45.157252 37.200328 72.273202 44.680341 49.114480 50.740056 26.630562
 [29] 63.513562 66.151740 58.365691 17.809217 34.163292 31.333628 49.359174
 [36] 64.932463 55.905637 51.303499 48.664345 55.205506 48.696355 32.394111
 [43] 59.890085  9.996272 52.430079 43.950810 54.530212 34.717261 52.792264
 [50] 51.650674 64.123219 80.882407 22.636552 55.868840 65.301023 71.240595
 [57] 45.448806 50.779385 56.609568 24.283275 64.786857 47.121191 63.402067
 [64] 44.815056 19.965055 52.155276 14.629236 42.466027  9.200883 49.169596
 [71] 44.551581 57.967761 61.609053 29.186706 37.763450 43.940805 47.600813
 [78] 36.795391 31.092958 52.359156 60.293293 56.979859 43.628457 37.451507
 [85] 11.548828 35.112209 56.212560 35.799179 47.639265 72.103454 40.426035
 [92] 31.669941 47.805739 55.935274 43.128029 55.516280 61.531067 55.766257
 [99] 45.450164 58.298843 42.466298 35.179916 65.742715 42.174264 33.305385
[106] 65.843710 24.450410 42.932408 56.239263 19.633297 41.418840 38.677036
[113] 50.322373 42.504846 51.710055 31.878707 38.298305 52.423603 59.836253
[120] 35.620124 70.205631  6.429107 85.841389 51.808253 41.679194 25.143050
[127] 40.687743 74.783295 45.425002 46.368226 63.337875 16.144619 36.211278
[134] 46.828178 52.712981 16.501505 55.391863 62.316399 64.365432 61.738711
[141] 46.397833 32.827172 33.892504 51.177124 39.448513 47.492462 35.440339
[148] 32.562827 27.658847 37.686859 78.425924 18.962484 38.871342 43.381676
[155] 64.411880 43.017736 37.128855 45.696545 46.058901 34.388914 63.405383
[162] 60.470534 38.958918 43.287056 38.654235 54.076633 37.187517 33.690742
[169] 15.665697 53.108267 45.091503 52.314297 55.993255 53.312113 36.650116
[176] 27.391712 27.360018 43.338011 52.037510 37.444421 22.562548 35.849669
[183] 55.136060 61.123512 47.394195 17.240694 58.455752 38.222722 53.827806
[190] 32.983612 58.986609 57.645086 62.100358 56.428984 40.590749 36.553855
[197]  3.331607 50.738524 30.132436 55.659810
```

```r
scores <- data.frame(scores)
scores
```

```
       scores
1   44.773831
2   76.014737
3   42.800109
4   43.028250
5   57.532969
6   28.213615
7   55.592696
8   53.490589
9   26.397558
10  37.824664
11  45.982286
12  54.155798
13  51.183025
14  20.604243
15  27.859735
16  46.937388
17  46.994639
18  32.007853
19  17.748024
20  47.613488
21  44.971213
22  45.157252
23  37.200328
24  72.273202
25  44.680341
26  49.114480
27  50.740056
28  26.630562
29  63.513562
30  66.151740
31  58.365691
32  17.809217
33  34.163292
34  31.333628
35  49.359174
36  64.932463
37  55.905637
38  51.303499
39  48.664345
40  55.205506
41  48.696355
42  32.394111
43  59.890085
44   9.996272
45  52.430079
46  43.950810
47  54.530212
48  34.717261
49  52.792264
```

```
50  51.650674
51  64.123219
52  80.882407
53  22.636552
54  55.868840
55  65.301023
56  71.240595
57  45.448806
58  50.779385
59  56.609568
60  24.283275
61  64.786857
62  47.121191
63  63.402067
64  44.815056
65  19.965055
66  52.155276
67  14.629236
68  42.466027
69   9.200883
70  49.169596
71  44.551581
72  57.967761
73  61.609053
74  29.186706
75  37.763450
76  43.940805
77  47.600813
78  36.795391
79  31.092958
80  52.359156
81  60.293293
82  56.979859
83  43.628457
84  37.451507
85  11.548828
86  35.112209
87  56.212560
88  35.799179
89  47.639265
90  72.103454
91  40.426035
92  31.669941
93  47.805739
94  55.935274
95  43.128029
96  55.516280
97  61.531067
98  55.766257
99  45.450164
100 58.298843
101 42.466298
102 35.179916
103 65.742715
```

104 42.174264
105 33.305385
106 65.843710
107 24.450410
108 42.932408
109 56.239263
110 19.633297
111 41.418840
112 38.677036
113 50.322373
114 42.504846
115 51.710055
116 31.878707
117 38.298305
118 52.423603
119 59.836253
120 35.620124
121 70.205631
122 6.429107
123 85.841389
124 51.808253
125 41.679194
126 25.143050
127 40.687743
128 74.783295
129 45.425002
130 46.368226
131 63.337875
132 16.144619
133 36.211278
134 46.828178
135 52.712981
136 16.501505
137 55.391863
138 62.316399
139 64.365432
140 61.738711
141 46.397833
142 32.827172
143 33.892504
144 51.177124
145 39.448513
146 47.492462
147 35.440339
148 32.562827
149 27.658847
150 37.686859
151 78.425924
152 18.962484
153 38.871342
154 43.381676
155 64.411880
156 43.017736
157 37.128855

```
158 45.696545
159 46.058901
160 34.388914
161 63.405383
162 60.470534
163 38.958918
164 43.287056
165 38.654235
166 54.076633
167 37.187517
168 33.690742
169 15.665697
170 53.108267
171 45.091503
172 52.314297
173 55.993255
174 53.312113
175 36.650116
176 27.391712
177 27.360018
178 43.338011
179 52.037510
180 37.444421
181 22.562548
182 35.849669
183 55.136060
184 61.123512
185 47.394195
186 17.240694
187 58.455752
188 38.222722
189 53.827806
190 32.983612
191 58.986609
192 57.645086
193 62.100358
194 56.428984
195 40.590749
196 36.553855
197  3.331607
198 50.738524
199 30.132436
200 55.659810
```

```
## Rename the dataset
random <-scores
head(random,5)
```

```
    scores
1 44.77383
2 76.01474
3 42.80011
4 43.02825
5 57.53297
```

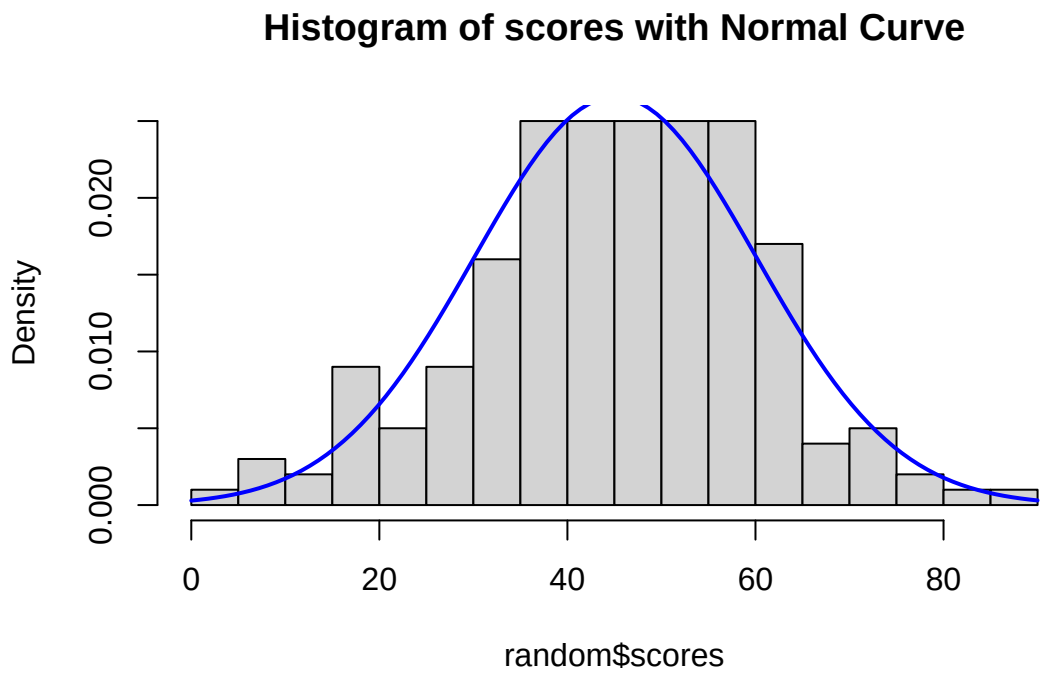
```r
attach(random)
```

```r
shapiro.test(random$scores)
```

```
    Shapiro-Wilk normality test

data:  random$scores
W = 0.99227, p-value = 0.3719
```

Make a histogram to Visualize the results

```r
### Make a histogram
hist(random$scores, breaks = 14, prob = TRUE, main = "Histogram of scores with Normal Curve")
curve(dnorm(x, mean = mean(random$scores), sd = sd(random$scores)), add = TRUE, col = "blue", lwd = 2)
```



Parametric Test

```r
t.test(scores, mu=40, alternative = "greater", conf.level = 0.90)
```

One sample T-test

```
    One Sample t-test

data:  scores
t = 4.8014, df = 199, p-value = 1.546e-06
alternative hypothesis: true mean is greater than 40
90 percent confidence interval:
 43.73141      Inf
sample estimates:
mean of x
 45.09618
```

Interpretation!!! (student's part)

Two Samples T-test

```
weight_before <- c(75,90,78,65,78,89,87,65,67,78,45,67,67,87,90)
weight_after <- c(73,85,70,59,72,90,81,60,64,71,39,69,73,82,83)
```

Dependent t-test

Data frame the dataset

```
paired <- data.frame(weight_before, weight_after)
head(paired,5)
```

```
  weight_before weight_after
1            75           73
2            90           85
3            78           70
4            65           59
5            78           72
```

Perform the test

```
t.test(weight_before, weight_after, alternative = "two.sided", paired = T, var.equal = T, conf.level = (
```

```
    Paired t-test

data:  weight_before and weight_after
t = 3.7262, df = 14, p-value = 0.002257
alternative hypothesis: true mean difference is not equal to 0
99 percent confidence interval:
 0.7642042 6.8357958
sample estimates:
mean difference
            3.8
```

Student's part!!!

Unpaired T-test/ Independent t-test

```
library(magrittr)
library(gapminder)
attach(gapminder)
```

```
df1 <- gapminder %>%
  dplyr::select(country, lifeExp, year)%>%
  filter(country == "Kenya"|
           country == "Tanzania")

head(df1,14)
```

```
# A tibble: 14 x 3
   country  lifeExp  year
   <fct>      <dbl> <int>
 1 Kenya       42.3  1952
 2 Kenya       44.7  1957
 3 Kenya       47.9  1962
 4 Kenya       50.7  1967
 5 Kenya       53.6  1972
 6 Kenya       56.2  1977
 7 Kenya       58.8  1982
 8 Kenya       59.3  1987
 9 Kenya       59.3  1992
10 Kenya       54.4  1997
11 Kenya       51.0  2002
12 Kenya       54.1  2007
13 Tanzania    41.2  1952
14 Tanzania    43.0  1957
```

```
tail(df1,14)
```

```
# A tibble: 14 x 3
   country  lifeExp  year
   <fct>      <dbl> <int>
 1 Kenya       51.0  2002
 2 Kenya       54.1  2007
 3 Tanzania    41.2  1952
 4 Tanzania    43.0  1957
 5 Tanzania    44.2  1962
 6 Tanzania    45.8  1967
 7 Tanzania    47.6  1972
 8 Tanzania    49.9  1977
 9 Tanzania    50.6  1982
10 Tanzania    51.5  1987
11 Tanzania    50.4  1992
12 Tanzania    48.5  1997
13 Tanzania    49.7  2002
14 Tanzania    52.5  2007
```

```r
t.test(data = df1, lifeExp ~ country, alternative = "greater", conf.level = 0.99, var.equal = F)
```

```
    Welch Two Sample t-test

data:  lifeExp by country
t = 2.482, df = 18.78, p-value = 0.01135
alternative hypothesis: true difference in means between group Kenya and group Tanzania is greater than
99 percent confidence interval:
 -0.115541       Inf
sample estimates:
   mean in group Kenya mean in group Tanzania
              52.68100               47.91233
```

```r
df2 <- gapminder %>%
  dplyr::select(country, gdpPercap, year)%>%
  filter(country == "Kenya"|
           country == "Tanzania")

head(df2,14)
```

```
# A tibble: 14 x 3
   country  gdpPercap  year
   <fct>        <dbl> <int>
 1 Kenya         854.  1952
 2 Kenya         944.  1957
 3 Kenya         897.  1962
 4 Kenya        1057.  1967
 5 Kenya        1222.  1972
 6 Kenya        1268.  1977
 7 Kenya        1348.  1982
 8 Kenya        1362.  1987
 9 Kenya        1342.  1992
10 Kenya        1360.  1997
11 Kenya        1288.  2002
12 Kenya        1463.  2007
13 Tanzania      717.  1952
14 Tanzania      699.  1957
```

```r
tail(df2,14)
```

```
# A tibble: 14 x 3
   country  gdpPercap  year
   <fct>        <dbl> <int>
 1 Kenya        1288.  2002
 2 Kenya        1463.  2007
 3 Tanzania      717.  1952
 4 Tanzania      699.  1957
 5 Tanzania      722.  1962
 6 Tanzania      848.  1967
 7 Tanzania      916.  1972
 8 Tanzania      962.  1977
```

```
 9 Tanzania      874.  1982
10 Tanzania      832.  1987
11 Tanzania      826.  1992
12 Tanzania      789.  1997
13 Tanzania      899.  2002
14 Tanzania     1107.  2007
```

```
t.test(data = df2, gdpPercap ~ country, alternative = "greater", conf.level = 0.99, var.equal = F)
```

```
    Welch Two Sample t-test

data:  gdpPercap by country
t = 5.1159, df = 17.262, p-value = 4.106e-05
alternative hypothesis: true difference in means between group Kenya and group Tanzania is greater than
99 percent confidence interval:
 175.2233      Inf
sample estimates:
   mean in group Kenya mean in group Tanzania
             1200.4157               849.2813
```

One-way ANOVA

```
head(gapminder,5)
```

```
# A tibble: 5 x 6
  country     continent  year lifeExp      pop gdpPercap
  <fct>       <fct>     <int>   <dbl>    <int>     <dbl>
1 Afghanistan Asia       1952    28.8  8425333      779.
2 Afghanistan Asia       1957    30.3  9240934      821.
3 Afghanistan Asia       1962    32.0 10267083      853.
4 Afghanistan Asia       1967    34.0 11537966      836.
5 Afghanistan Asia       1972    36.1 13079460      740.
```

```
results <- aov(lifeExp~continent)
summary(results)
```

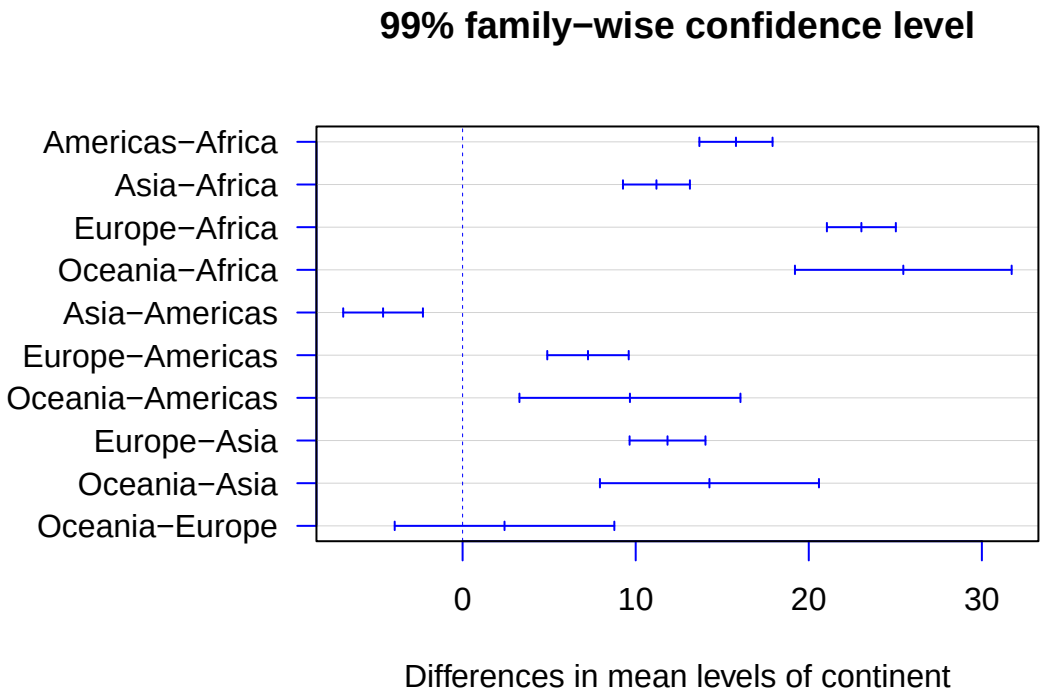
```
             Df Sum Sq Mean Sq F value Pr(>F)
continent     4 139343   34836   408.7 <2e-16 ***
Residuals  1699 144805      85
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

Post Hoc Analysis

```
library(agricolae)
TKy <- TukeyHSD(results)
TKy
```

```
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = lifeExp ~ continent)

$continent
                       diff       lwr       upr     p adj
Americas-Africa  15.793407 14.022263 17.564550 0.0000000
Asia-Africa      11.199573  9.579887 12.819259 0.0000000
Europe-Africa    23.038356 21.369862 24.706850 0.0000000
Oceania-Africa   25.460878 20.216908 30.704848 0.0000000
Asia-Americas    -4.593833 -6.523432 -2.664235 0.0000000
Europe-Americas   7.244949  5.274203  9.215696 0.0000000
Oceania-Americas  9.667472  4.319650 15.015293 0.0000086
Europe-Asia      11.838783 10.002952 13.674614 0.0000000
Oceania-Asia     14.261305  8.961718 19.560892 0.0000000
Oceania-Europe    2.422522 -2.892185  7.737230 0.7250559
```

```r
par(oma=c(0,5,0,0)) # adjust the margins because the factor names are long
plot(TukeyHSD(results, conf.level = 0.99),las=1, col = "blue")
```



Two-way ANOVA

```r
ANOVA <- read.csv("yields.csv")
attach(ANOVA)
head(ANOVA,5)
```

```
  S.No   Yields Blocks Fertilizer.Used
1    1 54.40207      1               1
2    2 41.00511      1               1
3    3 38.92977      1               1
4    4 54.74481      1               1
5    5 46.72406      1               1
```

Create Factors and Levels

```
ANOVA$Fertilizer.Used<-factor(ANOVA$Fertilizer.Used, levels = c(1,2,3,4),
                              labels = c("DAP", "NPK", "AMONNIA","PHOSPHATE"))
ANOVA$Blocks <- factor(ANOVA$Blocks, levels = c(1,2,3,4),
                       labels = c("Block1", "Block2","Block3","Block4"))
```

View the dataset

```
head(ANOVA,5)
```

```
  S.No   Yields Blocks Fertilizer.Used
1    1 54.40207 Block1             DAP
2    2 41.00511 Block1             DAP
3    3 38.92977 Block1             DAP
4    4 54.74481 Block1             DAP
5    5 46.72406 Block1             DAP
```

```
View(ANOVA)
```

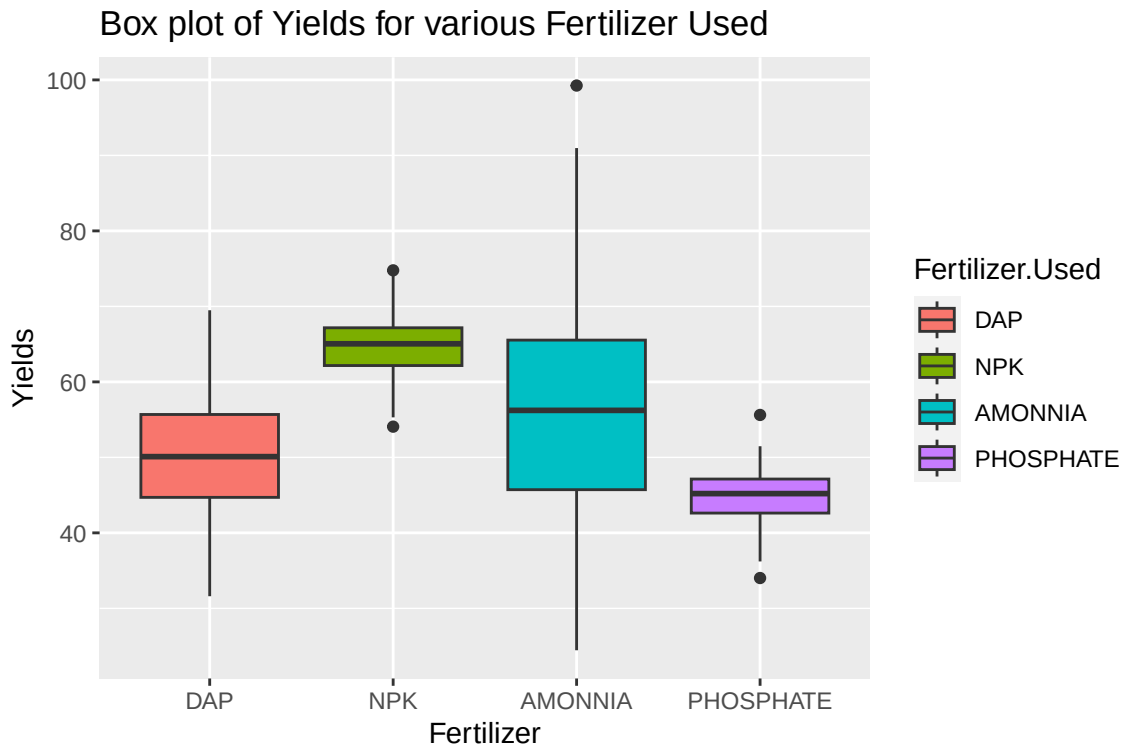
Get the grouped summary Statistics for Yields across fertilizer used

```
grouped_summaries <- ANOVA %>%
  group_by(Fertilizer.Used) %>%
  get_summary_stats(Yields, type = "mean_sd")
grouped_summaries
```

```
# A tibble: 4 x 5
  Fertilizer.Used variable     n  mean    sd
  <fct>           <fct>    <dbl> <dbl> <dbl>
1 DAP             Yields     100  50.4  7.27
2 NPK             Yields     100  65.1  4.07
3 AMONNIA         Yields     100  56.1 14.0
4 PHOSPHATE       Yields     100  44.8  3.8
```

Box plot of Yields for Various Fertilizer Used

```
ggplot(ANOVA, aes(x = Fertilizer.Used, y = Yields, fill = Fertilizer.Used)) +
  labs(title = "Box plot of Yields for various Fertilizer Used", y = "Yields", x = "Fertilizer")+
  geom_boxplot()
```



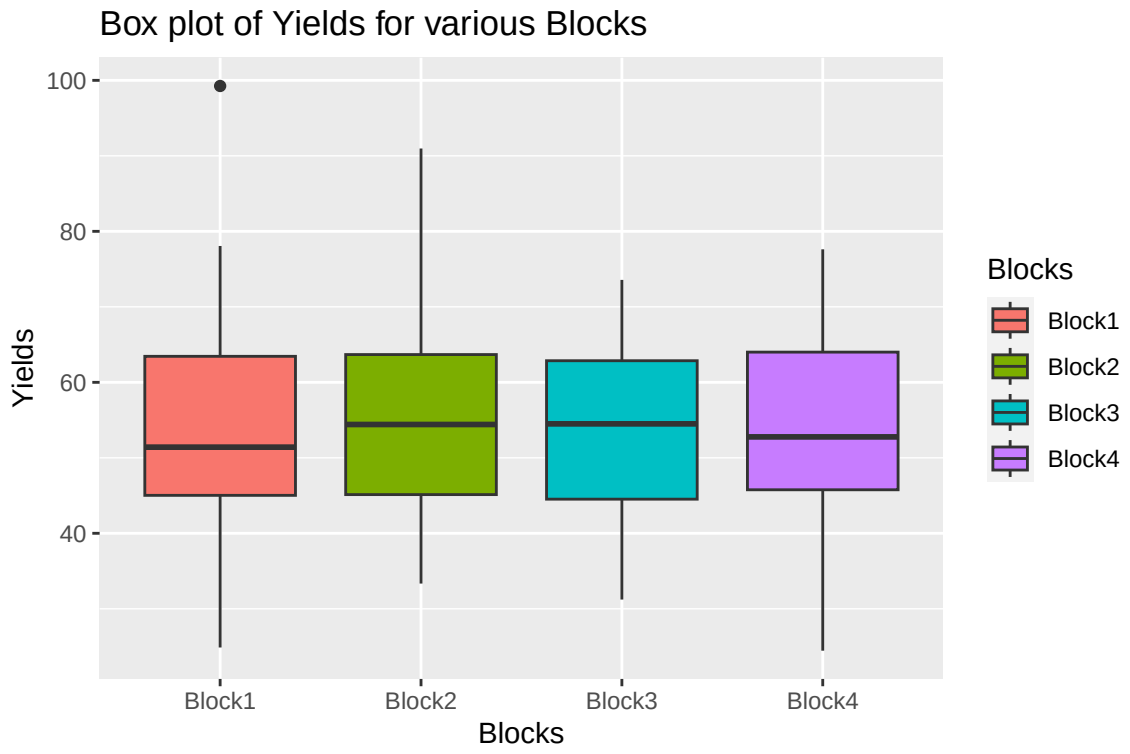
Get the summary statistics for Yields across blocks

```
grouped_summaries2 <- ANOVA %>%
  group_by(Blocks) %>%
  get_summary_stats(Yields, type = "mean_sd")
grouped_summaries2
```

```
# A tibble: 4 x 5
  Blocks variable     n  mean    sd
  <fct>  <fct>    <dbl> <dbl> <dbl>
1 Block1 Yields     100  53.9  11.6
2 Block2 Yields     100  54.8  11.2
3 Block3 Yields     100  53.8  10.8
4 Block4 Yields     100  54.0  11.4
```

Box plot of Yields for Various Blocks

```
ggplot(ANOVA, aes(x = Blocks, y = Yields, fill = Blocks)) +
  labs(title = "Box plot of Yields for various Blocks", y = "Yields", x = "Blocks")+
  geom_boxplot()
```



Estimate the linear Model and Extract the ANOVA Results

```
linear_model <- lm(Yields ~ Fertilizer.Used+Blocks, data = ANOVA)
my_model <- anova(linear_model)
my_model
```

```
Analysis of Variance Table

Response: Yields
                Df  Sum Sq Mean Sq  F value Pr(>F)
Fertilizer.Used   3 22433.1  7477.7 106.1659 <2e-16 ***
Blocks            3    72.6    24.2   0.3436 0.7938
Residuals       393 27680.6    70.4
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

NON_PARAMETRIC TESTS

Wilcoxon Signed-Rank Test:

The Wilcoxon signed-rank test is used to compare paired or related samples. It assesses whether there is a significant difference between the measurements or observations taken from the same subjects or units under two different conditions or time points. The test is appropriate for data that are not normally distributed or when the assumption of normality is violated. #### Create the dataset

```r
before <- c(87,98,77,89,90,100,110,89,77,68,65,67,87,98,67,87)
after <- c(80,90,68,77,81,95,102,77,69,60,59,60,79,88,63,80)
```

Data Frame the data

```r
weight_frame <- data.frame(before,after)
head(weight_frame,5)
```

```
  before after
1     87    80
2     98    90
3     77    68
4     89    77
5     90    81
```

Group the Data

```r
library(dplyr)
library(tidyverse)
library(ggpubr)
library(rstatix)
weight_frame <- weight_frame %>%
  gather(key = "time", value = "weight", before, after) %>%
  convert_as_factor(time)
head(weight_frame, 5)
```

```
    time weight
1 before     87
2 before     98
3 before     77
4 before     89
5 before     90
```

```r
View(weight_frame)
```

Perform the test

```r
result <- wilcox.test(weight~time, paired = TRUE, data = weight_frame)
result
```

```
    Wilcoxon signed rank test with continuity correction

data:  weight by time
V = 0, p-value = 0.0004556
alternative hypothesis: true location shift is not equal to 0
```

Mann-Whitney U-Test

The Mann-Whitney U test (also called the Wilcoxon rank-sum test) compares two independent groups or conditions to determine if there is a significant difference between their distributions or medians. This test is appropriate when comparing two groups without assuming normality or when the data are ordinal or skewed.

The data

```
head(df1,5)
```

```
# A tibble: 5 x 3
  country lifeExp  year
  <fct>     <dbl> <int>
1 Kenya      42.3  1952
2 Kenya      44.7  1957
3 Kenya      47.9  1962
4 Kenya      50.7  1967
5 Kenya      53.6  1972
```

Perform the Test

```
result1 <- wilcox.test(lifeExp~country, paired = F, data = df1)
result1
```

```
    Wilcoxon rank sum exact test

data:  lifeExp by country
W = 113, p-value = 0.01727
alternative hypothesis: true location shift is not equal to 0
```

Kruskall Wallis

The Kruskal-Wallis test is a nonparametric test used to compare the medians of three or more independent groups. It is an extension of the Mann-Whitney U test (Wilcoxon rank-sum test) for two groups. The Kruskal-Wallis test does not assume that the data are normally distributed and can handle ordinal or non-normally distributed data.

The data

```
library(magrittr)
df23 <- gapminder %>%
  dplyr::select(country, lifeExp)%>%
  filter(country == "Kenya"|
           country == "Morocco"|
           country == "United States"|
```

```
           country == "Afghanistan"|
           country == "Canada")

head(df23,5)
```

```
# A tibble: 5 x 2
  country     lifeExp
  <fct>         <dbl>
1 Afghanistan    28.8
2 Afghanistan    30.3
3 Afghanistan    32.0
4 Afghanistan    34.0
5 Afghanistan    36.1
```

```
View(df23)
```

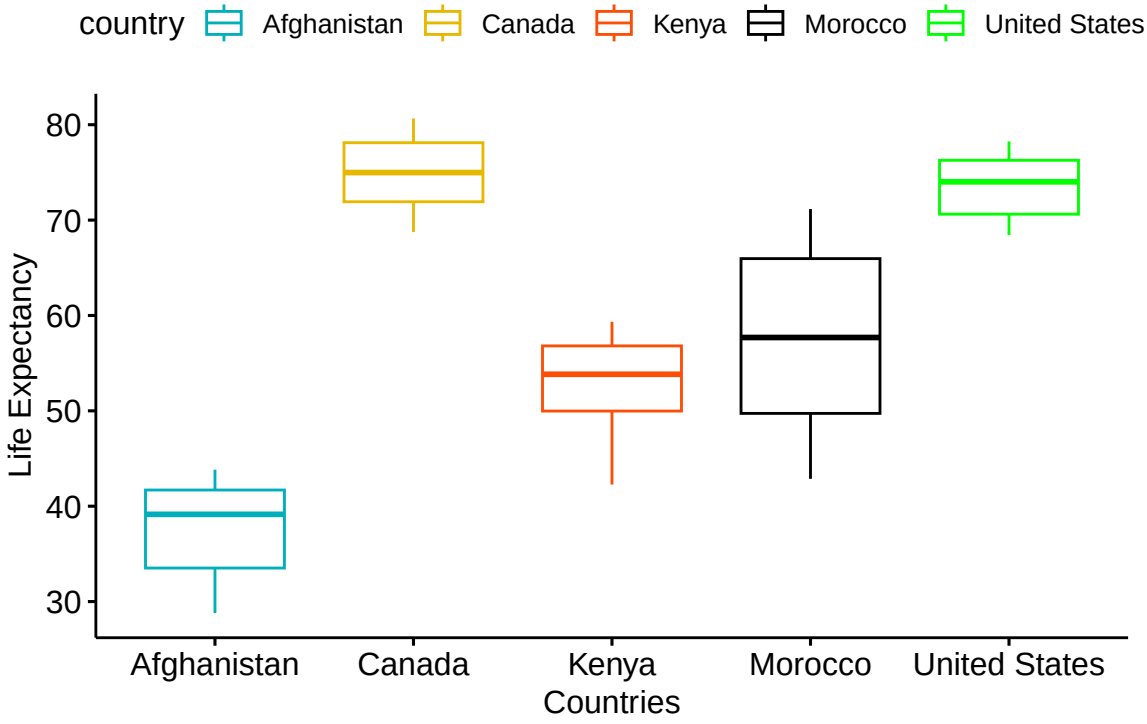
Grouped Summary Statistics

```
library(dplyr)
group_by(df23, country) %>%
  summarise(
    count = n(),
    mean = mean(lifeExp, na.rm = TRUE),
    sd = sd(lifeExp, na.rm = TRUE),
    median = median(lifeExp, na.rm = TRUE),
    max = max(lifeExp, na.rm = TRUE),
    min = min(lifeExp, na.rm = TRUE),
    IQR = IQR(lifeExp, na.rm = TRUE)
  )
```

```
# A tibble: 5 x 8
  country       count  mean    sd median   max   min   IQR
  <fct>         <int> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl>
1 Afghanistan      12  37.5  5.10   39.1  43.8  28.8  8.18
2 Canada           12  74.9  3.95   75.0  80.7  68.8  6.19
3 Kenya            12  52.7  5.60   53.8  59.3  42.3  6.83
4 Morocco          12  57.6  9.81   57.7  71.2  42.9 16.2
5 United States    12  73.5  3.34   74.0  78.2  68.4  5.65
```

Use Box Plot to Visualize the Data

```
library("ggpubr")
ggboxplot(df23, x = "country", y = "lifeExp",
          color = "country", palette = c("#00AFBB", "#E7B800", "#FC4E07","#000000","#00FF00"),
          order = c("Afghanistan", "Canada", "Kenya","Morocco","United States"),
          ylab = "Life Expectancy", xlab = "Countries")
```



Compute Kruskal-Wallis test

We want to see if the median life expectancy in five continents vary significantly. The test can be run using the kruskal.test() function as follows.

The data

```
head(df23,5)
```

```
# A tibble: 5 x 2
  country     lifeExp
  <fct>         <dbl>
1 Afghanistan    28.8
2 Afghanistan    30.3
3 Afghanistan    32.0
4 Afghanistan    34.0
5 Afghanistan    36.1
```

Conduct the Test

```
kruskal.test(lifeExp ~ country, data = df23)
```

```
    Kruskal-Wallis rank sum test
```

```
data:  lifeExp by country
Kruskal-Wallis chi-squared = 49.879, df = 4, p-value = 3.827e-10
```

LINEAR REGRESSION ANALYSIS

Statistical techniques are tools that enable us to answer questions about possible patterns in empirical data. It is not surprising, then, to learn that many important techniques of statistical analysis were developed by scientists who were interested in answering very specific empirical questions. So it was with regression analysis. The history of this particular statistical technique can be traced back to late nineteenth-century England and the pursuits of a gentleman scientist, Francis Galton. Galton was born into a wealthy family that produced more than its share of geniuses; he and Charles Darwin, the famous biologist, were first cousins. During his lifetime, Galton studied everything from fingerprint classification to meteorology, but he gained widespread recognition primarily for his work on inheritance. His most important insight came to him while he was studying the inheritance of one of the most obvious of all human characteristics: height. In order to understand how the characteristic of height was passed from one generation to the next, Galton collected data on the heights of individuals and the heights of their parents. After constructing frequency tables that classified these individuals both by their height and by the average height of their parents, Galton came to the unremarkable conclusion that tall people usually had tall parents and short people usually had short parents.

Assumption of Regression Analysis

1. The error term has a population mean of zero 2. All independent variables are uncorrelated with the error term 3. Observations of the error term are uncorrelated with each other 4. The error term has a constant variance (no heteroscedasticity) 5. No independent variable is a perfect linear function of other explanatory variables 6. The error term is normally distributed (optional)

Load the data

```
mydatta <- read.csv("Unemployment.csv")
attach(mydatta)
head(mydatta,5)
```

```
  year Unemployment Inflation  FedRate
1 1859     5.133333 0.9084719 3.933333
2 1860     5.233333 1.8107772 3.696667
3 1861     5.533333 1.6227203 2.936667
4 1862     6.266667 1.7953352 2.296667
5 1863     6.800000 0.5370330 2.003333
```

To be continued!!!!

Estimate the Model

```
my_model <- lm(log(Inflation)~log(Unemployment)+log(FedRate), data = mydatta)
```

Visualize the Model Using Stargazer Library

```
library(stargazer)
stargazer(my_model, type = "text")
```

```
================================================
                        Dependent variable:
                    ----------------------------
                             log(Inflation)
------------------------------------------------
log(Unemployment)                 0.176
                                 (0.157)

log(FedRate)                     0.976***
                                 (0.085)

Constant                         -0.902***
                                 (0.291)

------------------------------------------------
Observations                       164
R2                                0.473
Adjusted R2                       0.466
Residual Std. Error         0.498 (df = 161)
F Statistic             72.167*** (df = 2; 161)
================================================
Note:                *p<0.1; **p<0.05; ***p<0.01
```

Model Interpretation

The model above presents the results of a regression analysis with the dependent variable "log(Inflation)" and three independent variables: "log(Unemployment)", "log(FedRate)", and the constant term. The coefficients estimated for each independent variable represent the relationship between that variable and the dependent variable while holding other variables constant. The coefficients are accompanied by standard errors in parentheses.

log(Unemployment): The coefficient estimate for log(Unemployment) is 0.176. This means that a one-unit increase in the natural logarithm of unemployment is associated with a 0.176 unit increase in the natural logarithm of inflation. The standard error of 0.157 indicates the uncertainty in this estimate.

log(FedRate): The coefficient estimate for log(FedRate) is 0.976. This suggests that a one-unit increase in the natural logarithm of the Federal Reserve interest rate is associated with a 0.976 unit increase in the natural logarithm of inflation. The standard error of 0.085 provides an indication of the precision of this estimate.

Constant: The constant term in the model is -0.902. This represents the expected value of the natural logarithm of inflation when all independent variables are zero. The standard error of 0.291 reflects the uncertainty in this estimation.

The observations in the dataset used for the analysis are 164. The R-squared value of 0.473 indicates that approximately 47.3% of the variance in the natural logarithm of inflation can be explained by the independent variables included in the model. The adjusted R-squared value of 0.466 on the other hand accounts for the

degrees of freedom in the model and provides a more conservative estimate of the proportion of variance explained.

The residual standard error of 0.498 indicates the average deviation of the observed values of the dependent variable from the predicted values, taking into account the degrees of freedom in the model.

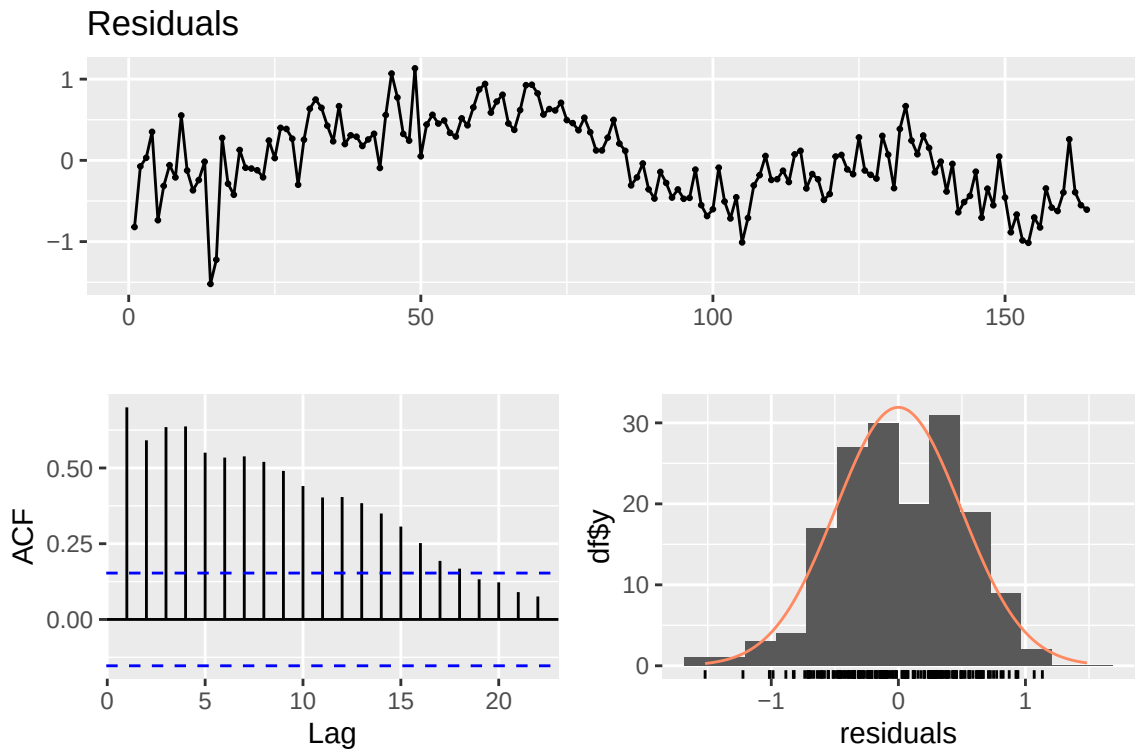
The F-statistic of 72.167, with 2 and 161 degrees of freedom, suggests that the overall model is statistically significant. The associated p-value is less than 0.01, indicating strong evidence against the null hypothesis of no relationship between the independent variables and the dependent variable.

In summary, the model suggests that only log(Unemployment) has a statistically significant relationship with log(Inflation). However, it is important to note that these interpretations are based on the given coefficients, standard errors, and significance levels. Further analysis and consideration of the model's assumptions are necessary for a comprehensive understanding of the relationships between the variables.

Testing the Assumptions

Normality of the error term

```
library(forecast)
checkresiduals(my_model)
```




```
    Breusch-Godfrey test for serial correlation of order up to 10

data:  Residuals
LM test = 98.912, df = 10, p-value < 2.2e-16
```

Zero Conditional Mean

```
ReSid<-resid(my_model)
```

Add the residual variable to the data set

```
mydatta$ReSid <- ReSid
head(mydatta,5)
```

```
  year Unemployment Inflation  FedRate       ReSid
1 1859     5.133333 0.9084719 3.933333 -0.82027544
2 1860     5.233333 1.8107772 3.696667 -0.07333963
3 1861     5.533333 1.6227203 2.936667  0.03190288
4 1862     6.266667 1.7953352 2.296667  0.35104981
5 1863     6.800000 0.5370330 2.003333 -0.73682650
```

View Summary Statistics

Additional Way of Displaying Summary Statistics.

```
### Load the libraries
library("ggplot2")
library("devtools")
library("predict3d")
library("psych")
library("dplyr")
library("gtsummary")
library("DescTools")
library("nortest")
library("lmtest")
library("sandwich")
```

Display the Summary Statistics

```
knitr::kable(
  describeBy(mydatta[,-1]) %>% round(3)
)
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
Unemployment	1	164	5.960	1.510	5.700	5.878	1.557	3.400	10.667	7.267	0.576	0.250	0.118
Inflation	2	164	3.877	2.482	3.207	3.591	2.116	0.349	12.049	11.700	1.005	0.478	0.194
FedRate	3	164	6.590	3.184	5.758	6.214	2.792	1.683	17.780	16.097	1.210	1.576	0.249
ReSid	4	164	0.000	0.494	-0.016	0.006	0.525	-1.521	1.133	2.654	-0.140	-0.320	0.039

The variance covariance assumption

```
cov(ReSid, Unemployment)
```

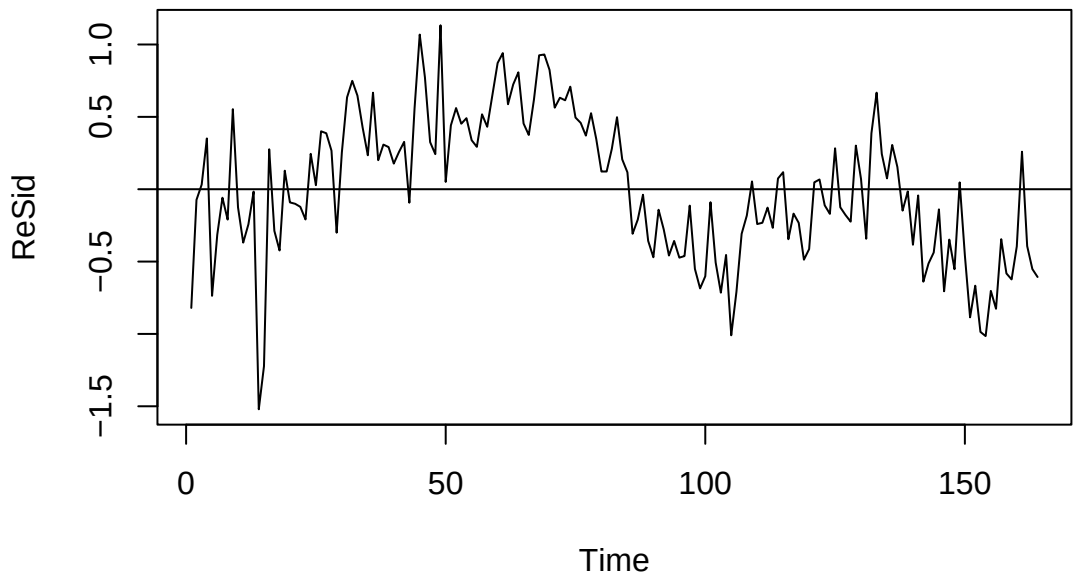
```
[1] 0.002701838
```

```
cov(ReSid, FedRate)
```

```
[1] -0.02217665
```

Plot the Residuals

```
ts.plot(ReSid)
abline(0,0.0000)
```



Multicollinearity

```
library(car)
library(tseries)
vif(my_model)
```

```
log(Unemployment)      log(FedRate)
         1.034369          1.034369
```

Heteroscedasticity

```
ncvTest(my_model)
```

```
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 2.623628, Df = 1, p = 0.10528
```

Since the p-value is greater than the conventional significance level of 0.05, we fail to reject the null hypothesis of constant variance. This means that there is not enough evidence to conclude that the variance of the residuals varies across the range of the predictor variable(s).

Autocorrelation

```
durbinWatsonTest(my_model)
```

```
 lag Autocorrelation D-W Statistic p-value
   1       0.7001071     0.5736665       0
 Alternative hypothesis: rho != 0
```

The Durbin-Watson (D-W) statistic is a test used to detect the presence of autocorrelation in the residuals of a regression model. It measures the degree of correlation between adjacent residuals. The D-W statistic ranges from 0 to 4, with values around 2 indicating no autocorrelation, values below 2 suggesting positive autocorrelation, and values above 2 indicating negative autocorrelation.

Suppose the variance of the Error terms was not homoscedastic

Estimating the Regression Model with Robust Standard errors

Robust standard errors, also known as heteroscedasticity-robust standard errors or White's standard errors, are a method to estimate the standard errors in regression analysis that account for potential heteroscedasticity (unequal variances) in the error terms.

In ordinary least squares (OLS) regression, the standard errors assume that the error terms have constant variance. However, in real-world data, it is common to encounter situations where the variability of the error terms changes across different levels of the independent variables. This violates the assumption of homoscedasticity, leading to incorrect standard error estimates, t-statistics, and p-values.

Robust standard errors address this issue by providing more accurate estimates of the standard errors that are robust to heteroscedasticity. They are calculated by estimating the variance-covariance matrix of the coefficient estimates using methods that do not assume constant variance of the errors.

There are different types of robust standard errors, including the HC1, HC2, and HC3 estimators, which differ in the specific assumptions they make about the structure of heteroscedasticity. These estimators are implemented in the sandwich package in R.

By using robust standard errors, researchers can obtain more reliable inference in regression analysis, particularly when there is evidence or suspicion of heteroscedasticity. Robust standard errors allow for valid hypothesis tests, confidence intervals, and t-statistics, even in the presence of heteroscedasticity, providing more accurate and robust statistical inference.

```r
library(sandwich)
library(lmtest)
robust_se <- sqrt(diag(vcovHC(my_model, type = "HC1")))
robust_se
```

```
      (Intercept) log(Unemployment)      log(FedRate)
       0.27530083        0.14986315        0.07540774
```

View the Model

```r
coeftest(my_model, vcov = vcovHC(my_model, type = "HC1"))
```

```
t test of coefficients:

                   Estimate Std. Error t value  Pr(>|t|)
(Intercept)       -0.901611   0.275301 -3.2750  0.001294 **
log(Unemployment)  0.176483   0.149863  1.1776  0.240683
log(FedRate)       0.976433   0.075408 12.9487 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

LINEAR REGRESSION ANALYSIS

Statistical techniques are tools that enable us to answer questions about possible patterns in empirical data. It is not surprising, then, to learn that many important techniques of statistical analysis were developed by scientists who were interested in answering very specific empirical questions. So it was with regression analysis. The history of this particular statistical technique can be traced back to late nineteenth-century England and the pursuits of a gentleman scientist, Francis Galton. Galton was born into a wealthy family that produced more than its share of geniuses; he and Charles Darwin, the famous biologist, were first cousins. During his lifetime, Galton studied everything from fingerprint classification to meteorology, but he gained widespread recognition primarily for his work on inheritance. His most important insight came to him while he was studying the inheritance of one of the most obvious of all human characteristics: height. In order to understand how the characteristic of height was passed from one generation to the next, Galton collected data on the heights of individuals and the heights of their parents. After constructing frequency tables that classified these individuals both by their height and by the average height of their parents, Galton came to the unremarkable conclusion that tall people usually had tall parents and short people usually had short parents.

Assumption of Regression Analysis

1. The regression model is linear in the coefficients and the error term 2. The error term has a population mean of zero 3. All independent variables are uncorrelated with the error term 4. Observations of the error term are uncorrelated with each other 5. The error term has a constant variance (no heteroscedasticity) 6. No independent variable is a perfect linear function of other explanatory variables 7. The error term is normally distributed (optional)

Load the data

```
mydatta <- read.csv("Unemployment.csv")
attach(mydatta)
head(mydatta,5)
```

```
  year Unemployment Inflation  FedRate
1 1859     5.133333 0.9084719 3.933333
2 1860     5.233333 1.8107772 3.696667
3 1861     5.533333 1.6227203 2.936667
4 1862     6.266667 1.7953352 2.296667
5 1863     6.800000 0.5370330 2.003333
```

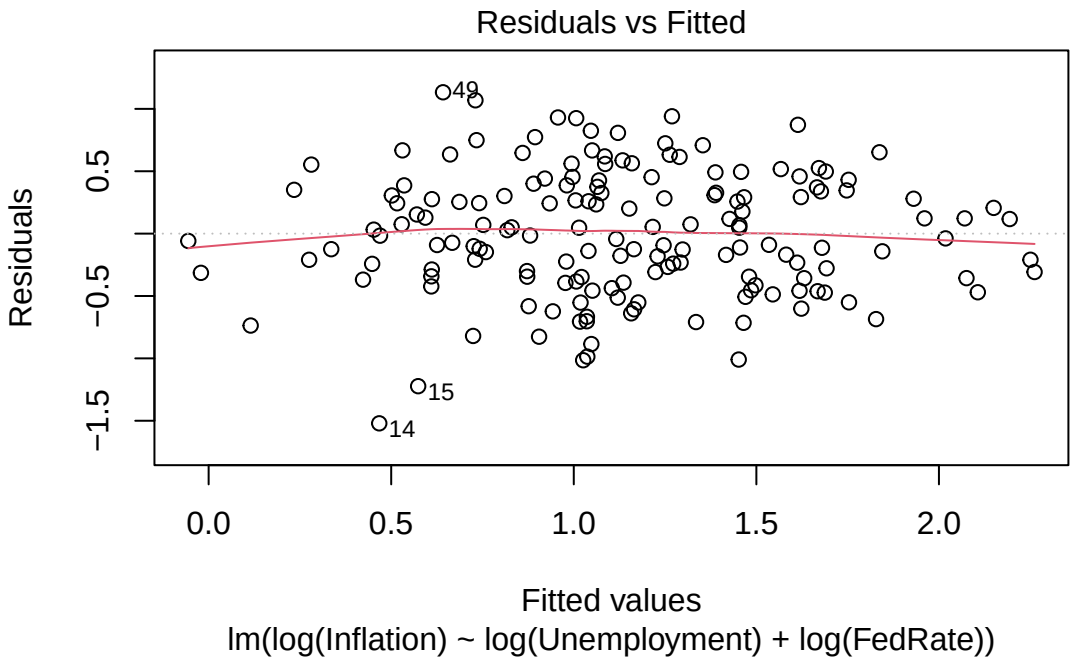
Estimate the Model

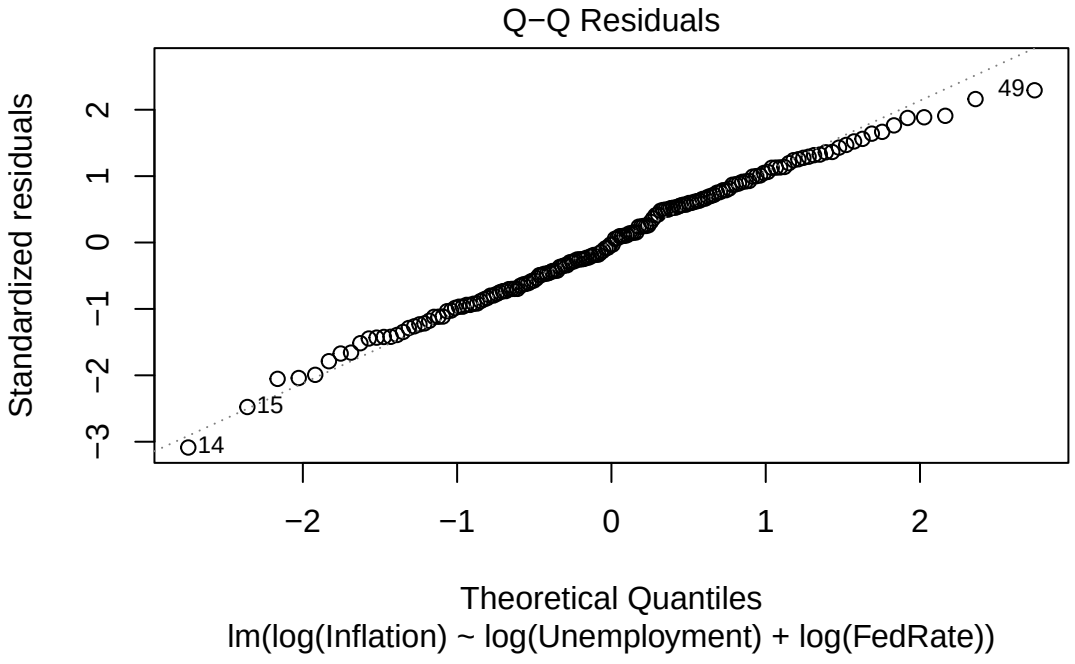
```
my_model <- lm(log(Inflation)~log(Unemployment)+log(FedRate), data = mydatta)
summary(my_model)
```

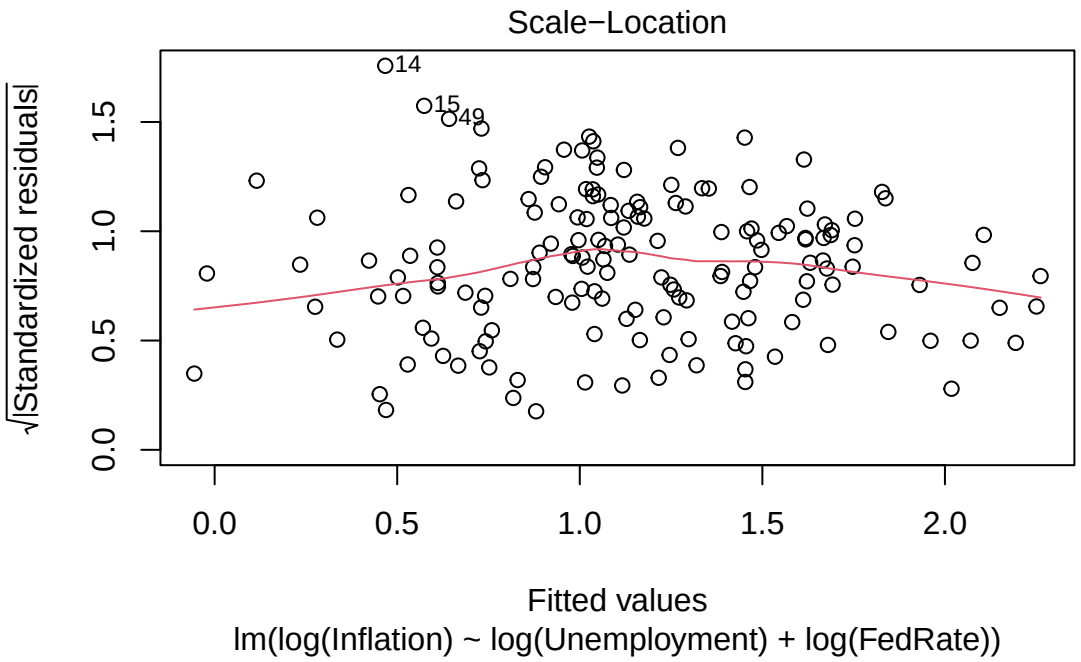
```
Call:
lm(formula = log(Inflation) ~ log(Unemployment) + log(FedRate),
    data = mydatta)

Residuals:
     Min       1Q   Median       3Q      Max
-1.52073 -0.35028 -0.01588  0.35593  1.13292

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)        -0.9016     0.2907  -3.102  0.00227 **
log(Unemployment)   0.1765     0.1567   1.127  0.26161
log(FedRate)        0.9764     0.0845  11.555  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4975 on 161 degrees of freedom
Multiple R-squared:  0.4727,    Adjusted R-squared:  0.4662
F-statistic: 72.17 on 2 and 161 DF,  p-value: < 2.2e-16
```

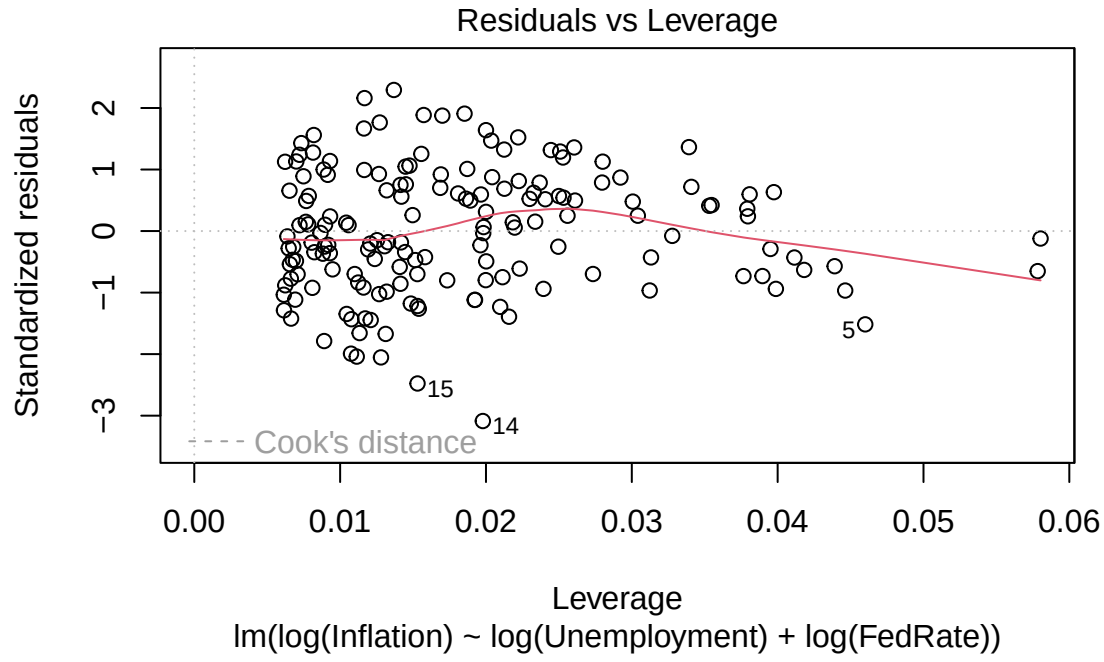
Plot the Model

```
plot(my_model)
```


Residuals vs Fitted
Residuals
0.5
−0.5
−1.5
49
15
14
0.0
0.5
1.0
1.5
2.0
Fitted values
lm(log(Inflation) ~ log(Unemployment) + log(FedRate))


Q−Q Residuals
Standardized residuals
2
1
0
−1
−2
−3
49
15
14
−2
−1
0
1
2
Theoretical Quantiles
lm(log(Inflation) ~ log(Unemployment) + log(FedRate))


Scale−Location
√|Standardized residuals|
14
15
49
0.0
0.5
1.0
1.5
0.0
0.5
1.0
1.5
2.0
Fitted values
lm(log(Inflation) ~ log(Unemployment) + log(FedRate))


Residuals vs Leverage
Standardized residuals
5
15
14
Cook's distance
−3
−1
0
1
2
0.00
0.01
0.02
0.03
0.04
0.05
0.06
Leverage
lm(log(Inflation) ~ log(Unemployment) + log(FedRate))

Visualize the Model Using Stargazer Library

```
library(stargazer)
stargazer(my_model, type = "text")
```

```
================================================
                        Dependent variable:
                    ----------------------------
                             log(Inflation)
------------------------------------------------
log(Unemployment)                 0.176
                                 (0.157)

log(FedRate)                     0.976***
                                 (0.085)

Constant                         -0.902***
                                 (0.291)

------------------------------------------------
Observations                       164
R2                                0.473
Adjusted R2                       0.466
Residual Std. Error         0.498 (df = 161)
F Statistic             72.167*** (df = 2; 161)
================================================
Note:                *p<0.1; **p<0.05; ***p<0.01
```

Obtain AIC and BIC for you Model

```
library(broom)
glance(my_model)
```

```
# A tibble: 1 x 12
  r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC
      <dbl>         <dbl> <dbl>     <dbl>    <dbl> <dbl>  <dbl> <dbl> <dbl>
1     0.473         0.466 0.498      72.2 4.22e-23     2  -117.  241.  254.
# i 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

Model Interpretation

The model above presents the results of a regression analysis with the dependent variable "log(Inflation)" and three independent variables: "log(Unemployment)", "log(FedRate)", and the constant term. The coefficients estimated for each independent variable represent the relationship between that variable and the dependent variable while holding other variables constant. The coefficients are accompanied by standard errors in parentheses.

log(Unemployment): The coefficient estimate for log(Unemployment) is 0.176. This means that a one-unit increase in the natural logarithm of unemployment is associated with a 0.176 unit increase in the natural logarithm of inflation. The standard error of 0.157 indicates the uncertainty in this estimate.

log(FedRate): The coefficient estimate for log(FedRate) is 0.976. This suggests that a one-unit increase in the natural logarithm of the Federal Reserve interest rate is associated with a 0.976 unit increase in the natural logarithm of inflation. The standard error of 0.085 provides an indication of the precision of this estimate.

Constant: The constant term in the model is -0.902. This represents the expected value of the natural logarithm of inflation when all independent variables are zero. The standard error of 0.291 reflects the uncertainty in this estimation.

The observations in the dataset used for the analysis are 164. The R-squared value of 0.473 indicates that approximately 47.3% of the variance in the natural logarithm of inflation can be explained by the independent variables included in the model. The adjusted R-squared value of 0.466 on the other hand accounts for the degrees of freedom in the model and provides a more conservative estimate of the proportion of variance explained.

The residual standard error of 0.498 indicates the average deviation of the observed values of the dependent variable from the predicted values, taking into account the degrees of freedom in the model.

The F-statistic of 72.167, with 2 and 161 degrees of freedom, suggests that the overall model is statistically significant. The associated p-value is less than 0.01, indicating strong evidence against the null hypothesis of no relationship between the independent variables and the dependent variable.

In summary, the model suggests that only log(Unemployment) has a statistically significant relationship with log(Inflation). However, it is important to note that these interpretations are based on the given coefficients, standard errors, and significance levels. Further analysis and consideration of the model's assumptions are necessary for a comprehensive understanding of the relationships between the variables.

Obtain the Root Mean Square Error

```
summary_model <- summary(my_model)
```

Extract RMSE from the summary

```
rmse <- sqrt(summary_model$sigma^2)
rmse
```

```
[1] 0.4975318
```

Print the summary and RMSE

```
print(summary_model)
```

```
Call:
lm(formula = log(Inflation) ~ log(Unemployment) + log(FedRate),
    data = mydatta)

Residuals:
     Min       1Q   Median       3Q      Max
-1.52073 -0.35028 -0.01588  0.35593  1.13292
```

```
Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)        -0.9016     0.2907  -3.102  0.00227 **
log(Unemployment)   0.1765     0.1567   1.127  0.26161
log(FedRate)        0.9764     0.0845  11.555  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4975 on 161 degrees of freedom
Multiple R-squared:  0.4727,    Adjusted R-squared:  0.4662
F-statistic: 72.17 on 2 and 161 DF,  p-value: < 2.2e-16
```

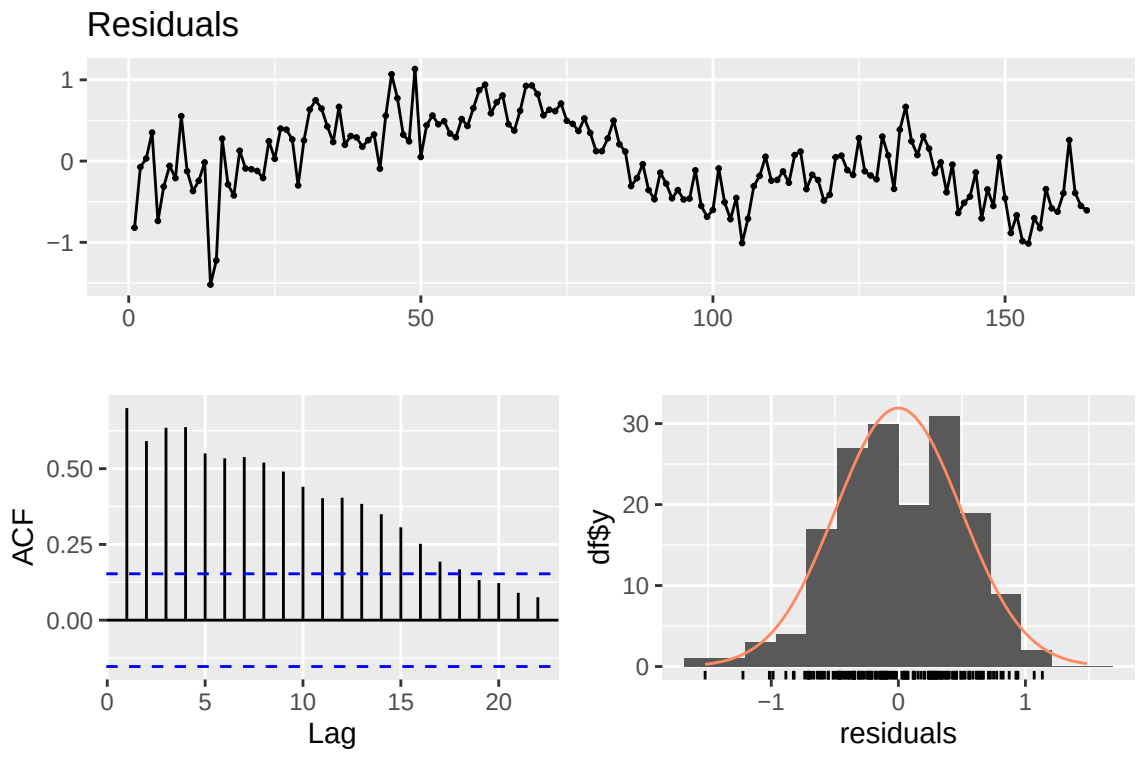
```
cat("Root Mean Square Error (RMSE):", rmse, "\n")
```

```
Root Mean Square Error (RMSE): 0.4975318
```

Testing the Assumptions

Normality of the error term

```
library(forecast)
checkresiduals(my_model)
```




```
Breusch-Godfrey test for serial correlation of order up to 10
```

```
data:  Residuals
LM test = 98.912, df = 10, p-value < 2.2e-16
```

Zero Conditional Mean

```
ReSid<-resid(my_model)
```

Add the residual variable to the data set

```
mydatta$ReSid <- ReSid
head(mydatta,5)
```

```
  year Unemployment Inflation  FedRate       ReSid
1 1859     5.133333 0.9084719 3.933333 -0.82027544
2 1860     5.233333 1.8107772 3.696667 -0.07333963
3 1861     5.533333 1.6227203 2.936667  0.03190288
4 1862     6.266667 1.7953352 2.296667  0.35104981
5 1863     6.800000 0.5370330 2.003333 -0.73682650
```

View Summary Statistics

```
library(stargazer)
library(gtsummary)
stargazer(mydatta[,-1], type = "text")
```

```
================================================
Statistic     N  Mean  St. Dev.  Min    Max
------------------------------------------------
Unemployment 164 5.960  1.510   3.400  10.667
Inflation    164 3.877  2.482   0.349  12.049
FedRate      164 6.590  3.184   1.683  17.780
ReSid        164 0.000  0.494   -1.521 1.133
------------------------------------------------
```

The variance covariance assumption

```
cov(ReSid, Unemployment)
```

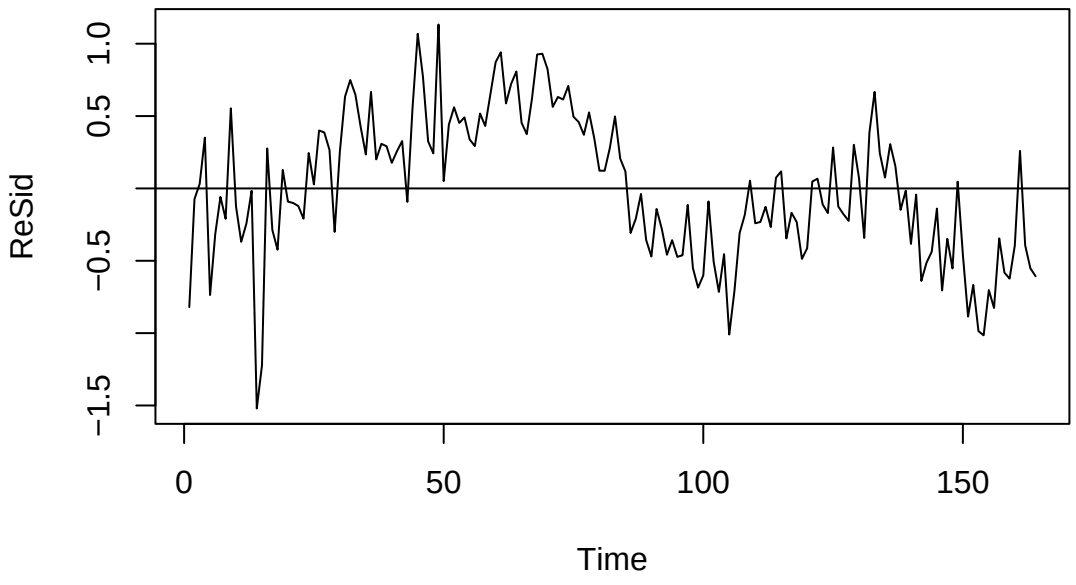
```
[1] 0.002701838
```

```
cov(ReSid, FedRate)
```

```
[1] -0.02217665
```

Plot the Residuals

```
ts.plot(ReSid)
abline(0,0.0000)
```



Multicollinearity

```
library(car)
library(tseries)
vif(my_model)
```

```
log(Unemployment)      log(FedRate)
         1.034369          1.034369
```

Heteroscedasticity

```
ncvTest(my_model)
```

```
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 2.623628, Df = 1, p = 0.10528
```

Autocorrelation

```
durbinWatsonTest(my_model)
```

```
 lag Autocorrelation D-W Statistic p-value
   1       0.7001071     0.5736665       0
 Alternative hypothesis: rho != 0
```

Suppose the variance of the Error terms was not homoscedastic

Estimating the Regression Model with Robust Standard errors

Robust standard errors, also known as heteroscedasticity-robust standard errors or White's standard errors, are a method to estimate the standard errors in regression analysis that account for potential heteroscedasticity (unequal variances) in the error terms.

In ordinary least squares (OLS) regression, the standard errors assume that the error terms have constant variance. However, in real-world data, it is common to encounter situations where the variability of the error terms changes across different levels of the independent variables. This violates the assumption of homoscedasticity, leading to incorrect standard error estimates, t-statistics, and p-values.

Robust standard errors address this issue by providing more accurate estimates of the standard errors that are robust to heteroscedasticity. They are calculated by estimating the variance-covariance matrix of the coefficient estimates using methods that do not assume constant variance of the errors.

There are different types of robust standard errors, including the HC1, HC2, and HC3 estimators, which differ in the specific assumptions they make about the structure of heteroscedasticity. These estimators are implemented in the sandwich package in R.

By using robust standard errors, researchers can obtain more reliable inference in regression analysis, particularly when there is evidence or suspicion of heteroscedasticity. Robust standard errors allow for valid hypothesis tests, confidence intervals, and t-statistics, even in the presence of heteroscedasticity, providing more accurate and robust statistical inference.

```
library(sandwich)
library(lmtest)
robust_se <- sqrt(diag(vcovHC(my_model, type = "HC1")))
robust_se
```

```
      (Intercept) log(Unemployment)      log(FedRate)
       0.27530083        0.14986315        0.07540774
```

View the Model

```
coeftest(my_model, vcov = vcovHC(my_model, type = "HC1"))
```

```
t test of coefficients:

                  Estimate Std. Error t value  Pr(>|t|)
(Intercept)       -0.901611   0.275301 -3.2750  0.001294 **
```

```
log(Unemployment)  0.176483   0.149863  1.1776  0.240683
log(FedRate)       0.976433   0.075408 12.9487 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

OMMITTED VARIABLE BIASE

```
fit <- lm(Inflation~Unemployment+FedRate, data = mydatta)
```

```
library(rempsyc)
library(jtools)
stargazer(fit,type = "text")
```

Estimate the first model

```
================================================
                        Dependent variable:
                    ---------------------------
                               Inflation
------------------------------------------------
Unemployment                    0.043
                               (0.100)

FedRate                        0.518***
                               (0.048)

Constant                        0.209
                               (0.607)

------------------------------------------------
Observations                     164
R2                              0.452
Adjusted R2                     0.445
Residual Std. Error       1.850 (df = 161)
F Statistic           66.318*** (df = 2; 161)
================================================
Note:                *p<0.1; **p<0.05; ***p<0.01
```

Run the second model

```
fit2 <- lm(Inflation~FedRate, data = mydatta)
```

```
stargazer(fit2,confint = TRUE, digits = 3, type = "text")
```

```
================================================
                        Dependent variable:
                    ----------------------------
                               Inflation
------------------------------------------------
FedRate                       0.524***
                               (0.045)

Constant                       0.426
                              (0.332)

------------------------------------------------
Observations                    164
R2                             0.451
Adjusted R2                    0.448
Residual Std. Error      1.845 (df = 162)
F Statistic          133.127*** (df = 1; 162)
================================================
Note:                *p<0.1; **p<0.05; ***p<0.01

====
TRUE
----
```

Test the omitted variable bias

```
library(lmtest)
waldtest(fit,fit2,test = "F")
```

```
Wald test

Model 1: Inflation ~ Unemployment + FedRate
Model 2: Inflation ~ FedRate
  Res.Df Df      F Pr(>F)
1    161
2    162 -1 0.1818 0.6704
```

The null hypothesis for this test states that the coefficient of the omitted variable is zero. Here the implication is that if we accept the null hypothesis, the variable was correctly omitted. On the other hand, the alternative hypothesis states that the coefficient of the omitted variable is not equal to zero. Therefore, rejecting the null hypothesis indicates that the variable was incorrectly omitted. From the results above, the p-value is approximately 0.6704, which indicates that we fail to reject the null hypothesis and conclude that the variable was correctly omitted. Thus, the omitted variable does not helps to explain the variation in the response variable.

Estimate the third model

```
fit3 <- lm(Inflation~Unemployment, data = mydatta)
```

```
stargazer(fit3,confint = TRUE, digits = 3, type = "text")
```

```
================================================
                        Dependent variable:
                    ----------------------------
                               Inflation
------------------------------------------------
Unemployment                  0.357***
                               (0.126)

Constant                       1.748**
                               (0.775)

------------------------------------------------
Observations                     164
R2                              0.047
Adjusted R2                     0.041
Residual Std. Error       2.431 (df = 162)
F Statistic           8.021*** (df = 1; 162)
================================================
Note:               *p<0.1; **p<0.05; ***p<0.01

====
TRUE
----
```

Test for the omitted variable bias

```
waldtest(fit,fit3,test = "F")
```

```
Wald test

Model 1: Inflation ~ Unemployment + FedRate
Model 2: Inflation ~ Unemployment
  Res.Df Df      F    Pr(>F)
1    161
2    162 -1 118.78 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The null hypothesis for this test states that the coefficient of the omitted variable is zero. Here the implication is that if we accept the null hypothesis, the variable was correctly omitted. On the other hand, the alternative hypothesis states that the coefficient of the omitted variable is not equal to zero. Therefore, rejecting the null hypothesis indicates that the variable was incorrectly omitted. From the results above, the p-value is approximately 0.001, which indicates that we reject the null hypothesis and conclude that the variable was incorrectly omitted. Thus, the omitted variable helps to explain the variation in the response variable.