Project 1

Venkata Satya Nagendra Sai Krishna Mohan Kocherlakota

College of Professional Studies, Northeastern University

ALY6000: Introduction to Analytics

Prof. Roy Wada

September 26, 2023

Introduction

This is an introductory project session that helps us understand the foundations of R, including arithmetic and logic operators like '+', '*', TRUE, FALSE, etc. I create vectors, datasets, and data frames to utilize operators and various functions such as 'hist' and 'ggplot' using the 'pacman' and 'tidyverse' libraries.

I have also used statistical functions for our knowledge for data analysis. Used min, max, mean, median & sd (minimum value, maximum value, mean, median and standard deviation of the vectors).

The built-in functions and operators used throughout the project are ‘c’, ‘seq’, ‘rep’, ‘range operator’, ‘sum’, ‘greater than/ less than’, ‘cumsum’, etc.

Used set.seed() function, runif and rnorm function for the first time, learnt there usage in R.

Key Findings: -

1. In the below mentioned snapshot, I have used the arithmetic operators like multiplication, squares, factorial and division of numbers. In addition to that I have used logical operator ‘AND’ and ‘OR’.

```
Console   Background Jobs
R 4.3.1 · ~/Rprojects/
> 123 * 453 #Multiplication of two numbers
[1] 55719
> 5^2 * 40
[1] 1000
> TRUE & FALSE #AND
[1] FALSE
> TRUE | FALSE #OR
[1] TRUE
> 75 %% 10 #Remainder
[1] 5
> 75 / 10 #Division
[1] 7.5
>
```

2. The below code is to increase, multiplication, greater than or equal to of a vector and showing the results of the vector based on the operators used respectively

```
second_vector + 20 #Increasing second vector value by 20 using addition operator
second_vector * 20 #Multiplication of second vector with 20 using multiplication
second_vector >= 20 #Checking if any values in second vector are greater than 20
second_vector !=20 #Checking any values in second vector is not equal to 20
#15
total <- sum(one_to_one_hundred) # Sum of vector
total
#16
average_value <- mean(one_to_one_hundred) # Mean of vector
average_value
#17
52:1   (Top Level)
Console   Background Jobs
R 4.3.1 · ~/Rprojects/
> second_vector + 20 #Increasing second vector value by 20 using addition operator
[1] 30 32 34 36 38 40 42 44 46 48 50
> second_vector * 20 #Multiplication of second vector with 20 using multiplication
[1] 200 240 280 320 360 400 440 480 520 560 600
> second_vector >= 20 #Checking if any values in second vector are greater than 20
[1] FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
> second_vector !=20 #Checking any values in second vector is not equal to 20
[1]  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE
>
```

3. The below vector gives us the TRUE values on the bases of the inform I provided for the first vector using logical operator.

```
#23
vector_from_boolean_brackets <- first_vector[c(FALSE, TRUE, FALSE, TRUE)] #Creating a vector using Boolean brackets
vector_from_boolean_brackets
#24
second_vector >= 20 #Checking all the values greater than or equal to 20
#25
ages_vector <- seq(from=10, to=30, by=2) #Creating a vector using Seq(from, to, by) function
ages_vector
#26
ages_vector [ages_vector >= 20] #Numbers greater than or equal to 20 in ages_vector
#27
lowest_grades_removed <- grades [grades >=85] #In grades vector removed all the grades less than or equal to 85
lowest_grades_removed
#28
```

81:1 (Top Level) R Script

Console Background Jobs

R 4.3.1 · ~/Rprojects/

```
> vector_from_boolean_brackets
[1] 12  5
>
```

4. The below operator gives us all the values greater than or equal to 20 of the second vector and ages_vector which I have created using 'seq' function.

```
> #24
> second_vector >= 20 #Checking all the values greater than or equal to 20
 [1] FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
> #26
> ages_vector [ages_vector >= 20] #Numbers greater than or equal to 20 in ages_vector
[1] 20 22 24 26 28 30
>
```

5. set.seed() : - This is a function when random variable is created, this function is helped to reproduce this random variable which I have created to store it.
   runif: - This function is used to create a random variable by giving the number of random variables with minimum value and maximum value of the vector.
   rnorm: - Function is used to create random variables using normal distribution and the number of variables, mean and standard deviation is given as input in order to generate the random variable (random vector)
   Used these functions to create two random variables/vectors as shown below.

```
#30
set.seed(5) # Used to create the exact same Random Variables every time.
random_vector <- runif(n=10, min=0, max=1000)#Used to create the vector of given length, which each value being random.
```

```
#37
set.seed(5)
random_vector <- rnorm(n=1000, mean=50, sd=15)#Used to create a normal distribution with 1000 values with mean as 50 and sd as 15
```

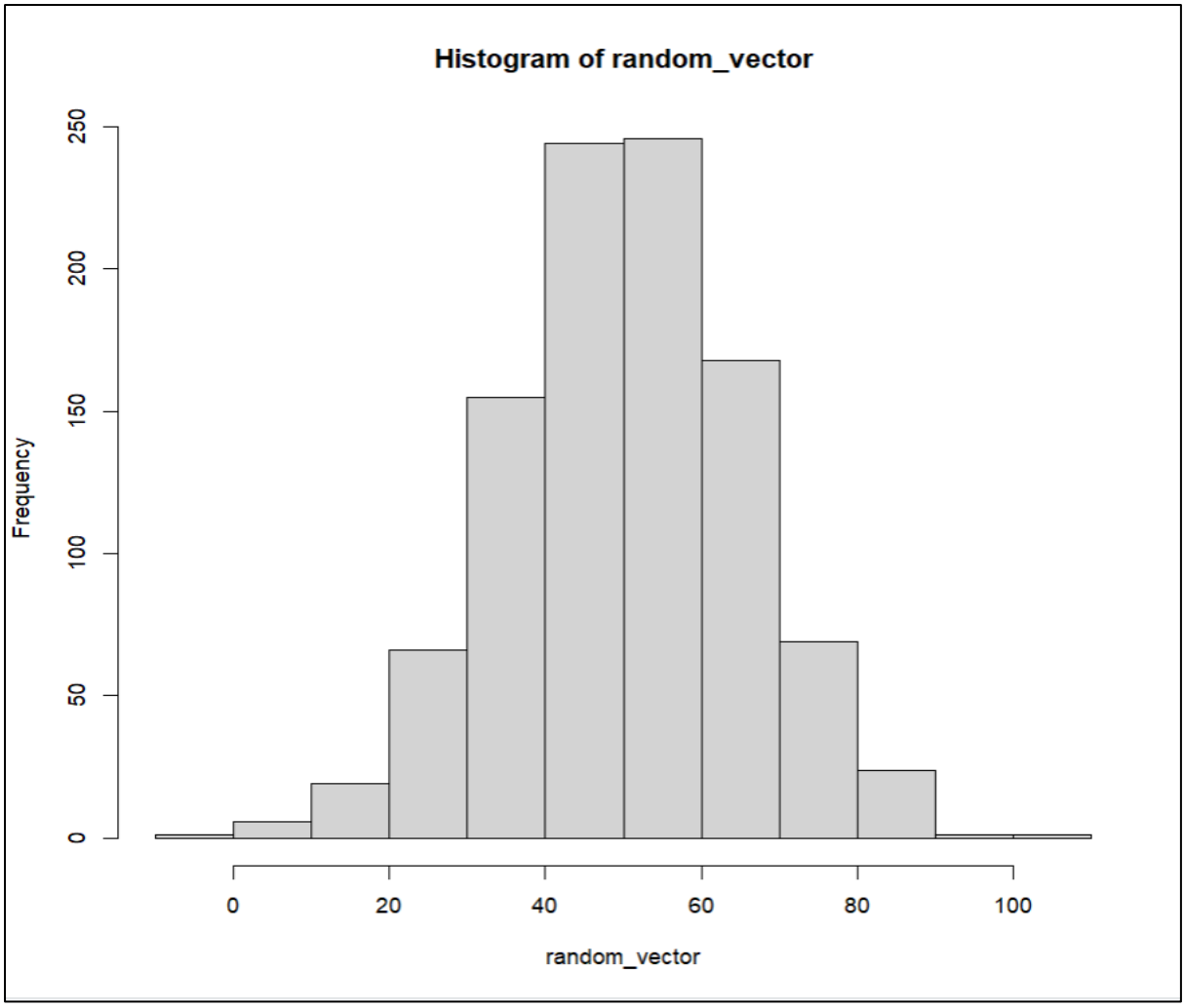
6. Histogram is used in the below screenshot using a random vector which was created using 'rnorm' function and 'hist' function in R.

```
#38
hist(random_vector) #Histogram of random vector
```

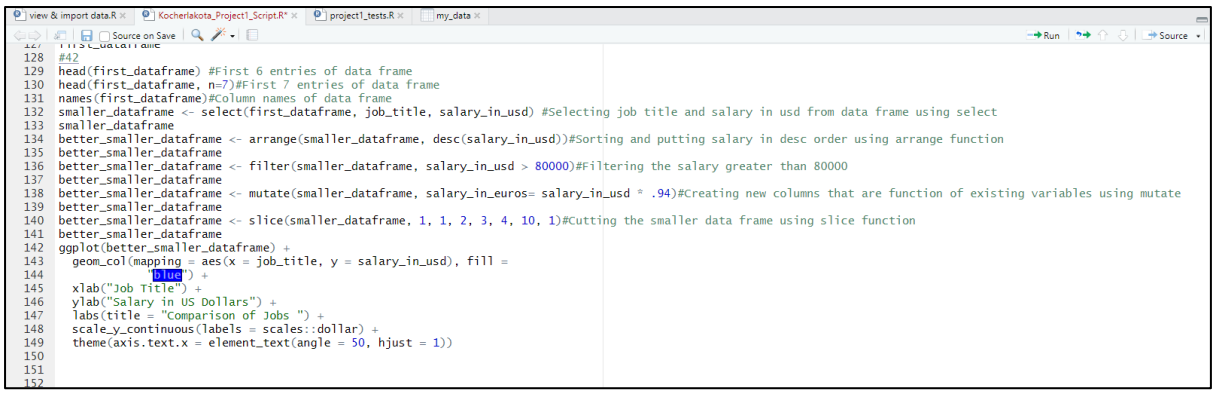
Console Background Jobs

R 4.3.1 · ~/Rprojects/

```
> #38
> hist(random_vector) #Histogram of random vector
>
```



7. The below screenshots have shown various operators and functions. I have created a data frame using 'read_csv' function and tidyverse library.
   head(data_frame): -The below function shows the first six rows of the data frame.
   head(data_frame, n=): - The below function shows the first seven rows of the data frame because I have number of rows to show using 'n' variable.
   names(data_frame): - The column names of the data frame.
   select,arrange, filter, mutate, slice: - These are various functions which are used for sorting, selection, filtering, slicing respectively.
   ggpolt: - This uses to plot the graph of given data frame and I have provided the titles, labels, etc.



```
#42
head(first_dataframe) #First 6 entries of data frame
head(first_dataframe, n=7)#First 7 entries of data frame
names(first_dataframe)#Column names of data frame
smaller_dataframe <- select(first_dataframe, job_title, salary_in_usd) #Selecting job title and salary in usd from data frame using select
smaller_dataframe
better_smaller_dataframe <- arrange(smaller_dataframe, desc(salary_in_usd))#Sorting and putting salary in desc order using arrange function
better_smaller_dataframe
better_smaller_dataframe <- filter(smaller_dataframe, salary_in_usd > 80000)#Filtering the salary greater than 80000
better_smaller_dataframe
better_smaller_dataframe <- mutate(smaller_dataframe, salary_in_euros= salary_in_usd * .94)#Creating new columns that are function of existing variables using mutate
better_smaller_dataframe
better_smaller_dataframe <- slice(smaller_dataframe, 1, 1, 2, 3, 4, 10, 1)#Cutting the smaller data frame using slice function
better_smaller_dataframe
ggplot(better_smaller_dataframe) +
  geom_col(mapping = aes(x = job_title, y = salary_in_usd), fill =
             "blue") +
  xlab("Job Title") +
  ylab("Salary in US Dollars") +
  labs(title = "Comparison of Jobs ") +
  scale_y_continuous(labels = scales::dollar) +
  theme(axis.text.x = element_text(angle = 50, hjust = 1))
```

```
> head(first_dataframe) #First 6 entries of data frame
  X work_year experience_level employment_type                 job_title salary salary_currency salary_in_usd employee_residence remote_ratio company_location
1 0      2020              MI              FT             Data Scientist  70000             EUR         79833                DE            0               DE
2 1      2020              SE              FT Machine Learning Scientist 260000             USD        260000                JP            0               JP
3 2      2020              SE              FT          Big Data Engineer  85000             GBP        109024                GB           50               GB
4 3      2020              MI              FT       Product Data Analyst  20000             USD         20000                HN            0               HN
5 4      2020              SE              FT  Machine Learning Engineer 150000             USD        150000                US           50               US
6 5      2020              EN              FT               Data Analyst  72000             USD         72000                US          100               US
  company_size
1            L
2            S
3            M
4            S
5            L
6            L
```

```
> head(first_dataframe, n=7)#First 7 entries of data frame
  X work_year experience_level employment_type                 job_title salary salary_currency salary_in_usd employee_residence remote_ratio company_location
1 0      2020              MI              FT             Data Scientist  70000             EUR         79833                DE            0               DE
2 1      2020              SE              FT Machine Learning Scientist 260000             USD        260000                JP            0               JP
3 2      2020              SE              FT          Big Data Engineer  85000             GBP        109024                GB           50               GB
4 3      2020              MI              FT       Product Data Analyst  20000             USD         20000                HN            0               HN
5 4      2020              SE              FT  Machine Learning Engineer 150000             USD        150000                US           50               US
6 5      2020              EN              FT               Data Analyst  72000             USD         72000                US          100               US
7 6      2020              SE              FT        Lead Data Scientist 190000             USD        190000                US          100               US
  company_size
1            L
2            S
3            M
4            S
5            L
6            L
7            S
```

```
> names(first_dataframe)#Column names of data frame
 [1] "X"                 "work_year"         "experience_level"  "employment_type"   "job_title"         "salary"            "salary_currency"   "salary_in_usd"
 [9] "employee_residence" "remote_ratio"      "company_location"  "company_size"
>
```

Console Background Jobs

R 4.3.1 · ~/Rprojects/

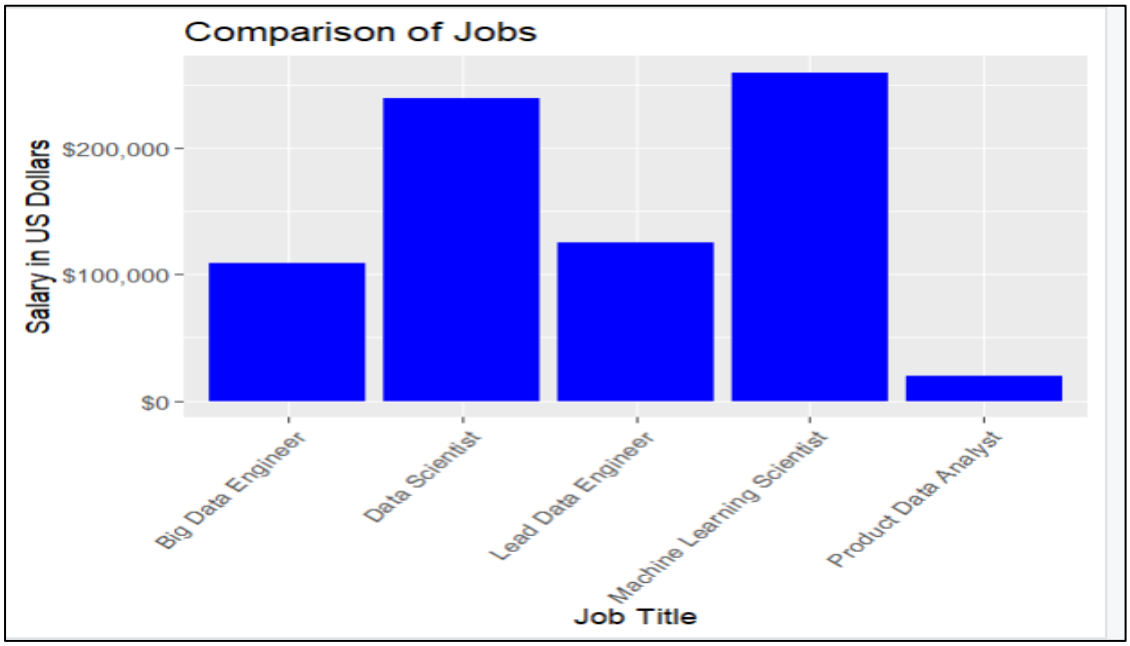
```
> smaller_dataframe <- select(first_dataframe, job_title, salary_
> smaller_dataframe
                                  job_title salary_in_usd
1                            Data Scientist         79833
2                  Machine Learning Scientist        260000
3                       Big Data Engineer        109024
4                     Product Data Analyst         20000
5                 Machine Learning Engineer        150000
6                            Data Analyst         72000
7                      Lead Data Scientist        190000
8                            Data Scientist         35735
9                    Business Data Analyst        135000
10                      Lead Data Engineer        125000
11                           Data Scientist         51321
12                           Data Scientist         40481
13                           Data Scientist         39916
14                       Lead Data Analyst         87000
15                           Data Analyst         85000
16                           Data Analyst          8000
```

```
Console   Background Jobs
R 4.3.1 · ~/Rprojects/
> better_smaller_dataframe <- arrange(smaller_dataframe, desc(s
> better_smaller_dataframe
                                  job_title salary_in_usd
1                    Principal Data Engineer        600000
2                         Research Scientist        450000
3                      Financial Data Analyst        450000
4          Applied Machine Learning Scientist        423000
5                    Principal Data Scientist        416000
6                              Data Scientist        412000
7                         Data Analytics Lead        405000
8                      Applied Data Scientist        380000
9                    Director of Data Science        325000
10                              Data Engineer        324000
11                         Lead Data Engineer        276000
12                                 ML Engineer        270000
13                              Data Architect        266400
14                 Machine Learning Scientist        260000
15                             Data Scientist        260000
16                                 ML Engineer        256000
```

```
Console   Background Jobs
R 4.3.1 · ~/Rprojects/
> better_smaller_dataframe <- filter(smaller_dataframe, salary_in_usd > 80000)#Filtering the salary greater than 8000
> better_smaller_dataframe
                              job_title salary_in_usd
1                 Machine Learning Scientist        260000
2                          Big Data Engineer        109024
3                  Machine Learning Engineer        150000
4                        Lead Data Scientist        190000
5                       Business Data Analyst        135000
6                          Lead Data Engineer        125000
7                           Lead Data Analyst         87000
8                               Data Analyst         85000
9                          Big Data Engineer        114047
10                           BI Data Analyst         98000
11                       Lead Data Scientist        115000
12                    Director of Data Science        325000
13                       Business Data Analyst        100000
14                    Machine Learning Manager        117104
15                          Research Scientist        450000
16                      Data Science Consultant        103000
```

```
Console   Background Jobs
R 4.3.1 · ~/Rprojects/
> better_smaller_dataframe <- mutate(smaller_dataframe, salary_in_euros= salary_in_usd * .94
> better_smaller_dataframe
                                  job_title salary_in_usd salary_in_euros
1                            Data Scientist         79833        75043.02
2                  Machine Learning Scientist        260000       244400.00
3                           Big Data Engineer        109024       102482.56
4                         Product Data Analyst         20000        18800.00
5                    Machine Learning Engineer        150000       141000.00
6                                Data Analyst         72000        67680.00
7                          Lead Data Scientist        190000       178600.00
8                            Data Scientist         35735        33590.90
9                         Business Data Analyst        135000       126900.00
10                           Lead Data Engineer        125000       117500.00
11                           Data Scientist         51321        48241.74
12                           Data Scientist         40481        38052.14
13                           Data Scientist         39916        37521.04
14                          Lead Data Analyst         87000        81780.00
15                               Data Analyst         85000        79900.00
16                               Data Analyst          8000         7520.00
```

```
Console   Background Jobs
R 4.3.1 · ~/Rprojects/
> better_smaller_dataframe <- slice(smaller_dataframe, 1, 1, 2, 3, 4, 10, 1)#Cutting the smaller data frame using slice function
> better_smaller_dataframe
                  job_title salary_in_usd
1             Data Scientist         79833
2             Data Scientist         79833
3 Machine Learning Scientist        260000
4           Big Data Engineer        109024
5       Product Data Analyst         20000
6         Lead Data Engineer        125000
7             Data Scientist         79833
>
```



Conclusion: -

In this module, I have learned the R language and how to use RStudio. I have also gained a basic understanding of arithmetic, logical, and statistical operators and functions. This has boosted my confidence in pursuit of my goal to become a data analyst. Learning about functions like 'runif,' 'rnorm,' and 'set.seed()' has been particularly valuable. Additionally, I have utilized various other operators such as 'range(),' 'mutate(),' 'slice(),' 'arrange(),' 'filter(),' etc. I have created numerous vectors, datasets, and data frames during this learning process.

I have successfully run all the test cases. Attaching screenshot for reference.

```
Rprojects - RStudio
File  Edit  Code  View  Plots  Session  Build  Debug  Profile  Tools  Help
Go to file/function   Addins
view & import data.R   Kocherlakota_Project1_Script.R*   project1_tests.R   my_data
Source on Save   Run   Source
132 smaller_dataframe <- select(first_dataframe, job_title, salary_in_usd) #Selecting job title and salary in usd from da
133 smaller_dataframe
134 better_smaller_dataframe <- arrange(smaller_dataframe, desc(salary_in_usd))#Sorting and putting salary in desc order
135 better_smaller_dataframe
136 better_smaller_dataframe <- filter(smaller_dataframe, salary_in_usd > 80000)#Filtering the salary greater than 80000
137 better_smaller_dataframe
138 better_smaller_dataframe <- mutate(smaller_dataframe, salary_in_euros= salary_in_usd * .94)#Creating new columns that
139 better_smaller_dataframe
140 better_smaller_dataframe <- slice(smaller_dataframe, 1, 1, 2, 3, 4, 10, 1)#Cutting the smaller data frame using slice
141 better_smaller_dataframe
142 ggplot(better_smaller_dataframe) +
143   geom_col(mapping = aes(x = job_title, y = salary_in_usd), fill =
144            "blue") +
145   xlab("Job Title") +
146   ylab("Salary in US Dollars") +
147   labs(title = "Comparison of Jobs ") +
148   scale_y_continuous(labels = scales::dollar) +
149   theme(axis.text.x = element_text(angle = 50, hjust = 1))
150
151
152
149:60   (Top Level)   R Script
Console   Background Jobs
R 4.3.1 · ~/Rprojects/
> library(pacman)
> p_load(testthat)
> test_file("project1_tests.R")
[ FAIL 0 | WARN 0 | SKIP 0 | PASS 72 ]
```

Citations: -

1. Zach. (2022). How (And When) to Use set.seed in R. Statology. https://www.statology.org/set-seed-in-r/#:~:text=The%20set.,time%20you%20run%20the%20code.Tu

2. *What Is the Algorithm Used by the rnorm Function in R? | Saturn Cloud Blog*. (2023, September 9). https://saturncloud.io/blog/what-is-the-algorithm-used-by-the-rnorm-function-in-r/#:~:text=In%20R%2C%20the%20rnorm%20function%20is%20used%20to%20generate%20random,standard%20deviation%20of%20the%20distribution.

3. H. T. (n.d.). Several ways to use runif. https://www.linkedin.com/pulse/several-ways-use-runif-henry-trunghieu-tu/

4. Subset rows using their positions — slice. (n.d.). https://dplyr.tidyverse.org/reference/slice.html

5. mutate function - RDocumentation. (n.d.). https://www.rdocumentation.org/packages/dplyr/versions/0.5.0/topics/mutate