

OMISOL

2023-10-10

#NUMBER 1 #A.

```
household <- data.frame (  
  Respondents = 1:20,  
  Sex = c(2, 2, 1, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2, 1, 2),  
  Fathers_Occupation = c(1, 3, 3, 3, 1, 2, 3, 1, 1, 1, 3, 2, 1, 3, 3, 1, 3, 1, 2, 1),  
  Person_at_Home = c(5, 7, 3, 8, 5, 9, 6, 7, 8, 4, 7, 5, 4, 7, 8, 8, 3, 11, 7, 6),  
  Siblings_at_school = c(6, 4, 4, 1, 2, 1, 5, 3, 1, 2, 3, 2, 5, 5, 2, 1, 2, 5, 3, 2),  
  Types_of_houses = c(1, 2, 3, 1, 1, 3, 3, 1, 2, 3, 2, 3, 2, 2, 3, 3, 3, 3, 3, 2)  
)  
household
```

##	Respondents	Sex	Fathers_Occupation	Person_at_Home	Siblings_at_school
## 1	1	2	1	5	6
## 2	2	2	3	7	4
## 3	3	1	3	3	4
## 4	4	2	3	8	1
## 5	5	2	1	5	2
## 6	6	2	2	9	1
## 7	7	2	3	6	5
## 8	8	2	1	7	3
## 9	9	2	1	8	1
## 10	10	2	1	4	2
## 11	11	1	3	7	3
## 12	12	2	2	5	2
## 13	13	2	1	4	5
## 14	14	2	3	7	5
## 15	15	2	3	8	2
## 16	16	2	1	8	1
## 17	17	2	3	3	2
## 18	18	2	1	11	5
## 19	19	1	2	7	3
## 20	20	2	1	6	2
##	Types_of_houses				
## 1	1				
## 2	2				
## 3	3				
## 4	1				
## 5	1				
## 6	3				
## 7	3				
## 8	1				
## 9	2				
## 10	3				
## 11	2				
## 12	3				

```
## 13          2
## 14          2
## 15          3
## 16          3
## 17          3
## 18          3
## 19          3
## 20          2
```

#B.

```
str(household)
```

```
## 'data.frame':  20 obs. of  6 variables:
## $ Respondents      : int  1 2 3 4 5 6 7 8 9 10 ...
## $ Sex              : num  2 2 1 2 2 2 2 2 2 2 ...
## $ Fathers_Occupation: num  1 3 3 3 1 2 3 1 1 1 ...
## $ Person_at_Home    : num  5 7 3 8 5 9 6 7 8 4 ...
## $ Siblings_at_school: num  6 4 4 1 2 1 5 3 1 2 ...
## $ Types_of_houses   : num  1 2 3 1 1 3 3 1 2 3 ...
```

```
summary(household)
```

```
##   Respondents      Sex      Fathers_Occupation Person_at_Home
##   Min.   : 1.00   Min.   :1.00   Min.   :1.00   Min.   : 3.0
##   1st Qu.: 5.75   1st Qu.:2.00   1st Qu.:1.00   1st Qu.: 5.0
##   Median :10.50   Median :2.00   Median :2.00   Median : 7.0
##   Mean   :10.50   Mean   :1.85   Mean   :1.95   Mean   : 6.4
##   3rd Qu.:15.25   3rd Qu.:2.00   3rd Qu.:3.00   3rd Qu.: 8.0
##   Max.   :20.00   Max.   :2.00   Max.   :3.00   Max.   :11.0
##   Siblings_at_school Types_of_houses
##   Min.   :1.00     Min.   :1.0
##   1st Qu.:2.00     1st Qu.:2.0
##   Median :2.50     Median :2.5
##   Mean   :2.95     Mean   :2.3
##   3rd Qu.:4.25     3rd Qu.:3.0
##   Max.   :6.00     Max.   :3.0
```

#C.

```
mean_siblings <- mean(household$Siblings_at_school)
mean_siblings == 5
```

```
## [1] FALSE
```

#D.

```
subset1 <- household[1:2, ]
subset1
```

```
##   Respondents Sex Fathers_Occupation Person_at_Home Siblings_at_school
## 1           1  2              1              5              6
## 2           2  2              3              7              4
##   Types_of_houses
## 1               1
## 2               2
```

#E.

```

subset2 <- household[c(3, 5), c(2, 4)]
subset2

##    Sex Person_at_Home
## 3    1              3
## 5    2              5

#F.
types_houses <- household$Types_of_houses
types_houses

## [1] 1 2 3 1 1 3 3 1 2 3 2 3 2 2 3 3 3 3 3 2

#G.
male_farmer <- subset(household, Sex == 1 & Fathers_Occupation == 1)
male_farmer

## [1] Respondents      Sex      Fathers_Occupation Person_at_Home
## [5] Siblings_at_school Types_of_houses
## <0 rows> (or 0-length row.names)

#H.
female_greater_than_5_siblings <- subset(household, Sex == 2 & Siblings_at_school >= 5)
female_greater_than_5_siblings

##    Respondents Sex Fathers_Occupation Person_at_Home Siblings_at_school
## 1             1  2                  1              5              6
## 7             7  2                  3              6              5
## 13            13  2                  1              4              5
## 14            14  2                  3              7              5
## 18            18  2                  1             11              5
##    Types_of_houses
## 1                 1
## 7                 3
## 13                2
## 14                2
## 18                3

#NUMBER 2
df <- data.frame(
  Ints = integer(),
  Doubles = double(),
  Characters = character(),
  Logicals = logical(),
  Factors = factor(),
  stringsAsFactors = FALSE
)
cat("Structure of the empty dataframe:\n")

## Structure of the empty dataframe:
str(df)

## 'data.frame':    0 obs. of  5 variables:
## $ Ints      : int
## $ Doubles   : num

```

```
## $ Characters: chr
## $ Logicals  : logi
## $ Factors   : Factor w/ 0 levels:
```

#Output The output shows that the data frame has 0 observations (rows) and 5 variables (columns) with their respective data types. The “Factors” column is empty since there are no levels defined yet.

#NUMBER 3

```
household_data <- data.frame (
  Respondents = 1:10,
  Sex = c("Male", "Female", "Female", "Male", "Male", "Female", "Female", "Male", "Female", "Male"),
  Fathers_Occupation = c(1,2,3,3,1,2,2,3,1,3),
  Person_at_Home = c(5,7,3,8,6,4,4,2,11,6),
  Siblings_at_school = c(2,3,0,5,2,3,1,2,6,2),
  Types_of_houses = c("Wood", "Congrete", "Congrete", "Wood", "Semi-Congrete", "Semi-Congrete", "Wood",
)
household_data
```

```
##      Respondents      Sex Fathers_Occupation Person_at_Home Siblings_at_school
## 1             1    Male                1             5             2
## 2             2  Female                2             7             3
## 3             3  Female                3             3             0
## 4             4    Male                3             8             5
## 5             5    Male                1             6             2
## 6             6  Female                2             4             3
## 7             7  Female                2             4             1
## 8             8    Male                3             2             2
## 9             9  Female                1            11             6
## 10           10    Male                3             6             2
##      Types_of_houses
## 1             Wood
## 2             Congrete
## 3             Congrete
## 4             Wood
## 5      Semi-Congrete
## 6      Semi-Congrete
## 7             Wood
## 8      Semi-Congrete
## 9      Semi-Congrete
## 10            Congrete
```

```
write.csv(household_data, file = "HouseholdData.csv", row.names = FALSE)
```

#A.

```
household_data <- read.csv("HouseholdData.csv")
```

#B.

```
household_data$Sex <- factor(household_data$Sex)
household_data$Sex <- as.integer(factor(household_data$Sex,
                                         levels = c("Male", "Female"),
                                         labels = c(1, 2)))
household_data
```

```
##      Respondents Sex Fathers_Occupation Person_at_Home Siblings_at_school
## 1             1   1                1             5             2
```

```
## 2      2  2      2      7      3
## 3      3  2      3      3      0
## 4      4  1      3      8      5
## 5      5  1      1      6      2
## 6      6  2      2      4      3
## 7      7  2      2      4      1
## 8      8  1      3      2      2
## 9      9  2      1     11      6
## 10     10  1      3      6      2
##      Types_of_houses
## 1      Wood
## 2      Congrete
## 3      Congrete
## 4      Wood
## 5      Semi-Congrete
## 6      Semi-Congrete
## 7      Wood
## 8      Semi-Congrete
## 9      Semi-Congrete
## 10     Congrete
```

#C.

```
household_data$Types_of_houses <- factor(household_data$Types_of_houses)
household_data$Types_of_houses <- as.integer(factor(household_data$Types_of_houses,
                                                    levels = c("Wood", "Congrete", "Semi-Congrete"),
                                                    labels = c(1, 2, 3)))
print(household_data)
```

```
##      Respondents Sex Fathers_Occupation Person_at_Home Siblings_at_school
## 1      1      1      1      5      2
## 2      2      2      2      7      3
## 3      3      2      3      3      0
## 4      4      1      3      8      5
## 5      5      1      1      6      2
## 6      6      2      2      4      3
## 7      7      2      2      4      1
## 8      8      1      3      2      2
## 9      9      2      1     11      6
## 10     10      1      3      6      2
##      Types_of_houses
## 1      1
## 2      2
## 3      2
## 4      1
## 5      3
## 6      3
## 7      1
## 8      3
## 9      3
## 10     2
```

#D.

```
household_data$Fathers_Occupation <- factor(household_data$Fathers_Occupation)
household_data$Fathers_Occupation <- as.character(factor(household_data$Fathers_Occupation,
```

```

levels = c(1, 2, 3),
labels = c("Farmer", "Driver", "Others"))

# Print the updated data frame
print(household_data)

```

```

##      Respondents Sex Fathers_Occupation Person_at_Home Siblings_at_school
## 1             1   1           Farmer             5           2
## 2             2   2           Driver             7           3
## 3             3   2           Others             3           0
## 4             4   1           Others             8           5
## 5             5   1           Farmer             6           2
## 6             6   2           Driver             4           3
## 7             7   2           Driver             4           1
## 8             8   1           Others             2           2
## 9             9   2           Farmer            11           6
## 10           10   1           Others             6           2

```

```

##      Types_of_houses
## 1                   1
## 2                   2
## 3                   2
## 4                   1
## 5                   3
## 6                   3
## 7                   1
## 8                   3
## 9                   3
## 10                  2

```

#E.

```

female_driver <- subset(household_data, Sex == 2 & Fathers_Occupation == "Driver")
female_driver

```

```

##      Respondents Sex Fathers_Occupation Person_at_Home Siblings_at_school
## 2             2   2           Driver             7           3
## 6             6   2           Driver             4           3
## 7             7   2           Driver             4           1

```

```

##      Types_of_houses
## 2                   2
## 6                   3
## 7                   1

```

#F.

```

greater_than_5_siblings <- subset(household_data, Siblings_at_school >= 5)
greater_than_5_siblings

```

```

##      Respondents Sex Fathers_Occupation Person_at_Home Siblings_at_school
## 4             4   1           Others             8           5
## 9             9   2           Farmer            11           6

```

```

##      Types_of_houses
## 4                   1
## 9                   3

```

#4. INTERPRET THE GRAPH The data you've described pertains to sentiment analysis conducted on

Twitter data during specific dates in July 2020, namely July 14, 15, 17, 18, 20, and 21. This sentiment analysis classifies tweets into three categories: negative, neutral, and positive. Here's a rephrased interpretation:

The information presented represents an analysis of sentiments expressed on Twitter during the specified dates in July 2020. It segments tweets into three sentiment categories: negativity, neutrality, and positivity. A summary of the findings reveals:

- July 14, 2020:
 - Negative tweets: Close to 2,500
 - Neutral tweets: Approximately 1,500
 - Positive tweets: About 1,750
- July 15, 2020:
 - Negative tweets: Exceeding 4,000
 - Neutral tweets: Approximately 2,750
 - Positive tweets: Roughly 3,200
- July 17, 2020:
 - Negative tweets: Around 3,250
 - Neutral tweets: Approximately 1,800
 - Positive tweets: Nearly 2,500
- July 18, 2020:
 - Negative tweets: About 3,250
 - Neutral tweets: Approximately 2,000
 - Positive tweets: Roughly 2,500
- July 20, 2020:
 - Negative tweets: Nearly 2,500
 - Neutral tweets: About 1,500
 - Positive tweets: Almost 1,750
- July 21, 2020:
 - Negative tweets: Approximately 4,000
 - Neutral tweets: About 2,600
 - Positive tweets: Roughly 3,300

In summary, the data suggests that sentiment on Twitter exhibited fluctuations during these specific dates in July 2020. Notably, July 15th and July 21st saw higher volumes of both negative and positive tweets, indicating a potential surge in sentiment-related Twitter activity. Conversely, July 14th and July 20th had lower counts across all sentiment categories. This data illustrates the dynamic nature of sentiment expression on Twitter during this particular period in 2020.