HW-Group-19-Assignment-2

Abhinav,Mukul,Anurag

26/09/2021

IE-5379-Assignment-2 Abhinav, Mukul, Anurag (Homework group 19)

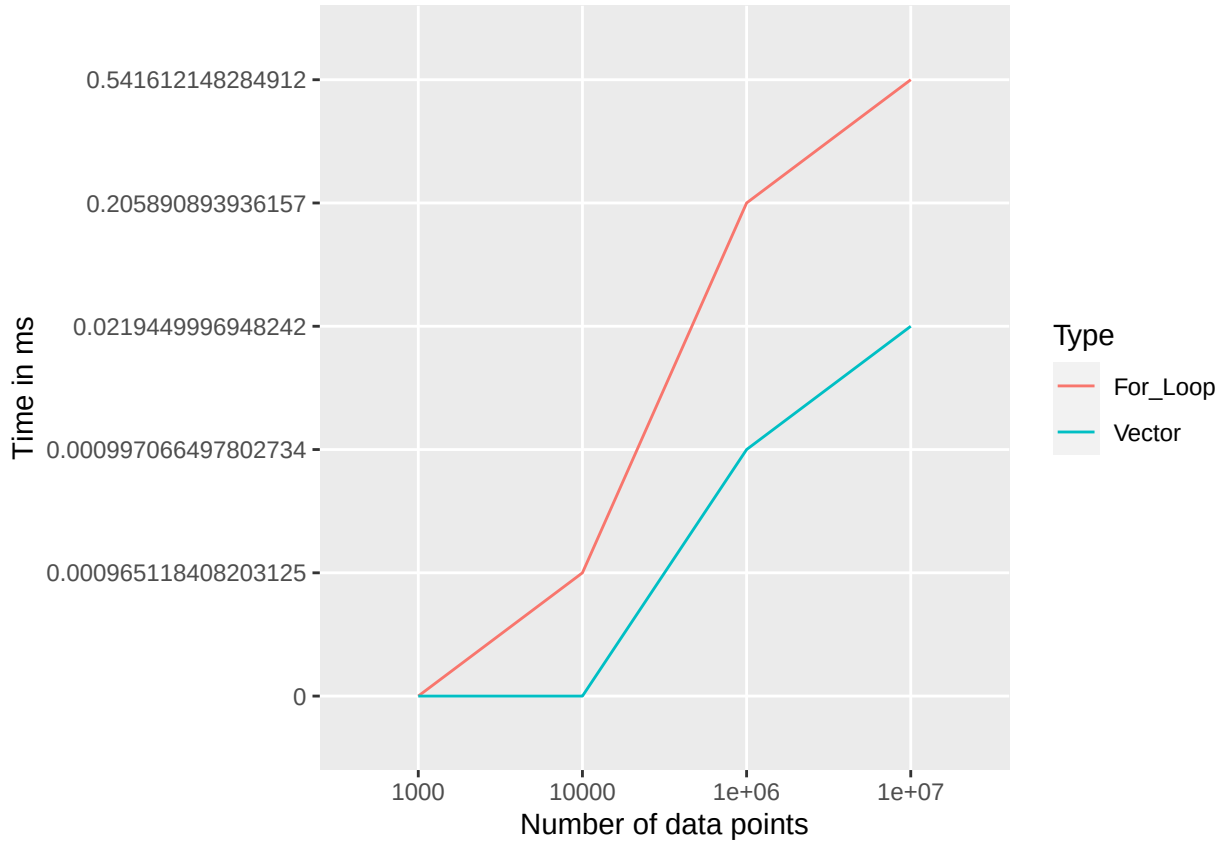
Problem 1 - Write your observations on the effect of vector length on computing time using "for loop" and vector operation

```
n_vec <- c(1000, 10000, 1000000, 10000000)
df <- data.frame(matrix(ncol = 3, nrow = 0))
colnames(df) <- c("Number", "For_Loop", "Vector")

for (n in n_vec) {
  x <- 1:n

  ## Use of for loop
  t1 <- Sys.time()
  for (i in 1:length(x)) {
    x[i] <- x[i] * 2
  }
  t2 <- Sys.time()

  ## Vector
  t3 <- Sys.time()
  x <- x * 2
  t4 <- Sys.time()

  ## calculate difference and store in variables
  diff1 <- t2 - t1
  diff2 <- t4 - t3

  ## store both time difference variables in data frame prepared above
  df[nrow(df) + 1, ] <- c(as.character(n), diff1, diff2)
}

## Pivot the data to get the both the types in a single column
df_longer <- df %>%
  pivot_longer(c("For_Loop", "Vector"), names_to = "Type", values_to = "Time")
print(df_longer)
```

```
## # A tibble: 8 x 3
```

```
##   Number Type     Time
##   <chr>  <chr>    <chr>
## 1 1000   For_Loop 0
## 2 1000   Vector   0
## 3 10000  For_Loop 0.000965118408203125
## 4 10000  Vector   0
## 5 1e+06  For_Loop 0.205890893936157
## 6 1e+06  Vector   0.000997066497802734
## 7 1e+07  For_Loop 0.541612148284912
## 8 1e+07  Vector   0.0219449996948242
```

```
## Plot the line plot
ggplot(data = df_longer, aes(x = Number, y = Time, group = Type, color = Type)) +
  geom_line() +
  xlab("Number of data points") +
  ylab("Time in ms")
```



> As we can observe from the above line plot that Vector method always consumes less time than For-Loop method. As the number of data points increase, using vectors sharply out performs using for loops.

Problem 2 -> Write a custom function that accepts a dataframe from the user and returns a dataframesample with a random number of records. The fraction of records must be given by the user inthe form of an argument. If the user enters a non-numeric argument, notify using an errormessage that they have to enter a numeric value.

```
get_sample_data <- function(df, n = 1000) {
  # Check if n is supplied a numeric value
  if (!is.numeric(n)) {
    stop("Value of n must be numeric")
  }

  # Get number of rows dataframe
  rows_in_df <- dim(df)[1]

  # If we have lesser values in dataframe than sample, throw error
  if (rows_in_df < n) {
    stop("Must have >= n values in data frame")
  }

  if (n <= 0) {
    stop("N must be positive and > 0")
  }

  # Sample rows to return values
  rows_idx <- floor(runif(n, 1, n))

  # Return sampled dataframe
  return(df[rows_idx, ])
}

num <- readline("Please enter a number:")
```

```
## Please enter a number:
```

```
num <- as.integer(num)

if ((is.na(num)) || (!is.numeric(num))) {
  print("Please enter a valid  number.")
} else {
  answer_2 <- get_sample_data(df_covid_nursing, num)
  head(answer_2, 5)
}
```

```
## [1] "Please enter a valid  number."
```

3 Covid nursing dataset questions

Before doing any analysis, first check the quality of the data. We have column in the dataset having title "Passed Quality Assurance Check". Drop all the rows having a value as either 'N' or NULL OR select rows with the value as 'Y'.

```
print(paste0("The rows in the dataframe before filter ", dim(df_covid_nursing)[1]))
```

```
## [1] "The rows in the dataframe before filter 1057181"
```

```
df_covid_nursing <- df_covid_nursing[(df_covid_nursing$passed_quality_assurance_check == "Y"), ]
print(paste0("The shape of dataframe after filter ", dim(df_covid_nursing)[1]))
```

```
## [1] "The shape of dataframe after filter 1045758"
```

Question 3.1 - Count the number of nursing home facilities by state and order them in descending order.

```
answer1 <- df_covid_nursing %>%
  group_by(provider_state) %>%
  summarise(count = n_distinct(federal_provider_number))
answer1[order(answer1$count, decreasing = T), ]
```

```
## # A tibble: 54 x 2
##    provider_state count
##    <chr>          <int>
##  1 TX              1213
##  2 CA              1178
##  3 OH               959
##  4 IL               715
##  5 FL               702
##  6 PA               686
##  7 NY               616
##  8 IN               526
##  9 MO               520
## 10 IA               434
## # ... with 44 more rows
```

Question 3.2 - Display the top five counties by number of nursing home facilities.

```
answer2 <- df_covid_nursing %>%
  group_by(provider_state, county) %>%
  summarise(count = n_distinct(federal_provider_number))
```

```
## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.
```

```
head(answer2[order(answer2$count, decreasing = T), ], 5)
```

```
## # A tibble: 5 x 3
## # Groups:   provider_state [4]
##   provider_state county      count
##   <chr>          <chr>       <int>
```

```
## 1 CA             Los Angeles   377
## 2 IL             Cook          209
## 3 TX             Harris        104
## 4 OH             Cuyahoga       98
## 5 TX             Dallas         86
```

3.3 Display four tables containing the states which experienced more than 20 weeks of shortage of nursing staff (table 1), clinical staff (table 2), aides (table 3), and other staff (table 4) (20points)

We have approached the analysis using two major approaches:

Approach 1

1. Find all federal providers within a state.
2. Find federal providers with missing staff where the shortage was prevalent for >= 20 weeks.
3. Join tables with total providers per state and state data with missing providers, to calculate missing percentage. 4.. Inspect the data and set a threshhold, for eg, for Nursing staff missing in 5 / 10 facilities in a state we count the same as (50%), and with a threshhold of 25%, we take all values >= 25.
4. Summarize and display all such states with missing staffs and mention our observations.

```
# Used to identify the columns relavant to the data
column_names <- names(df_covid_nursing)


# These are the column names relevant to shoage
columns_to_convert <- column_names[c(66:69)]

# Group and count number of providers by state
state_with_count <- df_covid_nursing %>%
  group_by(provider_state) %>%
  summarize(count_of_providers = n_distinct(federal_provider_number))

ind <- 1
for (col_name in columns_to_convert) {

  # Filter all values where current column is missing for more than 20 weeks, counting number of federa
  temp <- df_covid_nursing %>%
    filter(df_covid_nursing[col_name] == "Y") %>%
    group_by(provider_state, federal_provider_number) %>%
    summarize(count_of_shortage = n()) %>%
    filter(count_of_shortage >= 20) %>%
    group_by(provider_state) %>%
    summarize(count_of_missing = n())

  # Join this with state facility count table
  joined_temp <- inner_join(state_with_count, temp, by = "provider_state")

  # Calculate shortage percentage
  joined_temp$shortage_percent <- (joined_temp$count_of_missing / joined_temp$count_of_providers) * 100

  # Dynamically assign in new variable
  assign(paste("table", col_name, sep = "_"), joined_temp)
```

```
  # Increment index by one for next variable
  ind <- ind + 1
}
```

```
## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.
## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.
## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.
## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.
```

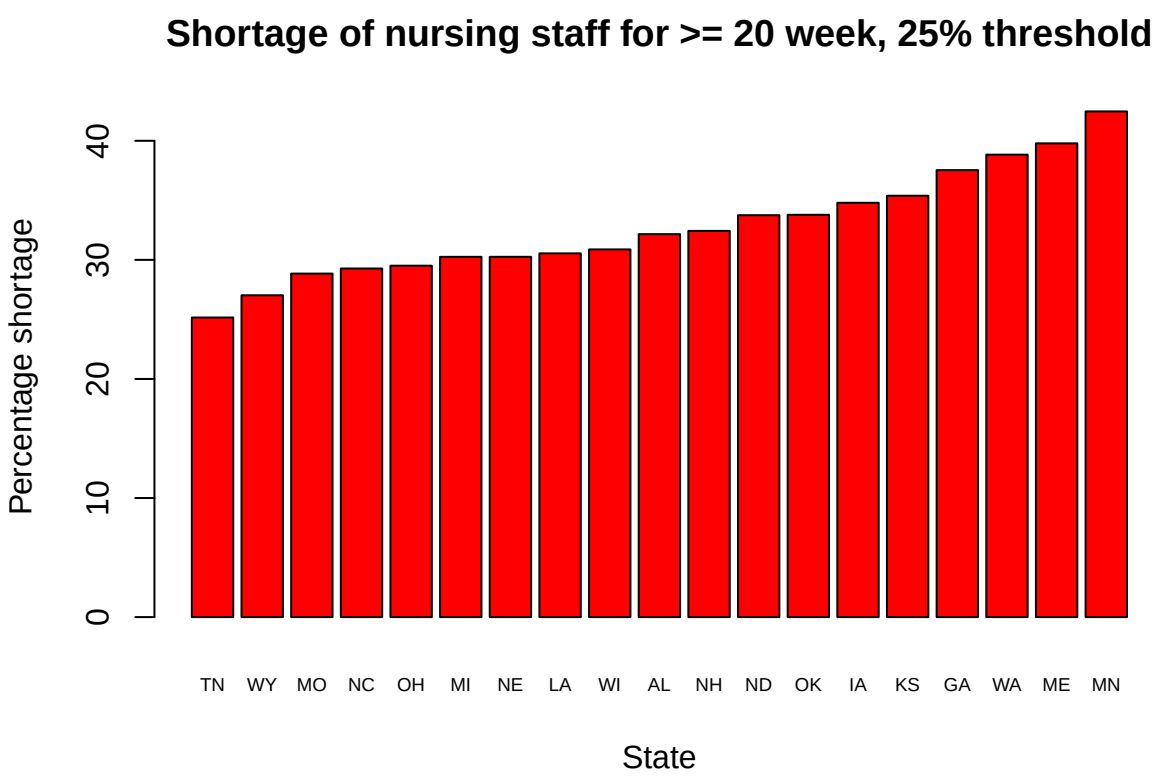
```
# After manual inspection of nursing staff table, a threshhold where the states faced a shortage of nur
table_shortage_of_nursing_staff_25 <- table_shortage_of_nursing_staff[table_shortage_of_nursing_staff$sh

# After manual inspection of clinical staff table, a threshold is hard to come up with. The maximum sho
table_shortage_of_clinical_staff_3 <- table_shortage_of_clinical_staff[table_shortage_of_clinical_staff


# After manual inspection of aides  table, a threshhold where the states faced a shortage of nursing st
table_shortage_of_aides_20 <- table_shortage_of_aides[table_shortage_of_aides$shortage_percent >= 20, ]

# After manual inspection of other staff table, a threshold where the states faced a shortage of other
table_shortage_of_other_staff_20 <- table_shortage_of_other_staff[table_shortage_of_other_staff$shortag
```

> As per analysis and understanding of the given data, a lot of states faced a shortage of greater than or equal to 20% and 25%, which can be considered as the threshhold for a warning situation, we have bifurcated and listed down our observations below.

Observations

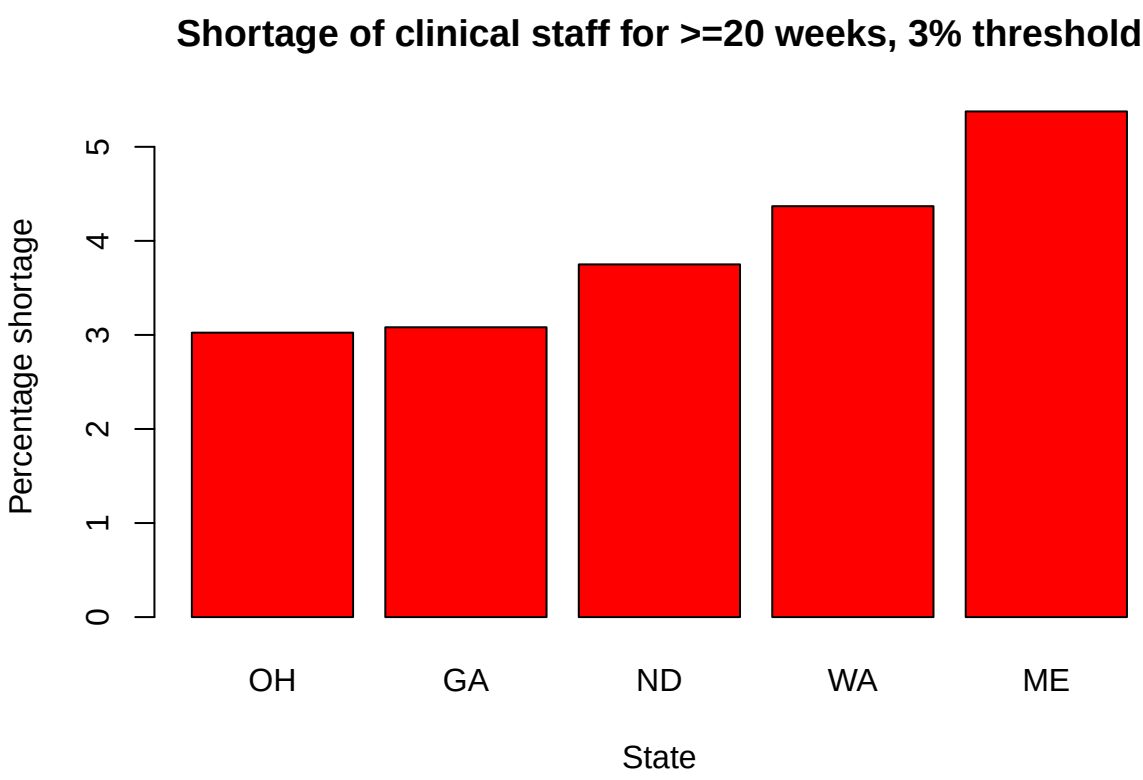
- **NURSING STAFF SHORTAGE**

```
barplot(table_shortage_of_nursing_staff_25$shortage_percent, main = "Shortage of nursing staff for >= 2
```



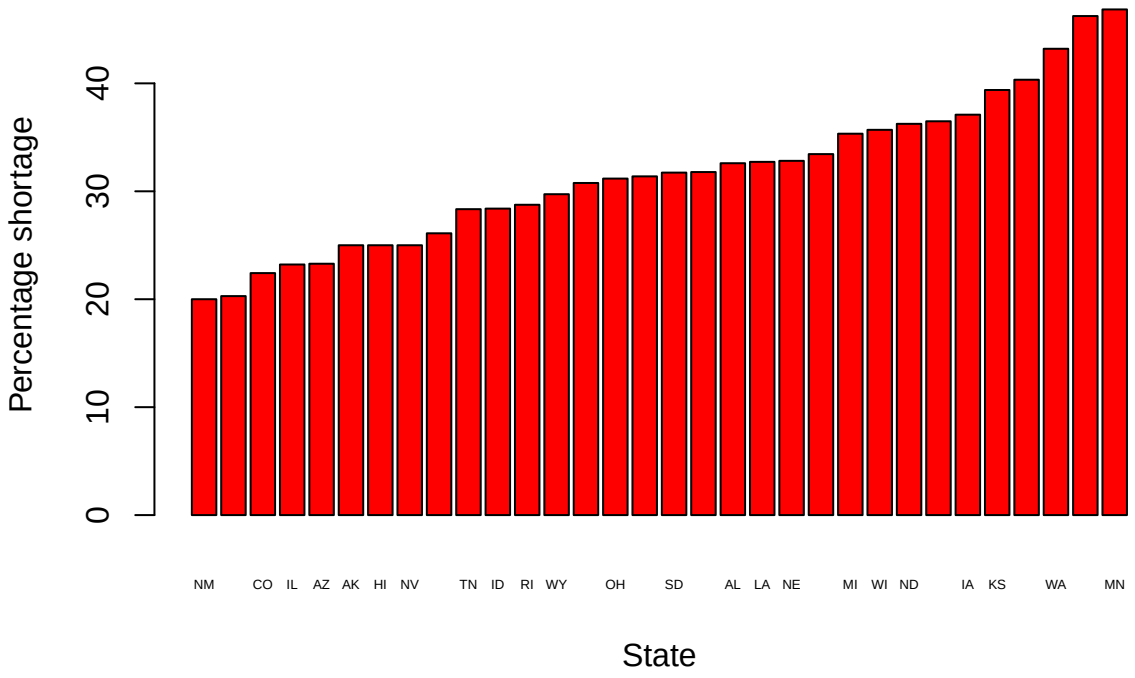
- For nursing staff shortage of more than 20 weeks, considering a threshold of 25% (1/4th of the states missing staff or more), there are 18 states in the given data. The maximum shortage was faced in Minnesota (42%) and the minimum in Wyoming (27%).
- **CLINICAL STAFF SHORTAGE**

```
barplot(table_shortage_of_clinical_staff_3$shortage_percent, main = "Shortage of clinical staff for >=2
```



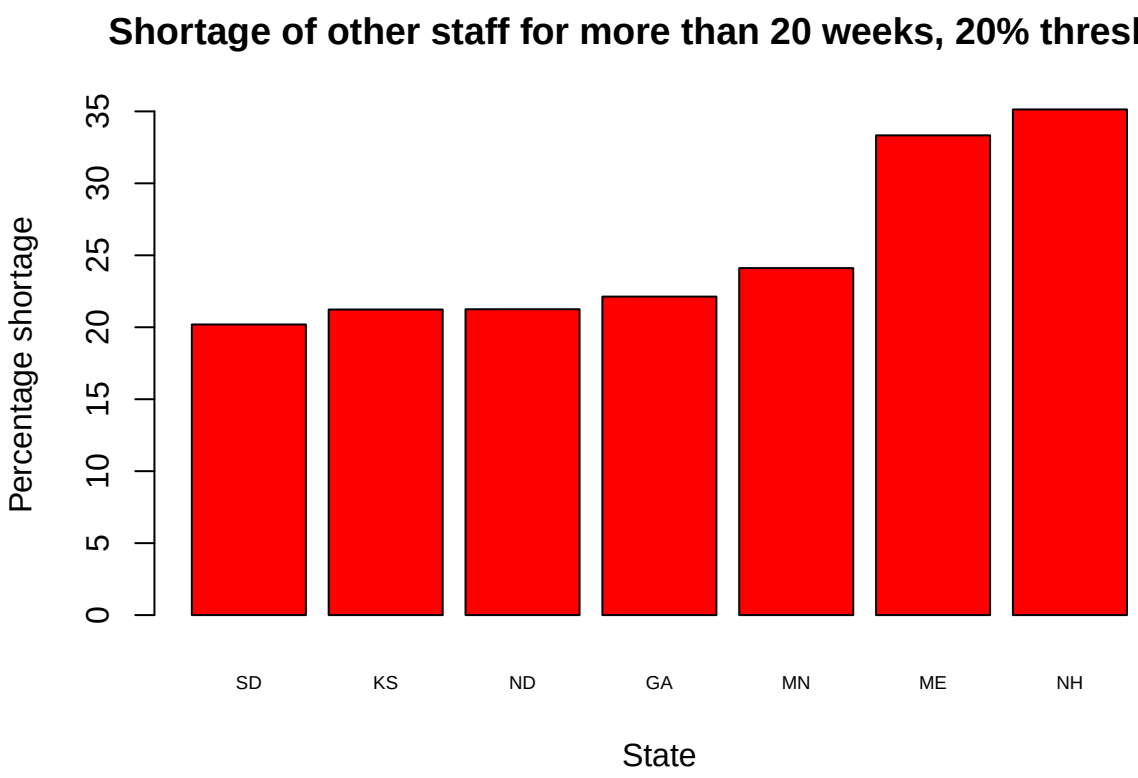
- Clinical staff is critical for treating patients. Fortunately, there were very few states that had a high shortage of clinical staff missing from facilities for more than 20 weeks. Taking a threshold of 3%, The highest shortage of clinical staff was in Maine (5.4%) while the minimum was in Ohio (3%).
- **AIDES STAFF SHORTAGE**

```
barplot(table_shortage_of_aides_20$shortage_percent, main = "Shortage of Aides staff for more than 20 w
```



- Aides staff working in these facilities seem to be the highest affected, considering a threshold of 20% (1/5th of the facilities had missing aides staff or more), there are 31 states in the given data.The maximum shortage was faced in Minnesota (46%) and the minimum in Montana (20%).
- **OTHER STAFF SHORTAGE**

```
barplot(table_shortage_of_other_staff_20$shortage_percent, main = "Shortage of other staff for more thar
```



- Similarly, for other staff, considering a threshold of 20% (1/5th of the facilities in state missing or more), there are 7 states in the given data. The maximum shortage was faced in New Hampshire (35%) and the minimum in South Dakota (20%).

Approach 2

1. Find percentage of Nursing Homes with shortage in a state in week.
2. If percentage is higher than the set threshold, we will consider that week as shortage week for that state.
3. Similar process will be done for all the states for each week. If the total shortage weeks are more than 20 for a state, this state will be included in our answer.

```
## Filter the data and select only relevant columns
covid_df <- df_covid_nursing[, c("week_ending", "federal_provider_number", "provider_state", "shortage_
col_names <- c("shortage_of_nursing_staff", "shortage_of_clinical_staff", "shortage_of_aides", "shortag

## Select the threshold which will be used to nursing homes in a state eligible to be considered for sh
threshold <- c(30, 5, 30, 20)

for (i in 1:length(col_names)) {
  ## Find total number of nursing homes for each state
  state_nursing_home_count <- covid_df %>%
    group_by(provider_state) %>%
    summarise(count = n_distinct(federal_provider_number)) %>%
    drop_na()
```

```r
  ## Filter and get all nursing homes with shortage in a week
  shortage_week_df <- subset(covid_df, covid_df[col_names[i]] == "Y")

  ## Groupby to get the number of nursing homes having a shortage in a state in a week
  shortage_week_df <- shortage_week_df %>%
    group_by(provider_state, week_ending) %>%
    summarise(count = n()) %>%
    drop_na()

  ## Merge the above 2 dataframe and find the % of nursing homes with shortage in a state in a week
  merged_df <- merge(shortage_week_df, state_nursing_home_count, by = c("provider_state"))
  merged_df$Percentage <- (merged_df$count.x / merged_df$count.y) * 100
  # hist(merged_df$Percentage, main = paste("Plot for ",col_names[i]),xlab = "Percentage", ylab="Freque

  ## Filter and get the data of only those states in which has a certain (above set threshold) % of nur
  merged_df <- merged_df[(merged_df$Percentage >= threshold[i]), ]
  merged_df1 <- merged_df %>%
    group_by(provider_state) %>%
    summarise(count_of_weeks = n())

  ## Find number of states having shortage of nursing staff for more than 20 weeks
  answer <- merged_df1[(merged_df1$count_of_weeks > 20), ]
  answer <- answer[order(answer$count_of_weeks, decreasing = TRUE), ]
  colnames(answer) <- c("Provider_State", paste("Count_of_weeks_with_shortage_of", col_names[i], sep =
  print(head(answer, 5))
}
```

```
## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.
```

```
## # A tibble: 5 x 2
##   Provider_State Count_of_weeks_with_shortage_of_shortage_of_nursing_staff
##   <chr>                                                             <int>
## 1 MN                                                                   48
## 2 GA                                                                   47
## 3 ND                                                                   39
## 4 WA                                                                   34
## 5 KS                                                                   23
```

```
## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.
```

```
## # A tibble: 3 x 2
##   Provider_State Count_of_weeks_with_shortage_of_shortage_of_clinical_staff
##   <chr>                                                              <int>
## 1 ME                                                                    39
## 2 AK                                                                    26
## 3 ND                                                                    21
```

```
## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.
```

```
## # A tibble: 5 x 2
##   Provider_State Count_of_weeks_with_shortage_of_shortage_of_aides
```

```
##   <chr>                                                    <int>
## 1 GA                                                         60
## 2 MN                                                         60
## 3 ME                                                         54
## 4 WA                                                         48
## 5 ND                                                         46
```

```
## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.
```

```
## # A tibble: 3 x 2
##   Provider_State Count_of_weeks_with_shortage_of_shortage_of_other_staff
##   <chr>                                                           <int>
## 1 NH                                                                 68
## 2 ME                                                                 48
## 3 MN                                                                 35
```

3.4 Display counties that faced more than 10 weeks shortage of N95 masks (table 1), surgical masks (table 2), eye protection (table 3), supply of gowns (table 4), supply of gloves (table 5),and supply of hand sanitizer (table 6) (20 points)

Approach 1

1. Count number of federal providers available by county by grouping on county and state.
2. Filter for all columns wherever the supply of equipment was not present, and count number of weeks by grouping on state, county and provider number.
3. Join total facilities in county with missing facilities, and calculate percentage of missing facilities per county.
4. Set a threshold based on observed data and write analysis

```
# Used to identify the columns relavant to the data
column_names <- names(df_covid_nursing)

# columns relevant for analysis
relevant_columns <- column_names[c(70, 72, 74, 76, 78, 80)]

# Group and count number of providers by county
county_with_count <- df_covid_nursing %>%
  group_by(provider_state, county) %>%
  summarize(count_of_providers = n_distinct(federal_provider_number))
```

```
## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.
```

```
ind <- 1

for (col_name in relevant_columns) {

  # Filter all values where current column is missing for more than 10 weeks, counting number of federa
  temp <- df_covid_nursing %>%
    filter(df_covid_nursing[col_name] == "N") %>%
    group_by(provider_state, county, federal_provider_number) %>%
    summarize(count_of_shortage = n()) %>%
    filter(count_of_shortage >= 10) %>%
```

```
    group_by(provider_state, county) %>%
    summarize(count_of_missing = n())

  # Join this with state facility count table
  joined_temp <- inner_join(county_with_count, temp, by = c("provider_state", "county"))

  # Calculate shortage percentage
  joined_temp$shortage_percent <- (joined_temp$count_of_missing / joined_temp$count_of_providers) * 100

  # Dynamically assign in new variable
  assign(paste("table", col_name, sep = "_"), joined_temp)

  # Increment index by one for next variable
  ind <- ind + 1
}
```

```
## `summarise()` has grouped output by 'provider_state', 'county'. You can override using the `.groups`
## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.

## `summarise()` has grouped output by 'provider_state', 'county'. You can override using the `.groups`

## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.

## `summarise()` has grouped output by 'provider_state', 'county'. You can override using the `.groups`

## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.

## `summarise()` has grouped output by 'provider_state', 'county'. You can override using the `.groups`

## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.

## `summarise()` has grouped output by 'provider_state', 'county'. You can override using the `.groups`

## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.

## `summarise()` has grouped output by 'provider_state', 'county'. You can override using the `.groups`

## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.
```

```
# After manual inspection of N95 mask table, a threshold where the counties faced a shortage of N95 mas
table_any_current_supply_of_n95_masks_25 <- table_any_current_supply_of_n95_masks[table_any_current_sup

# After manual inspection of surgical mask table, a threshold where the counties faced a shortage of su
table_any_current_supply_of_surgical_masks_25 <- table_any_current_supply_of_surgical_masks[table_any_c

# After manual inspection of Eye protection table, a threshold where the counties faced a shortage of E
table_any_current_supply_of_eye_protection_25 <- table_any_current_supply_of_eye_protection[table_any_c

# After manual inspection of Gowns table, a threshold where the counties faced a shortage of Gowns can 
table_any_current_supply_of_gowns_25 <- table_any_current_supply_of_gowns[table_any_current_supply_of_g
```

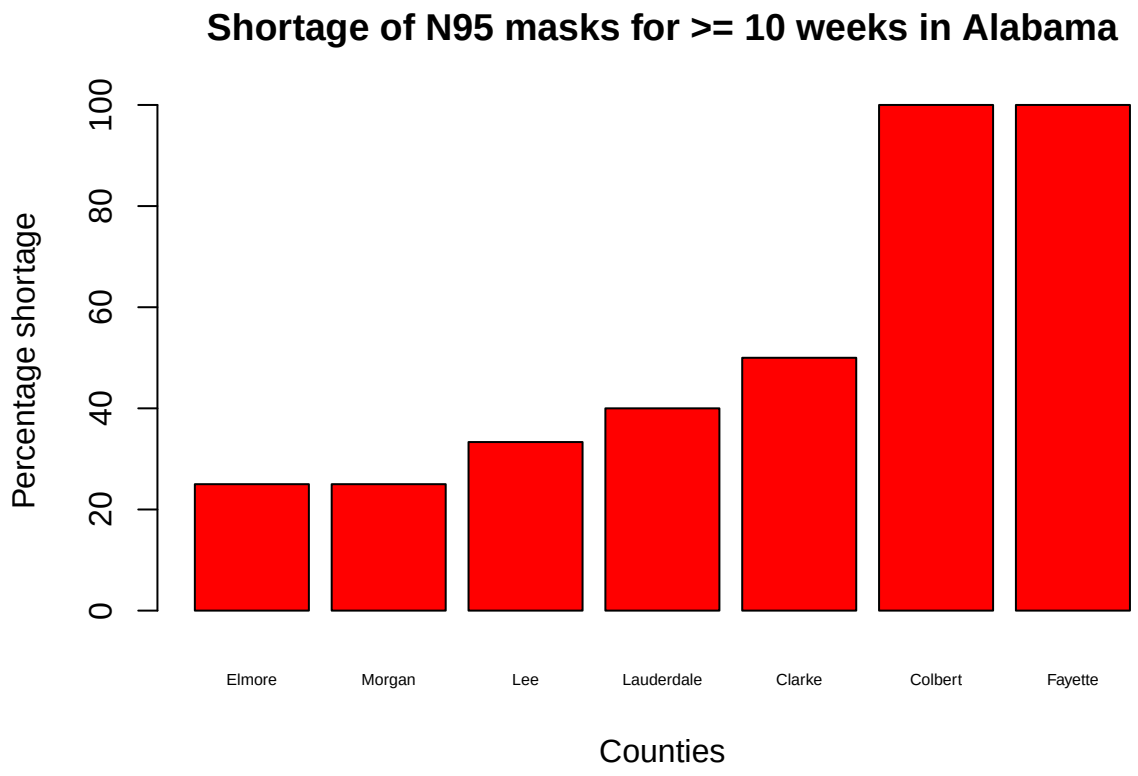
```
# After manual inspection of Gloves table, a threshold where the counties faced a shortage of Gloves ca
table_any_current_supply_of_gloves_25 <- table_any_current_supply_of_gloves[table_any_current_supply_of_

# After manual inspection of Hand Sanitizer table, a threshold where the counties faced a shortage of H
table_any_current_supply_of_hand_sanitizer_25 <- table_any_current_supply_of_hand_sanitizer[table_any_cu
```

> According to our analysis, There were a lot of counties in the dataset which had a shortage of equipment in their facilities for >= 10 week for different equipment. The shortages ranged from 5% facilities in a county facing shortage for >= 10 to every facility in a county facing shortage (for certain equipment).

Observations

- **N95 Mask shortage**

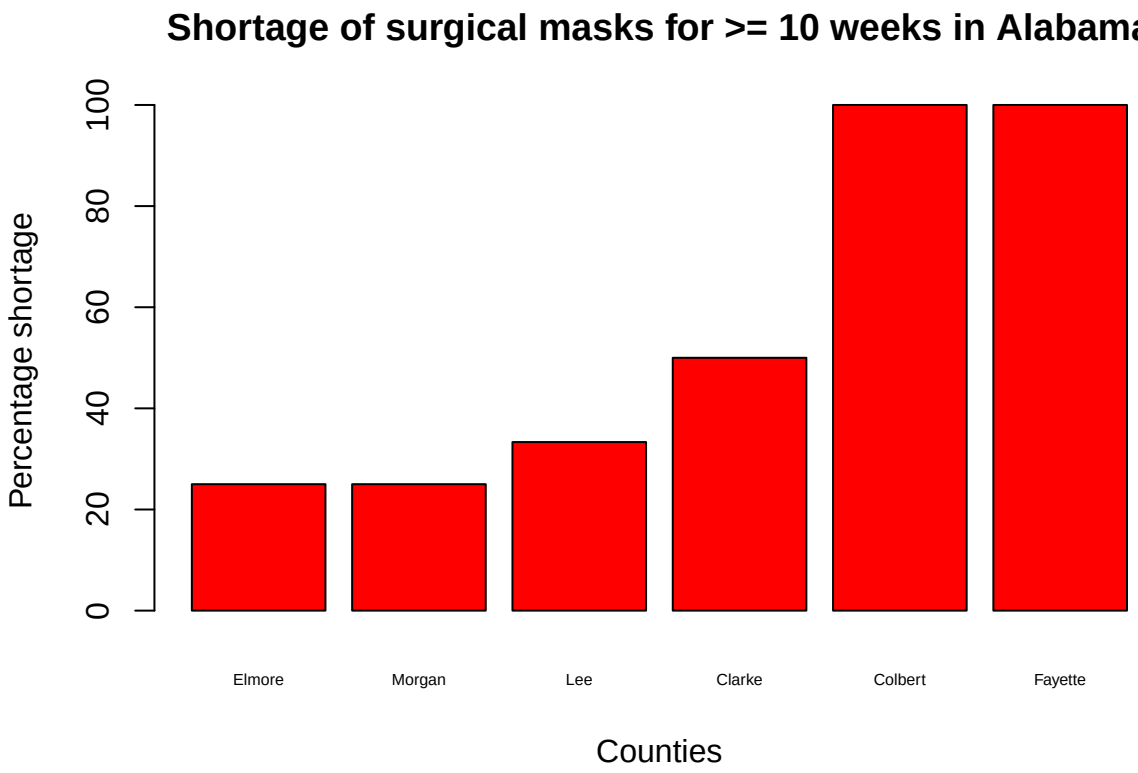
```
AL_counties_shortage <- table_any_current_supply_of_n95_masks_25[(table_any_current_supply_of_n95_masks_

barplot(AL_counties_shortage$shortage_percent, main = "Shortage of N95 masks for >= 10 weeks in Alabama
```



* There are 2897 unique counties in the dataset (some counties seem to be missing from the dataset). Out of these, 293 faced a shortage of >=10 weeks with a threshold set to 25%. Taking the example of the state **Alabama** above, 7 counties in alabama faced a shortage of 25% or more

- **Surgical Mask shortage**

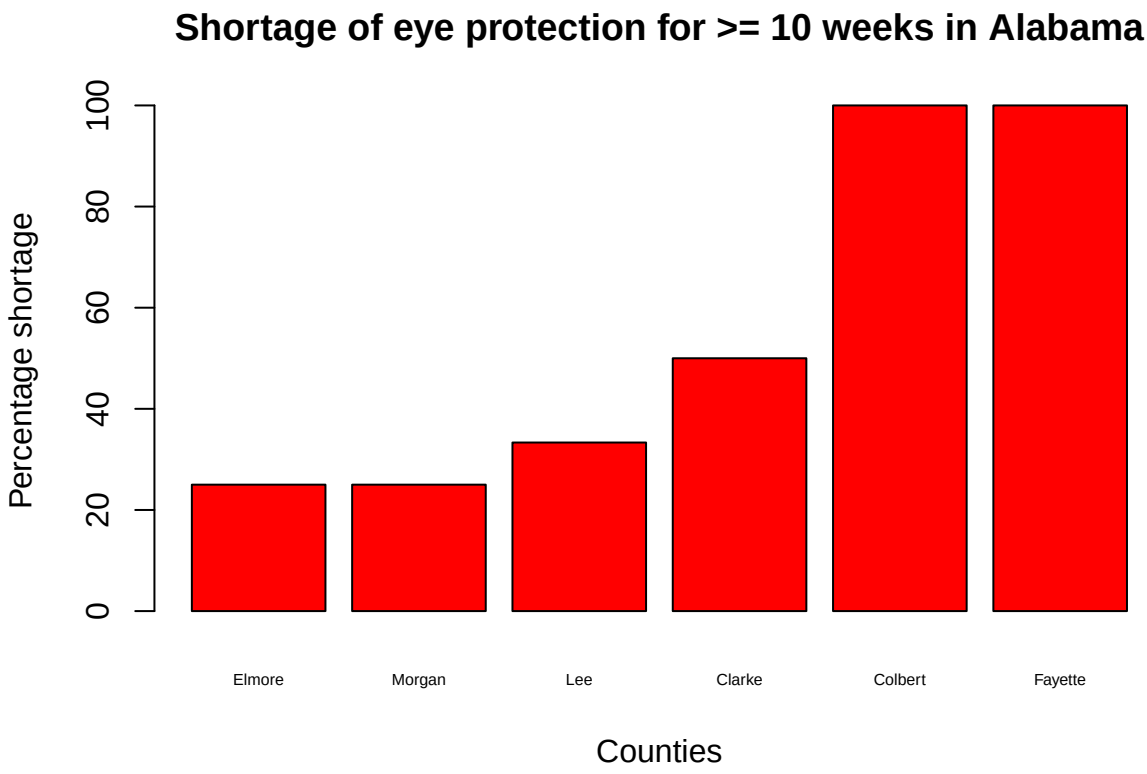
```
AL_counties_shortage_surgical <- table_any_current_supply_of_surgical_masks_25[(table_any_current_supply

barplot(AL_counties_shortage_surgical$shortage_percent, main = "Shortage of surgical masks for >= 10 wee
```



* Out of 2897 unique counties in the dataset, 175 counties faced a shortage of >=10 weeks (threshold = 25%). The plot of Alabama and shortage of surgical masks faced in percentage of facilities is shown above.

- **Eye protection shortage**

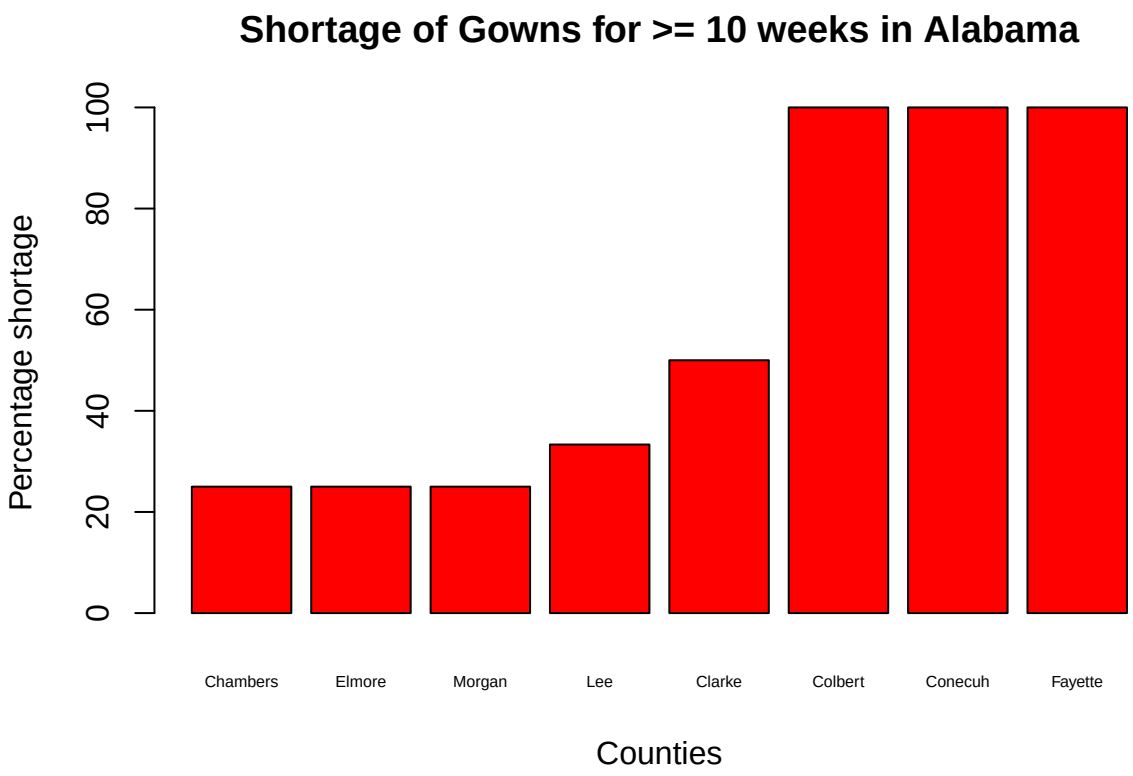
```
AL_counties_shortage_eye_protection <- table_any_current_supply_of_eye_protection_25[(table_any_current_

barplot(AL_counties_shortage_eye_protection$shortage_percent, main = "Shortage of eye protection for >=
```



* Out of 2897 unique counties in the dataset, 157 counties faced a shortage of >=10 weeks for Eye protection (threshold 25%). The plot of Alabama and shortage of eye protection faced in percentage of facilities is shown above. Eye protection and surgical masks seem to have been missing together in the same counties in Alabama.

- **Gowns shortage**

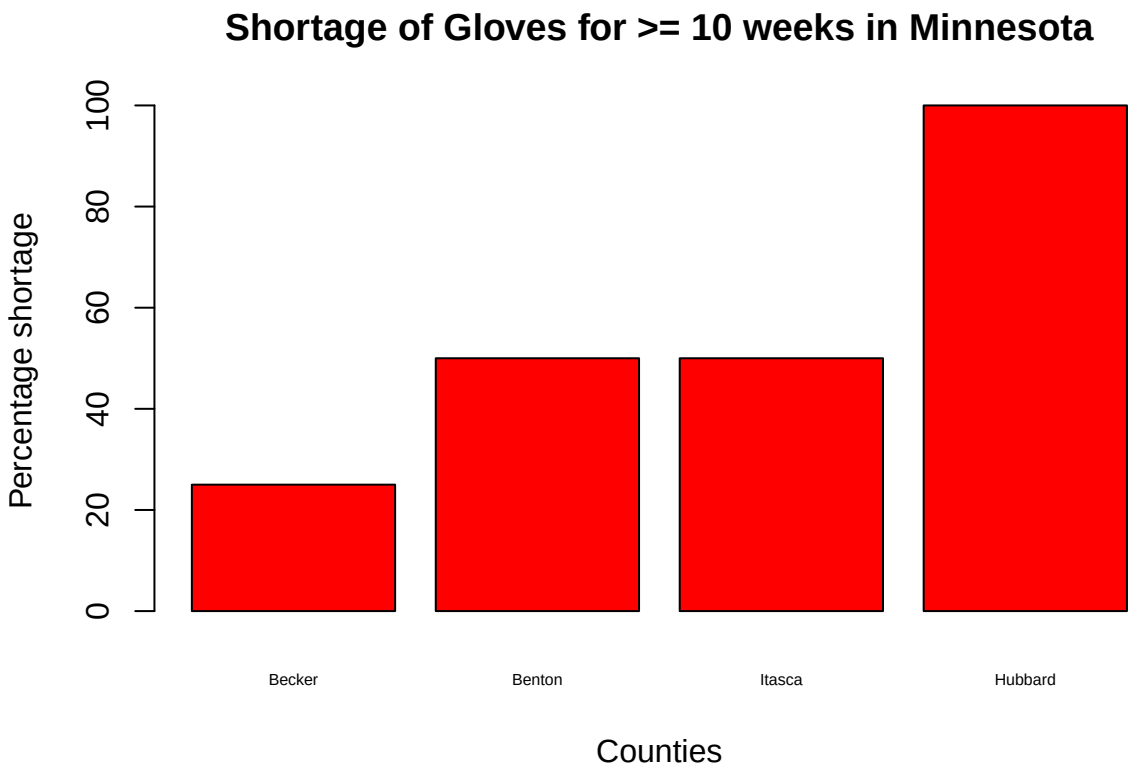
```
AL_counties_shortage_gowns <- table_any_current_supply_of_gowns_25[(table_any_current_supply_of_gowns_25

barplot(AL_counties_shortage_gowns$shortage_percent, main = "Shortage of Gowns for >= 10 weeks in Alaba
```



* Out of 2897 unique counties in the dataset, 134 counties faced a shortage of >=10 weeks for gowns (threshold = 25%). The plot of Alabama and shortage of gowns faced in percentage of facilities is shown above.

- **Gloves shortage**

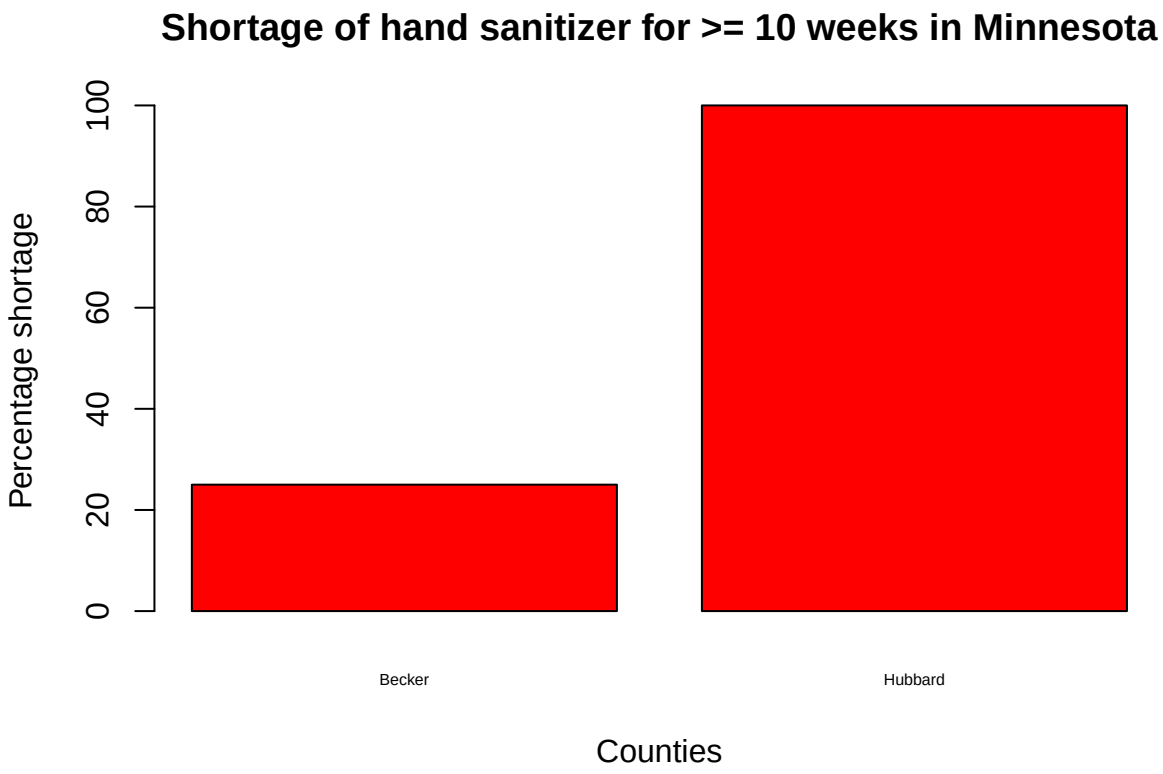
```
MN_counties_shortage_gloves <- table_any_current_supply_of_gloves_25[(table_any_current_supply_of_gloves

barplot(MN_counties_shortage_gloves$shortage_percent, main = "Shortage of Gloves for >= 10 weeks in Min
```



* Out of 2897 unique counties in the dataset, 14 counties faced a shortage of >=10 weeks for gloves (threshold = 25%). The plot of counties in Minnesota and shortage of gloves faced in percentage of facilities is shown above.

- **Hand Sanitizer shortage**

```
MN_counties_shortage_hand_sanitizer <- table_any_current_supply_of_hand_sanitizer_25[(table_any_current_

barplot(MN_counties_shortage_hand_sanitizer$shortage_percent, main = "Shortage of hand sanitizer for >=
```



* Out of 2897 unique counties in the dataset, 12 counties faced a shortage of >=10 weeks for hand sanitizers (threshold 25%).The plot of counties in Minnesota and shortage of sanitizer faced in percentage of facilities is shown above.

Approach 2

1. Find the pecentage of nursing Homes with shortage in a state-county in a week. 2. If percentage is higher than the set threshold, we will consider that week as shortage week for that state-county. Similar process will be done for all the state-counties for each week.
2. If the total shortage weeks are more than 20 for a state-county, this state-county will be included in our answer.

```
## Filter the data and select only relevant columns
covid_df <- df_covid_nursing[, c("week_ending", "federal_provider_number", "provider_state", "county",

col_names <- c("any_current_supply_of_eye_protection", "any_current_supply_of_n95_masks", "any_current_s

## Select the threshold which will be used to nursing homes in a state-county eligible to be considered
threshold <- c(40, 40, 40, 40, 40, 40)

for (i in 1:length(col_names)) {
  ## Find total number of nursing homes for each state-county
  state_nursing_home_count <- covid_df %>%
    group_by(provider_state, county) %>%
    summarise(count = n_distinct(federal_provider_number)) %>%
    drop_na()
```

```
  ## Filter and get all nursing homes with shortage in a week
  shortage_week_df <- subset(covid_df, covid_df[col_names[i]] == "N")

  ## Groupby to get the number of nursing homes having a shortage in a state-county in a week
  shortage_week_df <- shortage_week_df %>%
    group_by(provider_state, county, week_ending) %>%
    summarise(count = n()) %>%
    drop_na()

  ## Merge the above 2 dataframe and find the % of nursing homes with shortage in a state in a week
  merged_df <- merge(shortage_week_df, state_nursing_home_count, by = c("provider_state", "county"))
  merged_df$Percentage <- (merged_df$count.x / merged_df$count.y) * 100

  ## Plot the histogram
  # hist(merged_df$Percentage, main = paste("Plot for",col_names[i]), xlab = "Percentage", ylab="Freque

  ## Filter and get the data of only those states in which has a certain (above set threshold) % of nurs
  merged_df <- merged_df[(merged_df$Percentage >= threshold[i]), ]
  merged_df1 <- merged_df %>%
    group_by(provider_state, county) %>%
    summarise(count_of_weeks = n())

  ## find counties having shortage of nursing staff for more than 10 weeks
  answer <- merged_df1[(merged_df1$count_of_weeks > 10), ]
  answer <- answer[order(answer$count_of_weeks, decreasing = TRUE), ]
  colnames(answer) <- c("Provider_State", "County", paste("Count_of_weeks_with_shortage_of", col_names[i
  print(head(answer, 5))
}
```

```
## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.
```

```
## `summarise()` has grouped output by 'provider_state', 'county'. You can override using the `.groups`
```

```
## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.
```

```
## # A tibble: 5 x 3
## # Groups:   Provider_State [4]
##   Provider_State County   Count_of_weeks_with_shortage_of_any_current_supply_of~
##   <chr>          <chr>                                                    <int>
## 1 SC             Newberry                                                    34
## 2 PA             Bedford                                                     33
## 3 SD             Davison                                                     33
## 4 MN             Dodge                                                       30
## 5 MN             Hubbard                                                     30
```

```
## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.
```

```
## `summarise()` has grouped output by 'provider_state', 'county'. You can override using the `.groups`
```

```
## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.
```

```
## # A tibble: 5 x 3
## # Groups:   Provider_State [5]
##   Provider_State County    Count_of_weeks_with_shortage_of_any_current_supply_o~
##   <chr>          <chr>                                                    <int>
## 1 AK             Ketchikan                                                   42
## 2 NY             Jefferson                                                   42
## 3 MN             Hubbard                                                     38
## 4 IA             Audubon                                                     36
## 5 MI             Midland                                                     35

## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.

## `summarise()` has grouped output by 'provider_state', 'county'. You can override using the `.groups`

## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.

## # A tibble: 5 x 3
## # Groups:   Provider_State [5]
##   Provider_State County     Count_of_weeks_with_shortage_of_any_current_supply_~
##   <chr>          <chr>                                                    <int>
## 1 IA             Audubon                                                     36
## 2 CO             Baca                                                        34
## 3 AL             Fayette                                                     33
## 4 KY             Rockcastle                                                  33
## 5 PA             Bedford                                                     33

## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.

## `summarise()` has grouped output by 'provider_state', 'county'. You can override using the `.groups`

## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.

## # A tibble: 5 x 3
## # Groups:   Provider_State [4]
##   Provider_State County     Count_of_weeks_with_shortage_of_any_current_supply_~
##   <chr>          <chr>                                                    <int>
## 1 CO             Baca                                                        34
## 2 AL             Fayette                                                     33
## 3 TN             Benton                                                      33
## 4 TN             Hardeman                                                    33
## 5 VA             Gloucester                                                  33

## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.

## `summarise()` has grouped output by 'provider_state', 'county'. You can override using the `.groups`

## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.

## # A tibble: 5 x 3
## # Groups:   Provider_State [4]
##   Provider_State County   Count_of_weeks_with_shortage_of_any_current_supply_of~
```

```
##   <chr>          <chr>                                                <int>
## 1 CO             Baca                                                    33
## 2 LA             Grant                                                   31
## 3 MN             Hubbard                                                 30
## 4 MI             Chippewa                                                17
## 5 MN             Benton                                                  16
```

```
## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.
```

```
## `summarise()` has grouped output by 'provider_state', 'county'. You can override using the `.groups`
```

```
## `summarise()` has grouped output by 'provider_state'. You can override using the `.groups` argument.
```

```
## # A tibble: 5 x 3
## # Groups:   Provider_State [5]
##   Provider_State County   Count_of_weeks_with_shortage_of_any_current_supply_of~
##   <chr>          <chr>                                                   <int>
## 1 LA             Grant                                                      30
## 2 MN             Hubbard                                                    30
## 3 CO             Baca                                                       29
## 4 KS             Meade                                                      20
## 5 MI             Chippewa                                                   17
```

Question 3.5 - Display all the nursing home facilities that experienced a shortage of ventilator supplies for more than 10 weeks

```
covid_df <- df_covid_nursing[, c("week_ending", "federal_provider_number", "provider_name", "provider_s

col_names <- c("any_current_supply_of_ventilator_supplies")

for (i in 1:length(col_names)) {
  shortage_week_df <- subset(covid_df, covid_df[col_names[i]] == "N")

  shortage_week_df <- shortage_week_df %>%
    group_by(federal_provider_number, provider_name) %>%
    summarise(number_of_shortage_weeks = n()) %>%
    drop_na()

  ## find counties having shortage of ventilator supplies for more than 10 weeks
  answer <- shortage_week_df[(shortage_week_df$number_of_shortage_weeks > 10), ]
  answer <- answer[order(answer$number_of_shortage_weeks, decreasing = TRUE), ]
  print(head(answer, 5))
}
```

```
## `summarise()` has grouped output by 'federal_provider_number'. You can override using the `.groups` a
```

```
## # A tibble: 5 x 3
## # Groups:   federal_provider_number [5]
##   federal_provider_number provider_name                    number_of_shortage_~
##   <chr>                   <chr>                                           <int>
```

```
## 1 365033                CEDARWOOD PLAZA                                58
## 2 165397                HEARTLAND CARE CENTER                          51
## 3 365672                AUTUMN HILLS CARE CENTER                       48
## 4 366417                FOUR SEASONS OF WASHINGTON NURSI~              48
## 5 015178                GREENE COUNTY NURSING HOME                     26
```