FinalProject

Ajay Karatam and Michael Rothschilds

5/13/2020

An Analysis of U.S Hospital Bed Capacity During the COVID-19 Pandemic

by Michael Rothschilds and Ajay Karatam

Introduction:

The COVID-19 pandemic has altered the lives of millions over the past few months. People have practiced Social Distancing at an unprecedented level with the intention of keeping people safe and “flattening the curve”. The purpose of “flattening the curve” is to prevent hospitals from being overun with too many COVID-19 patients at once. In order to stop this from happening, public health and government officials need to project the number of cases, the rate of hospitilization, and the resulting number of hospital and ICU beds needed at any given time across all areas of the United States. The following data was collected from a team of researchers at the Harvard Global Data Institute. They made projections for different infection rates and used surveys conducted by The American Hospital Association to predict the true availability of hospital and ICU beds.

Examine the original dataset here:

https://www.kaggle.com/mrmorj/hospital-bed-capacity-and-covid19 (https://www.kaggle.com/mrmorj/hospital-bed-capacity-and-covid19)

R Libraries Used

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------------------------------------------------- tidyverse 1.3.0
--
```

```
## v ggplot2 3.3.0     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.5
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
```

```
## Warning: package 'ggplot2' was built under R version 3.6.3
```

```
## Warning: package 'dplyr' was built under R version 3.6.3
```

```
## -- Conflicts ------------------------------------------------------------------------------------ tidyverse_conflicts()
--
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:base':
##
##     date
```

```
library(dplyr)
library(broom)
library(leaflet)
```

```
## Warning: package 'leaflet' was built under R version 3.6.3
```

Part 1: Dataframe Setup and Tidying

We begin our setup by downloading the dataset into our local directory and opening here as a .csv file. Some columns that required separation were split using regex like the ones seen below. This process also included re-arranging as well as omitting some columns for the sake of readability. Furthermore, we added two more columns that served as a mathematical computation of two other columns. The final tidy dataframe is setup as “df” which is what we will be using later on in the analysis.

```
csv_file <- "HRR_Scorecard.csv"
hcb <- read_csv(csv_file)
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   `Total Hospital Beds` = col_number(),
##   `Total ICU Beds` = col_number(),
##   `Available Hospital Beds` = col_number(),
##   `Potentially Available Hospital Beds*` = col_number(),
##   `Available ICU Beds` = col_double(),
##   `Potentially Available ICU Beds*` = col_number(),
##   `Adult Population` = col_number(),
##   `Population 65+` = col_number(),
##   `Projected Infected Individuals` = col_number(),
##   `Projected Hospitalized Individuals` = col_number(),
##   `Projected Individuals Needing ICU Care` = col_number(),
##   `Hospital Beds Needed, Six Months` = col_number(),
##   `Hospital Beds Needed, Twelve Months` = col_number(),
##   `Hospital Beds Needed, Eighteen Months` = col_number(),
##   `ICU Beds Needed, Six Months` = col_number(),
##   `ICU Beds Needed, Twelve Months` = col_number(),
##   `ICU Beds Needed, Eighteen Months` = col_number()
## )
```

```
## See spec(...) for full column specifications.
```

```
#delete the first entry since it contains garbage values
hcb = hcb[-1,]
df <- hcb %>%
  #choose the relevant columns that we want to work with from hcb
  select(1:12)
df
```

```
## # A tibble: 305 x 12
##    HRR   `Total Hospital~ `Total ICU Beds` `Available Hosp~ `Potentially Av~
##    <chr>            <dbl>            <dbl>            <dbl>            <dbl>
##  1 Abil~              980              127              565              772
##  2 Akro~             1358              186              518              938
##  3 Alam~             2695              293              665             1680
##  4 Alba~              704               60              221              462
##  5 Alba~             4804              425             1579             3191
##  6 Albu~             2908              380             1102             2005
##  7 Alex~              917               43              402              660
##  8 Alle~             3267              334             1267             2267
##  9 Alto~              555               61              234              394
## 10 Amar~             1236              194              678              957
## # ... with 295 more rows, and 7 more variables: `Available ICU Beds` <dbl>,
## #   `Potentially Available ICU Beds*` <dbl>, `Adult Population` <dbl>,
## #   `Population 65+` <dbl>, `Projected Infected Individuals` <dbl>, `Projected
## #   Hospitalized Individuals` <dbl>, `Projected Individuals Needing ICU
## #   Care` <dbl>
```

```
#extract the state as a separate column from HRR
df$State <- str_extract(df$HRR, "([A-Z]{2})")

#extract the town as a separate column from HRR
df$Town <- sub(", [A-Z]{2}$", "", df$HRR)

#re-arrange the columns to make the data more presentable
df <- df[c(14,13,2,4,5,3,6,7,8,9,10,11,12)]

#calculate the percentage of occupied hospital beds
df$calc_hospital = ((df[c(4)] / df[c(3)])*100)
df$`Occupied Hospital Beds percentage` <- round(df$calc_hospital$`Available Hospital Beds`,digits=2)

#calculate the percent of occupied ICU beds
df$calc_ICU = ((df[c(7)] / df[c(6)])*100)
df$`Occupied ICU Beds percentage` <- round(df$calc_ICU$`Available ICU Beds`, digits=2)

#final columns re-arrangement
df <- df[c(1,2,3,4,5,15,6,7,8,17,9,10,11,12,13)]
head(df)
```

```
## # A tibble: 6 x 15
##   Town  State `Total Hospital~ `Available Hosp~ `Potentially Av~
##   <chr> <chr>            <dbl>            <dbl>            <dbl>
## 1 Abil~ TX                 980              565              772
## 2 Akron OH                1358              518              938
## 3 Alam~ CA                2695              665             1680
## 4 Alba~ GA                 704              221              462
## 5 Alba~ NY                4804             1579             3191
## 6 Albu~ NM                2908             1102             2005
## # ... with 10 more variables: `Occupied Hospital Beds percentage` <dbl>, `Total
## #   ICU Beds` <dbl>, `Available ICU Beds` <dbl>, `Potentially Available ICU
## #   Beds*` <dbl>, `Occupied ICU Beds percentage` <dbl>, `Adult
## #   Population` <dbl>, `Population 65+` <dbl>, `Projected Infected
## #   Individuals` <dbl>, `Projected Hospitalized Individuals` <dbl>, `Projected
## #   Individuals Needing ICU Care` <dbl>
```

==

Part 2 Data Analysis With a Focus on Hospital and ICU Bed Capacity

Given that our data deals with a statistical analysis of the hospital beds and ICU beds availability as well as population age distribtion across the top 300 US hospital markets; we decided to split out analysis into 3 parts to offer a detailed story.

Step 2.1: Hospital and ICU Dataframe Setup

Most of the attributes for the Hospital and ICU are very similar ex: 'Hospital Beds Availble', 'ICU Beds Available' or 'Total Hospital Beds', 'Total ICU beds'. It made sense to pull the relevant attributes from the tidy dataframe (df) and used it in our analysis for the Hospital and ICU. In the process of setting up this new dataframe, we decided to combine the regions for every state as one entry, we achieved this by adding up all the attributes ex: 'Total Hospital Beds', 'Total ICU Beds'. That way we end up with a more interpretable dataframe of 51 entries and 10 columns. Combining the attributes helps us generalize the analysis to each state rather than every region and this was what we were aiming to do with this project to begin with. The final two columns that I added were the rate of hospital beds availablity and rate of icu beds availability, both of these attributes help us in understanding how each state's hospital and icu ward compare with one another.

```
beds_df <- df %>%
  select(1:10)%>%
  group_by(State)%>%
  summarize(`No. of Regions` = n_distinct(Town),
            `Total Hospital Beds` = sum(`Total Hospital Beds`),
            `Hospital Beds Available` = sum(`Available Hospital Beds`),
            `Potential Hospital Beds Available` = sum(`Potentially Available Hospital Beds*`),
            `Total ICU Beds` = sum(`Total ICU Beds`),
            `ICU Beds Available` = sum(`Available ICU Beds`),
            `Potential ICU Beds Available` = sum(`Potentially Available ICU Beds*`)
            )%>%
  mutate(`Rate of Hospital beds availability` = (`Hospital Beds Available`/`Total Hospital Beds`)*100)%>%
  mutate(`Rate of ICU beds availability` = (`ICU Beds Available`/`Total ICU Beds`)*100)%>%
  select(1,2,3,4,5,9,6,7,8,10)%>%
  arrange(State)
beds_df
```

```
## # A tibble: 51 x 10
##    State `No. of Regions` `Total Hospital~ `Hospital Beds ~ `Potential Hosp~
##    <chr>            <int>            <dbl>            <dbl>            <dbl>
##  1 AK                   1             1583              533             1058
##  2 AL                   6            14793             5282            10037
##  3 AR                   5             8560             4063             6311
##  4 AZ                   4            12590             4763             8676
##  5 CA                  24            68074            22585            45328
##  6 CO                   7            10335             4417             7376
##  7 CT                   3             7034             1731             4382
##  8 DC                   1             5055             1595             3325
##  9 DE                   1             1845              601             1223
## 10 FL                  17            53744            18464            36105
## # ... with 41 more rows, and 5 more variables: `Rate of Hospital beds
## #   availability` <dbl>, `Total ICU Beds` <dbl>, `ICU Beds Available` <dbl>,
## #   `Potential ICU Beds Available` <dbl>, `Rate of ICU beds availability` <dbl>
```

```
beds_df$`Rate of Hospital beds availability` = round(beds_df$`Rate of Hospital beds availability`, digits=2)
beds_df$`Rate of ICU beds availability` = round(beds_df$`Rate of ICU beds availability`, digits=2)
```

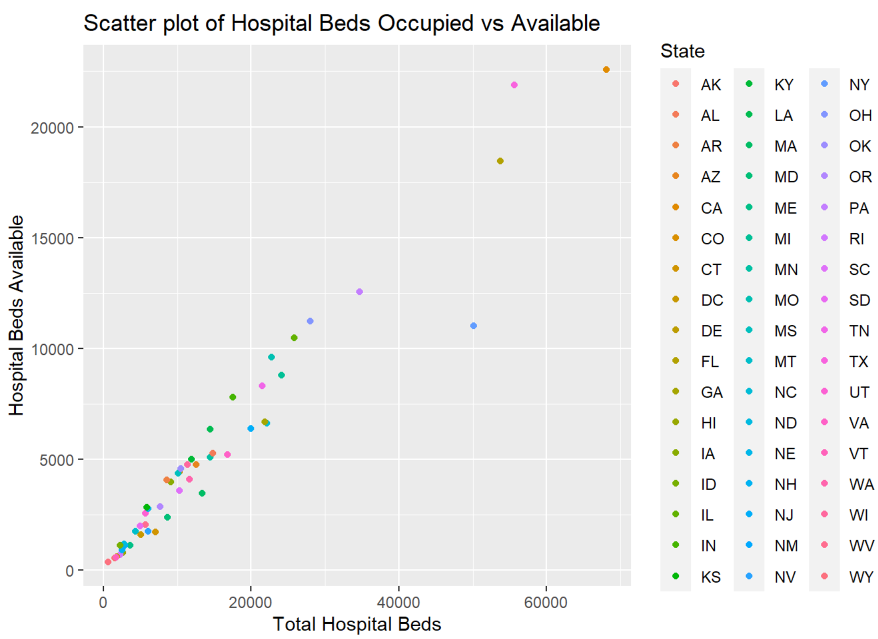
Step 2.2: Hospital Analysis - Scatter plot of Hospital Beds Occupied vs Available

To interprete this graph, we know that Hospital Beds Available <= Total Beds Available as a universal truth, therefore, a point which is higher on the y-axis is an indication that there are more beds available. Additionally, the closer the x axis is to 0 and a higher y axis point means that the hospital is operating extremely efficiently. From this plot, there seems be more concentration of points around x <= 20,000 and y <= 5,000; this means that roughly 25% hospital beds are available for most of these states. Ofcourse there are a few states which have more total beds but a higher ratio of occupied beds (>25%).

```
hosp_df <- beds_df %>%
  select(1:6)%>%
  group_by(State, `No. of Regions`)%>%
  arrange(`Rate of Hospital beds availability`)
hosp_df
```

```
## # A tibble: 51 x 6
## # Groups:   State, No. of Regions [51]
##    State `No. of Regions` `Total Hospital~ `Hospital Beds ~ `Potential Hosp~
##    <chr>            <int>            <dbl>            <dbl>            <dbl>
##  1 NY                  10            50102            11003            30554
##  2 CT                   3             7034             1731             4382
##  3 MA                   3            13352             3473             8412
##  4 MD                   3             8710             2368             5539
##  5 NV                   2             6051             1748             3899
##  6 NC                   9            22158             6610            14384
##  7 HI                   1             2623              795             1709
##  8 GA                   7            21861             6681            14270
##  9 ME                   2             3618             1110             2364
## 10 RI                   1             2249              690             1470
## # ... with 41 more rows, and 1 more variable: `Rate of Hospital beds
## #   availability` <dbl>
```

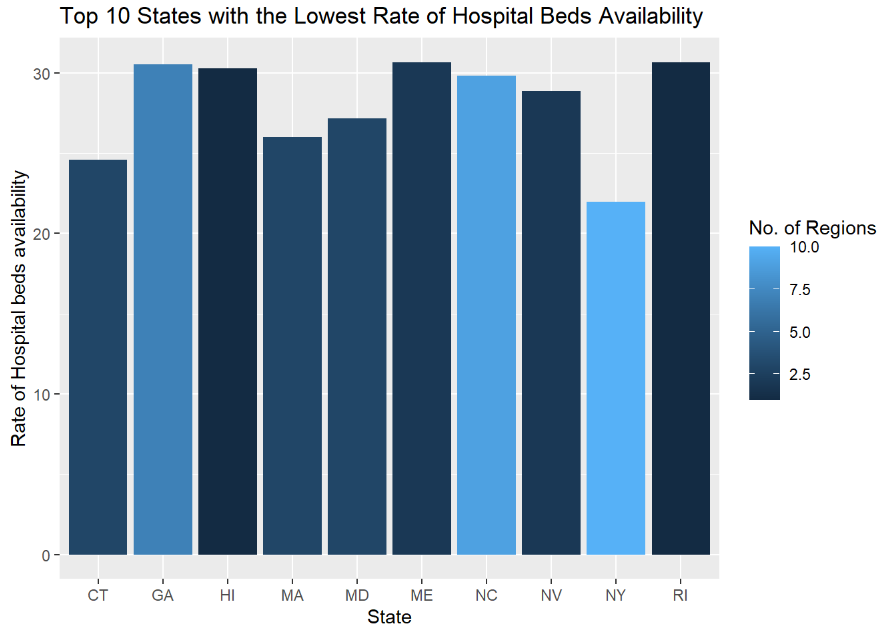
```
ggplot(hosp_df, mapping=aes(x=`Total Hospital Beds`, y=`Hospital Beds Available`))+
  geom_point(mapping=aes(color = State))+
  ggtitle("Scatter plot of Hospital Beds Occupied vs Available")
```



Step 2.3: Hospital Analysis - Bargraph of the Top 10 States with the Lowest Rate of Hospital Beds Availability

The states included in this graph gives the reader an understanding of the population demographics for these regions, it is easy to guess that there could be a significant older adult population. Also the number of regions for each of these states is another indicator of the intensity of beds occupancy. From the graph, New York has the most hospitals as well as the least availability, Rhode Island also has fewer hospitals and a tad lower availability rate. An interesting feature about the states in this list is the fact that most of them belong in the East coast.

```
hosp_df[1:10,] %>%
  ggplot(mapping=aes(x=State, y=`Rate of Hospital beds availability`, fill= `No. of Regions`))+
  geom_col(mapping=aes())+
  ggtitle("Top 10 States with the Lowest Rate of Hospital Beds Availability")
```

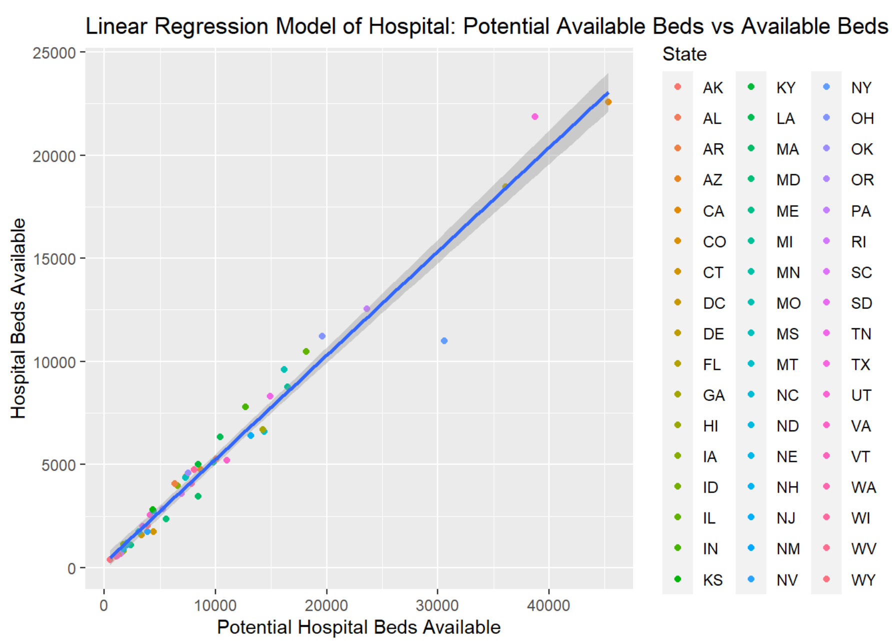


Step 2.4: Hospital Analysis - Linear Regression Model of Hospital: Potential Available Beds vs Available Beds

This regression plot features a unique attribute which is “Potential Available Beds”, this attribute was part of our original dataset and it is a numerical value that represents the scenrario if non-covid patients took up 50% less beds. By plotting this against the current available beds, the regression analysis will help us understand the correlation. Off the bat, it seems like the concentration lies around lower x and y values. The stright linear regression curve is a strong indication that many hospitals can promise 50% more hospital beds.

```
ggplot(hosp_df, mapping=aes(x=`Potential Hospital Beds Available`, y=`Hospital Beds Available`))+
  geom_point(mapping=aes(color = State))+
  geom_smooth(method=lm)+
  ggtitle("Linear Regression Model of Hospital: Potential Available Beds vs Available Beds")
```

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



##

==

Part 3: ICU Analysis

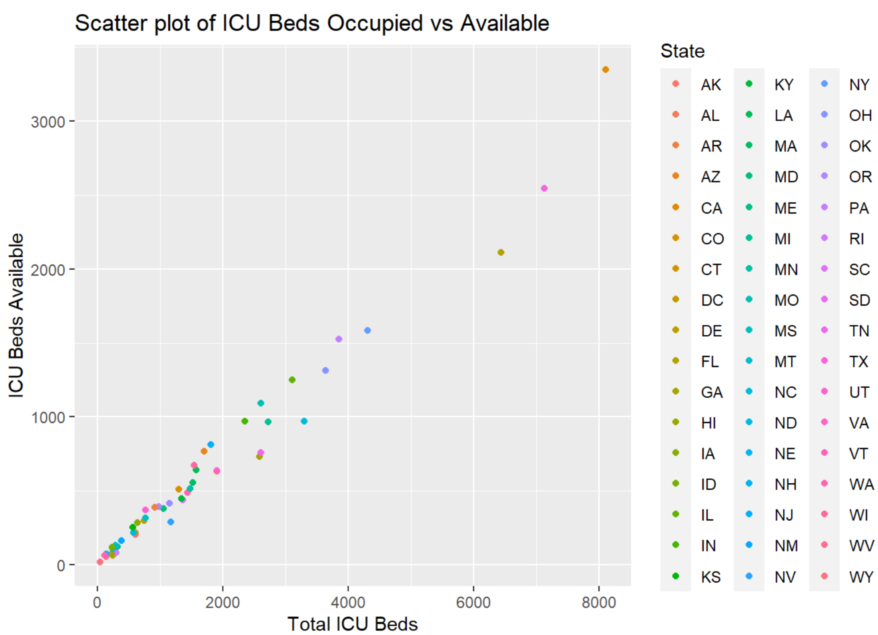
Step 3.1: Scatter plot of ICU Beds Occupied vs Available

ICU beds occupancy is an interesting factor to observe simply because of its dependent nature, in other words, ICU beds occupancy is dependant on demographic factors like number of old people or number of hospitals in a given state. From the graph below, similar to the results from the Hospital beds version of this graph, has a lower concentration at lower x and y values. From color inspection is seems like most states are maintaining similar hospital beds and icu beds availability rate. The states within the concetration seem to show that roughy 50% of the icu beds are available. States with more hospital beds show lower availability rate (~30%).

```
icu_df <- beds_df %>%
  select(1,2,7,8,9,10)%>%
  group_by(State, `No. of Regions`)%>%
  arrange(`Rate of ICU beds availability`)
icu_df
```

```
## # A tibble: 51 x 6
## # Groups:   State, No. of Regions [51]
##    State `No. of Regions` `Total ICU Beds` `ICU Beds Avail~ `Potential ICU ~
##    <chr>            <int>            <dbl>            <dbl>            <dbl>
##  1 NV                   2             1167              290              729
##  2 DE                   1              237               62              149
##  3 RI                   1              293               82              188
##  4 GA                   7             2582              730             1656
##  5 TN                   7             2601              760             1679
##  6 NC                   9             3294              970             2133
##  7 SC                   5             1358              437              897
##  8 FL                  17             6433             2113             4269
##  9 AL                   6             1903              633             1268
## 10 HI                   1              219               73              146
## # ... with 41 more rows, and 1 more variable: `Rate of ICU beds
## #   availability` <dbl>
```

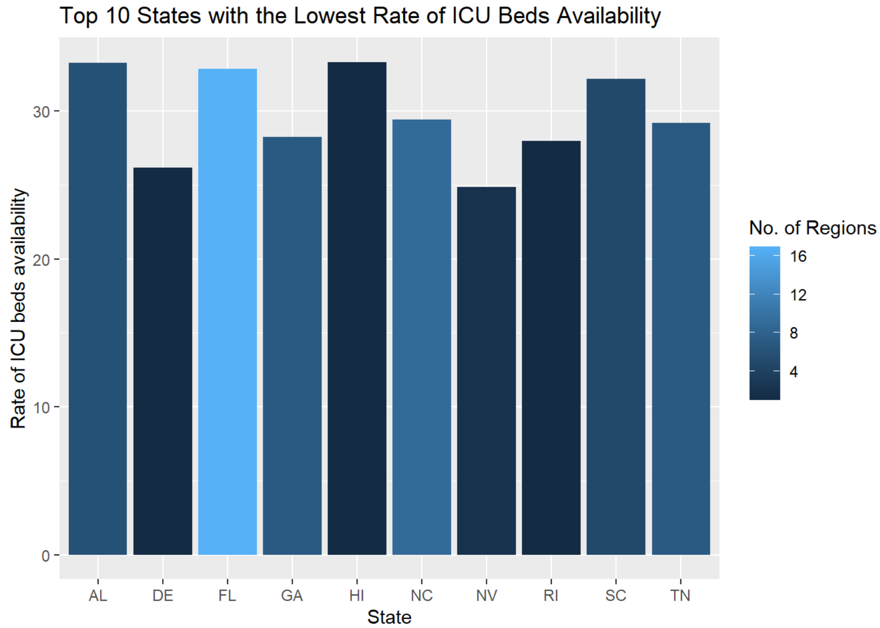
```
ggplot(icu_df, mapping=aes(x=`Total ICU Beds`, y=`ICU Beds Available`))+
  geom_point(mapping=aes(color = State))+
  ggtitle("Scatter plot of ICU Beds Occupied vs Available")
```



Step 3.2: Bargraph of the Top 10 States with the Lowest Rate of ICU Beds Availability

This plot just like the hospital plot version, compares the top 10 lowest ICU beds availability rates with respect to region. From observations, Nevada has the lowest availability rate while also having the least number of regions, Florida on the other hand has roughly 8% higher availability rate and significantly more regions. It is interesting to find that Georgia, Rhode Island, Nevada, Hawaii, North Carolina; all of which were featured in both hospital and ice bargraphs; this is an indication of hospital inefficiency in these regions as well higher demand.

```
icu_df[1:10,] %>%
  ggplot(mapping=aes(x=State, y=`Rate of ICU beds availability`, fill= `No. of Regions`))+
  geom_col(mapping=aes())+
  ggtitle("Top 10 States with the Lowest Rate of ICU Beds Availability")
```

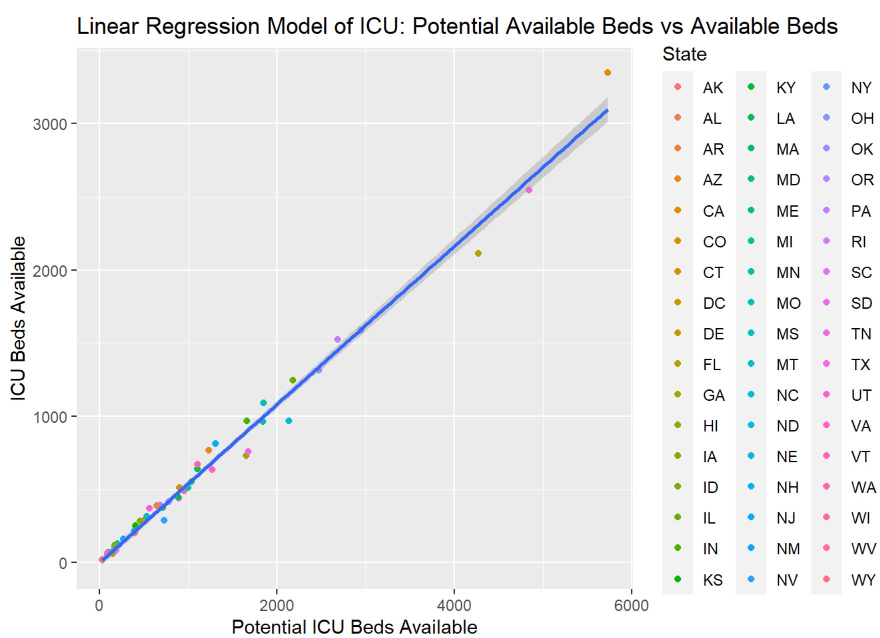


Step 3.3: Linear Regression Model of ICU: Potential Available Beds vs Available Beds

As discussed earlier in the hospital plot, the potential available beds attribute introduces an efficient method to increase hospital beds occupancy for covid patients. Interestingly enough, the linear regression is similar to the one we observed earlier, this means that most hospitals can promise 50% more ICU beds for most states in the concentraion. However, unlike hospital beds, ICU beds can be optimized to offer 50% more even for hospitals with 4000 or more beds. This is a nod to the 1:2 ratio nature of the relation.

```
ggplot(icu_df, mapping=aes(x=`Potential ICU Beds Available`, y=`ICU Beds Available`))+
  geom_point(mapping=aes(color = State))+
  geom_smooth(method=lm)+
  ggtitle("Linear Regression Model of ICU: Potential Available Beds vs Available Beds")
```

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



##

==

Part 4: Data Analysis with a Focus on State Population and Projected Infection Rates

The purpose of this section is to take the entire dataset and use it to create a smaller dataset that focuses on the population.

Step 4.1:

Obtain this smaller dataset that includes Town, State, Adult Population, Population 65+, Projected Infected Individual, Projected Hospitalized Individuals, and Projected Individuals Needing ICU Care.

```
pop_df <- df %>% select(Town, State, `Adult Population`, `Population 65+`,
`Projected Infected Individuals`, `Projected Hospitalized Individuals`,
`Projected Individuals Needing ICU Care`)
```

Step 4.2:

Turn the region data into statewide data by grouping by state and using summarize to add the totals for each state. The statewide data allows the opportunity to compare the risks that states are facing based on total population. The dataset contains all of the large regional hospitals, so we felt that the transition to statewide data would be seamless.

```
states_pop_df <- pop_df %>%
  group_by(State) %>%
  summarize(Adult_Population = sum(`Adult Population`),
  `Population 65+` = sum(`Population 65+`),
  Projected_Infected_Individuals = sum(`Projected Infected Individuals`),
  Projected_Hospitalized_Individuals = sum(`Projected Hospitalized Individuals`),
  Projected_ICU_Care = sum(`Projected Individuals Needing ICU Care`))
```

Step 4.3:

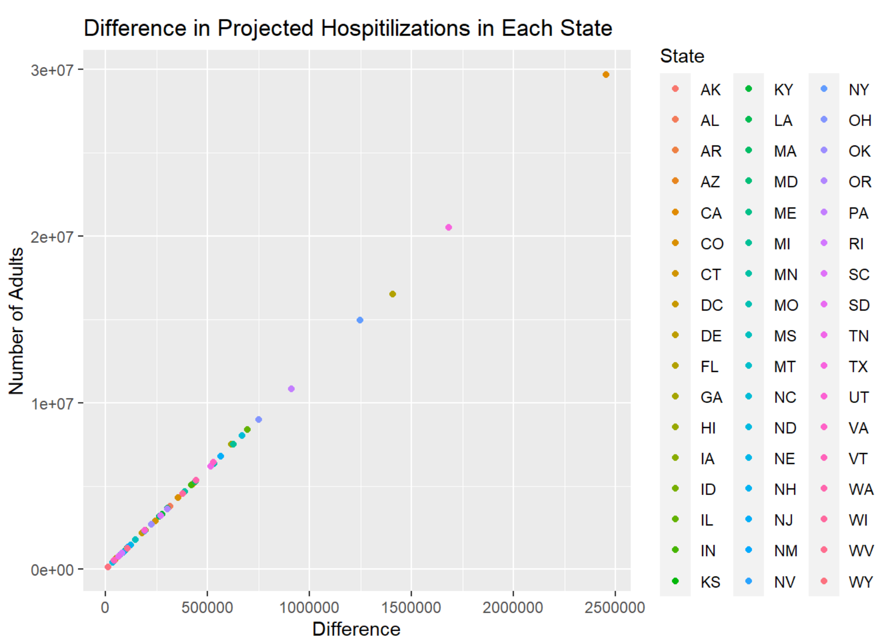
Once statewide data exists, we next wanted to show how much the risk would increase if 60 percent of the adult population contracted the virus in each state. In order to provide a snapshot, we tripled the number of Projected Infected Individuals, Projected Hospitalized Individuals, and Projected Individuals Needing ICU Care. This represents a 200 percent increase in each category over the original twenty percent. This is a plausible estimate according to various projection models.

```
states_pop_df <- states_pop_df %>%
  mutate(DoomsDay_Projected_Infected_Individuals = Projected_Infected_Individuals * 3,
  Doomsday_Projected_Hospitalized_Individuals = Projected_Hospitalized_Individuals *3,
  Doomsday_Projected_ICU_Care = Projected_ICU_Care * 3)
```

Step 4.4:

After doing this, we created a scatterplot showing the difference in the number of hospitalizations with the exact same adult population to emphasize how much worse this plausible scenario could make the situation. This scatterplot highlighted the risks that higher population states face if they aren't properly equipped.

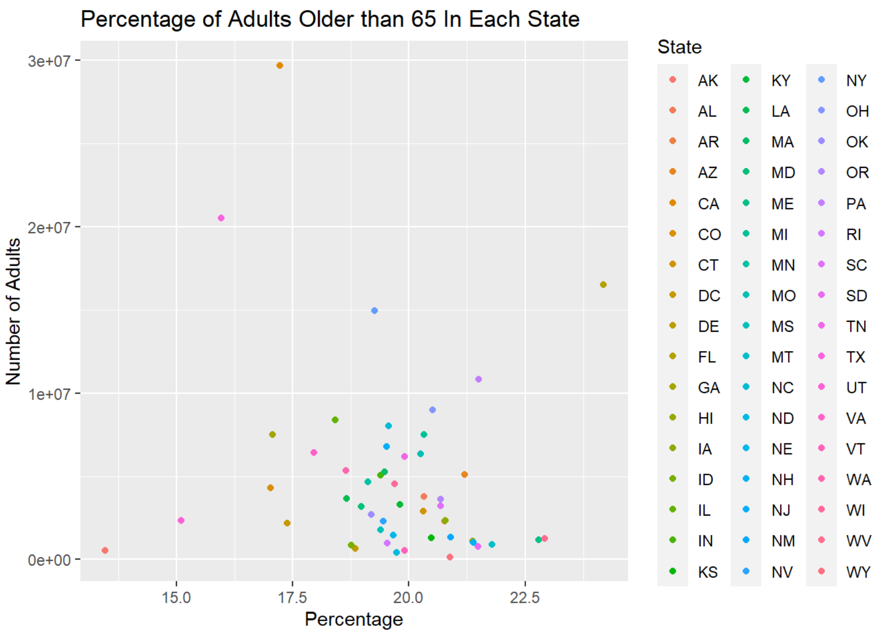
```
states_pop_df %>%
  mutate(difference_in_hospitilzations =
  Doomsday_Projected_Hospitalized_Individuals - Projected_Hospitalized_Individuals) %>%
  ggplot(mapping=aes(x = difference_in_hospitilzations, y= Adult_Population, color= State))+
  geom_point()+
  labs(title = "Difference in Projected Hospitilizations in Each State",
  x="Difference", y="Number of Adults")
```



Step 4.5:

When focusing solely on population, one of the most important things to consider is the proportion of people who are older than 65. It is well known that this demographic is at highest risk for this virus as they are much more likely to be hospitalized when they contract the virus. In order to depict this across the fifty states, we created a scatterplot with the adult population mapped to the y-axis and the percentage of adults older than 65 on the x axis for each state.

```
states_pop_df %>%
  mutate(Percentage_Adults_Older_65 = (`Population 65+` / Adult_Population) * 100) %>%
  ggplot(mapping=aes(x=Percentage_Adults_Older_65, y= Adult_Population, color = State))+
  geom_point()+
  labs(title = "Percentage of Adults Older than 65 In Each State", x="Percentage",
  y="Number of Adults")
```



Step 4.6: Furthermore, we used the proportion of adult population and the projected number of hospitalizations to calculate a risk level by state just according to the population. The purpose of this is to depict how a more frequent older population puts a state at risk. When you combine that with questionable hospital supplies, a state can be in major trouble.

```
states_pop_df <- states_pop_df %>%
  mutate(Risk_Level = (Projected_Hospitalized_Individuals + `Population 65+`) / Adult_Population) %>%
  arrange(desc(Risk_Level))

select(states_pop_df, State, Risk_Level)
```

```
## # A tibble: 51 x 2
##    State Risk_Level
##    <chr>      <dbl>
##  1 FL         0.284
##  2 WV         0.272
##  3 ME         0.270
##  4 MT         0.260
##  5 PA         0.257
##  6 SD         0.257
##  7 NH         0.256
##  8 HI         0.256
##  9 AZ         0.254
## 10 NM         0.251
## # ... with 41 more rows
```

==

Part 5: Merging the Two Data Frames To Calculate a Preparedness Score

Step 5.1:

We merged the beds data frame from parts 2 and 3 with the population data frame from part 4.

```
merge_df <- merge(beds_df, states_pop_df, by="State")
```

Step 5.2:

In order to quantify which states were the most prepared and which states were the least prepared, we used the merged dataset to calculate a preparedness score. In order to calculate the score, we derived a formula that uses the percentage of population 65 and older, the rate of hospital bed availability, and the rate of ICU bed availability. We felt that these factors differentiated the states the most and concluded that the rate of available hospital and ICU beds was more than twice as important as the percentage of elderly population in a state. This formula produced a score on a scale of 0 to 100 and is represented by the continuous attribute “preparedness_score”. The states who were the most prepared had scores in the 65-90 range, while the states who were the least prepared had scores in the 43-55 range.

```
merge_df <- merge_df %>%
  mutate(Preparedness_Score = ((`Population 65+`/Adult_Population)*.3 +
                          ((`Rate of Hospital beds availability`/100) +
                          (`Rate of ICU beds availability`/100)) *.70)*100)

merge_df
```

```
##    State No. of Regions Total Hospital Beds Hospital Beds Available
## 1     AK              1                1583                     533
## 2     AL              6               14793                    5282
## 3     AR              5                8560                    4063
## 4     AZ              4               12590                    4763
## 5     CA             24               68074                   22585
## 6     CO              7               10335                    4417
## 7     CT              3                7034                    1731
## 8     DC              1                5055                    1595
## 9     DE              1                1845                     601
## 10    FL             17               53744                   18464
## 11    GA              7               21861                    6681
## 12    HI              1                2623                     795
## 13    IA              8                9152                    3963
## 14    ID              2                2265                    1117
## 15    IL             13               25846                   10466
## 16    IN              9               17546                    7798
## 17    KS              2                5894                    2819
## 18    KY              5               11915                    5007
## 19    LA             10               14484                    6338
## 20    MA              3               13352                    3473
## 21    MD              3                8710                    2368
## 22    ME              2                3618                    1110
## 23    MI             15               24112                    8778
## 24    MN              6               14482                    5103
## 25    MO              6               22788                    9597
## 26    MS              6               10166                    4362
## 27    MT              3                4371                    1753
## 28    NC              9               22158                    6610
## 29    ND              3                2838                    1180
## 30    NE              2                6064                    2758
## 31    NH              2                2542                     915
## 32    NJ              7               19999                    6391
## 33    NM              1                2908                    1102
## 34    NV              2                6051                    1748
## 35    NY             10               50102                   11003
## 36    OH             10               27969                   11224
## 37    OK              3               10459                    4576
## 38    OR              5                7649                    2858
## 39    PA             14               34689                   12540
## 40    RI              1                2249                     690
## 41    SC              5               10313                    3587
## 42    SD              2                4943                    1996
## 43    TN              7               21489                    8314
## 44    TX             22               55629                   21879
## 45    UT              3                5690                    2562
## 46    VA              8               16782                    5208
## 47    VT              1                1743                     587
## 48    WA              6               11618                    4081
## 49    WI              8               11350                    4746
## 50    WV              3                5736                    2048
## 51    WY              1                 633                     378
##    Potential Hospital Beds Available Rate of Hospital beds availability
## 1                               1058                              33.67
## 2                              10037                              35.71
## 3                               6311                              47.46
## 4                               8676                              37.83
## 5                              45328                              33.18
## 6                               7376                              42.74
## 7                               4382                              24.61
## 8                               3325                              31.55
## 9                               1223                              32.57
## 10                             36105                              34.36
## 11                             14270                              30.56
## 12                              1709                              30.31
## 13                              6557                              43.30
## 14                              1691                              49.32
## 15                             18155                              40.49
## 16                             12672                              44.44
## 17                              4356                              47.83
## 18                              8460                              42.02
## 19                             10411                              43.76
## 20                              8412                              26.01
## 21                              5539                              27.19
## 22                              2364                              30.68
## 23                             16446                              36.41
## 24                              9794                              35.24
## 25                             16194                              42.11
## 26                              7263                              42.91
## 27                              3062                              40.11
## 28                             14384                              29.83
```

```
## 29                           2009                           41.58
## 30                           4411                           45.48
## 31                           1728                           36.00
## 32                          13194                           31.96
## 33                           2005                           37.90
## 34                           3899                           28.89
## 35                          30554                           21.96
## 36                          19595                           40.13
## 37                           7517                           43.75
## 38                           5254                           37.36
## 39                          23613                           36.15
## 40                           1470                           30.68
## 41                           6951                           34.78
## 42                           3469                           40.38
## 43                          14901                           38.69
## 44                          38756                           39.33
## 45                           4127                           45.03
## 46                          10994                           31.03
## 47                           1165                           33.68
## 48                           7850                           35.13
## 49                           8049                           41.81
## 50                           3892                           35.70
## 51                            505                           59.72
##    Total ICU Beds ICU Beds Available Potential ICU Beds Available
## 1             130                 55                           93
## 2            1903                633                         1268
## 3             908                390                          649
## 4            1702                766                         1235
## 5            8105               3350                         5729
## 6            1295                512                          904
## 7             741                298                          520
## 8             601                218                          409
## 9             237                 62                          149
## 10           6433               2113                         4269
## 11           2582                730                         1656
## 12            219                 73                          146
## 13            634                284                          460
## 14            230                119                          175
## 15           3106               1248                         2177
## 16           2351                969                         1660
## 17            554                251                          402
## 18           1331                447                          888
## 19           1572                640                         1105
## 20           1521                554                         1038
## 21           1053                377                          716
## 22            309                120                          215
## 23           2724                965                         1845
## 24           1477                513                          996
## 25           2604               1091                         1849
## 26            761                314                          537
## 27            277                131                          204
## 28           3294                970                         2133
## 29            130                 70                          100
## 30            565                217                          392
## 31            242                109                          175
## 32           1806                813                         1311
## 33            380                161                          270
## 34           1167                290                          729
## 35           4308               1586                         2947
## 36           3635               1314                         2474
## 37           1144                416                          781
## 38            973                394                          683
## 39           3846               1524                         2685
## 40            293                 82                          188
## 41           1358                437                          897
## 42            149                 72                          110
## 43           2601                760                         1679
## 44           7128               2544                         4837
## 45            758                371                          565
## 46           1902                634                         1268
## 47            117                 65                           91
## 48           1428                489                          958
## 49           1533                672                         1101
## 50            601                201                          400
## 51             32                 20                           26
##    Rate of ICU beds availability Adult_Population Population 65+
## 1                          42.31           551912          74327
## 2                          33.26          3795955         771541
## 3                          42.95          2316299         481100
## 4                          45.01          5105331        1082020
## 5                          41.33         29707362        5116290
```

```
## 6                          39.54         4312521         734223
## 7                          40.22         2912542         591758
## 8                          36.27         2179021         378878
## 9                          26.16          637120         120121
## 10                         32.85        16516645        3994067
## 11                         28.27         7498013        1280279
## 12                         33.33         1114023         238126
## 13                         44.79         2353064         489118
## 14                         51.74          871022         163390
## 15                         40.18         8399520        1547061
## 16                         41.22         5069753         982974
## 17                         45.31         1304408         267248
## 18                         33.58         3307888         655453
## 19                         40.71         3668745         684571
## 20                         36.42         5274900        1027216
## 21                         35.80         3176155         602855
## 22                         38.83         1194785         272413
## 23                         35.43         7490151        1522193
## 24                         34.73         4656898         890686
## 25                         41.90         6349481        1286055
## 26                         41.26         1766634         342486
## 27                         47.29          906533         197573
## 28                         29.45         8043918        1574295
## 29                         53.85          425099          83907
## 30                         38.41         1470000         289043
## 31                         45.04         1035178         221386
## 32                         45.02         6781169        1324017
## 33                         42.37         1336795         279457
## 34                         24.85         2310947         449422
## 35                         36.82        14955035        2881848
## 36                         36.15         8966961        1839711
## 37                         36.36         2714942         520901
## 38                         40.49         3615352         747753
## 39                         39.63        10833472        2329774
## 40                         27.99          986702         192737
## 41                         32.18         3222544         666653
## 42                         48.32          777153         166949
## 43                         29.22         6191882        1233128
## 44                         35.69        20506130        3273529
## 45                         48.94         2345662         354229
## 46                         33.33         6404664        1150633
## 47                         55.56          522720         104104
## 48                         34.24         5357559         998729
## 49                         43.84         4541350         894434
## 50                         33.44         1255837         287853
## 51                         62.50          152884          31930
##    Projected_Infected_Individuals Projected_Hospitalized_Individuals
## 1                          110382                              22385
## 2                          759191                             158906
## 3                          463259                              97160
## 4                         1021067                             214562
## 5                         5941474                            1226088
## 6                          862504                             177825
## 7                          582508                             121919
## 8                          435804                              90001
## 9                          127424                              26493
## 10                        3303330                             703520
## 11                        1499603                             309250
## 12                         222805                              46857
## 13                         470612                              98709
## 14                         174204                              36203
## 15                        1679904                             348576
## 16                        1013952                             211328
## 17                         260882                              54645
## 18                         661577                             138155
## 19                         733748                             152418
## 20                        1054980                             219963
## 21                         635232                             132148
## 22                         238957                              50577
## 23                        1498030                             313547
## 24                         931379                             193885
## 25                        1269896                             265716
## 26                         353327                              73639
## 27                         181306                              38202
## 28                        1608784                             335580
## 29                          85020                              17747
## 30                         294000                              61352
## 31                         207036                              43543
## 32                        1356233                             282839
## 33                         267359                              56108
## 34                         462190                              96356
```

```
## 35                    2991008                                   623048
## 36                    1793391                                   375698
## 37                     542988                                   113064
## 38                     723070                                   151591
## 39                    2166694                                   455937
## 40                     197340                                    41157
## 41                     644508                                   135122
## 42                     155431                                    32704
## 43                    1238376                                   258720
## 44                    4101225                                   841429
## 45                     469132                                    95865
## 46                    1280932                                   265238
## 47                     104544                                    21841
## 48                    1071511                                   222563
## 49                     908270                                   189566
## 50                     251168                                    53190
## 51                      30577                                     6416
##    Projected_ICU_Care DoomsDay_Projected_Infected_Individuals
## 1                4656                                  331146
## 2               34370                                 2277573
## 3               21065                                 1389777
## 4               46625                                 3063201
## 5              260664                                17824422
## 6               37762                                 2587512
## 7               26369                                 1747524
## 8               19152                                 1307412
## 9                5684                                  382272
## 10             155296                                 9909990
## 11              65691                                 4498809
## 12              10192                                  668415
## 13              21403                                 1411836
## 14               7763                                  522612
## 15              74607                                 5039712
## 16              45475                                 3041856
## 17              11829                                  782646
## 18              29799                                 1984731
## 19              32667                                 2201244
## 20              47355                                 3164940
## 21              28373                                 1905696
## 22              11085                                  716871
## 23              67816                                 4494090
## 24              41660                                 2794137
## 25              57450                                 3809688
## 26              15847                                 1059981
## 27               8329                                  543918
## 28              72285                                 4826352
## 29               3826                                  255060
## 30              13222                                  882000
## 31               9472                                  621108
## 32              60909                                 4068699
## 33              12174                                  802077
## 34              20741                                 1386570
## 35             133983                                 8973024
## 36              81345                                 5380173
## 37              24303                                 1628964
## 38              32852                                 2169210
## 39              99245                                 6500082
## 40               8863                                  592020
## 41              29284                                 1933524
## 42               7118                                  466293
## 43              55834                                 3715128
## 44             177598                                12303675
## 45              20133                                 1407396
## 46              56627                                 3842796
## 47               4714                                  313632
## 48              47694                                 3214533
## 49              40859                                 2724810
## 50              11665                                  753504
## 51               1392                                   91731
##    Doomsday_Projected_Hospitalized_Individuals Doomsday_Projected_ICU_Care
## 1                                        67155                       13968
## 2                                       476718                      103110
## 3                                       291480                       63195
## 4                                       643686                      139875
## 5                                      3678264                      781992
## 6                                       533475                      113286
## 7                                       365757                       79107
## 8                                       270003                       57456
## 9                                        79479                       17052
## 10                                     2110560                      465888
## 11                                      927750                      197073
```

```
## 12                                        140571                      30576
## 13                                        296127                      64209
## 14                                        108609                      23289
## 15                                       1045728                     223821
## 16                                        633984                     136425
## 17                                        163935                      35487
## 18                                        414465                      89397
## 19                                        457254                      98001
## 20                                        659889                     142065
## 21                                        396444                      85119
## 22                                        151731                      33255
## 23                                        940641                     203448
## 24                                        581655                     124980
## 25                                        797148                     172350
## 26                                        220917                      47541
## 27                                        114606                      24987
## 28                                       1006740                     216855
## 29                                         53241                      11478
## 30                                        184056                      39666
## 31                                        130629                      28416
## 32                                        848517                     182727
## 33                                        168324                      36522
## 34                                        289068                      62223
## 35                                       1869144                     401949
## 36                                       1127094                     244035
## 37                                        339192                      72909
## 38                                        454773                      98556
## 39                                       1367811                     297735
## 40                                        123471                      26589
## 41                                        405366                      87852
## 42                                         98112                      21354
## 43                                        776160                     167502
## 44                                       2524287                     532794
## 45                                        287595                      60399
## 46                                        795714                     169881
## 47                                         65523                      14142
## 48                                        667689                     143082
## 49                                        568698                     122577
## 50                                        159570                      34995
## 51                                         19248                       4176
##    Risk_Level Preparedness_Score
## 1   0.1752308           57.22615
## 2   0.2451154           54.37660
## 3   0.2496483           69.51806
## 4   0.2539663           64.34618
## 5   0.2134952           57.32369
## 6   0.2114884           62.70361
## 7   0.2450358           51.47627
## 8   0.2151787           52.69026
## 9   0.2301199           46.76712
## 10  0.2844153           54.30162
## 11  0.2119934           46.30347
## 12  0.2558143           50.96060
## 13  0.2498134           67.89893
## 14  0.2291481           76.36953
## 15  0.2256840           61.99453
## 16  0.2355740           65.77870
## 17  0.2467732           71.34442
## 18  0.2399138           58.86445
## 19  0.2281404           64.72686
## 20  0.2364365           49.54310
## 21  0.2314128           49.78720
## 22  0.2703332           55.49705
## 23  0.2450872           56.38478
## 24  0.2328956           54.71685
## 25  0.2443934           64.88335
## 26  0.2355468           64.73491
## 27  0.2600843           67.71831
## 28  0.2374309           47.36737
## 29  0.2391302           72.72247
## 30  0.2383639           64.62184
## 31  0.2559260           63.14388
## 32  0.2369586           59.74347
## 33  0.2510220           62.46050
## 34  0.2361707           43.45226
## 35  0.2343623           46.92703
## 36  0.2470635           59.55096
## 37  0.2335096           61.83294
## 38  0.2487570           60.69981
## 39  0.2571393           59.49760
## 40  0.2370462           46.92904
```

```
## 41  0.2488019           53.07815
## 42  0.2569031           68.53464
## 43  0.2409361           53.51157
## 44  0.2006697           57.30310
## 45  0.1918836           70.30944
## 46  0.2210687           50.44166
## 47  0.2409416           68.44275
## 48  0.2279568           54.15145
## 49  0.2386955           65.86360
## 50  0.2715663           55.27436
## 51  0.2508176           91.81953
```

Step 5.3:

Once we calculated our preparedness_score, we created a separate data frame to prepare for Logistic Regression. This data frame includes all of the pertinent variables in our regression model, including the building blocks for the calculation of each preparedness_score.

```
fit_df <- select(merge_df, State, `Rate of Hospital beds availability`,
`Rate of ICU beds availability`, Adult_Population, `Population 65+`, Preparedness_Score)

fit_df <- fit_df %>%
  mutate(Percentage_65_older = (`Population 65+` / Adult_Population) * 30,
  Hospital_Bed_Availability = ((`Rate of Hospital beds availability`/100) +
  (`Rate of ICU beds availability`/100))*70)
```

Step 5.4:

We perform logistic regression using the fit_df created in step 3. The purpose of doing this is to quantify the relationship between preparedness_scores and the two independent variables. The preparedness score is the dependent variable whilethe Percentage_65_older and the Hospital_Bed_Availability are the independent variables. Using the broom library, the tidy function with our model produces a p value of 0.25, which tells us that our relationship is relatively strong, but not strong enough to be statistically significant.

```
lm_df <- glm(Preparedness_Score ~ Percentage_65_older + Hospital_Bed_Availability, data=fit_df)
broom::tidy(lm_df)
```

```
## # A tibble: 3 x 5
##   term                       estimate std.error statistic p.value
##   <chr>                         <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)               -2.39e-14  2.05e-14  -1.16e 0   0.250
## 2 Percentage_65_older        1.00e+ 0  3.08e-15   3.25e14   0
## 3 Hospital_Bed_Availability  1.00e+ 0  1.91e-16   5.22e15   0
```

```
lm_df%>%
  tidy()%>%
  knitr::kable(digits=4)
```

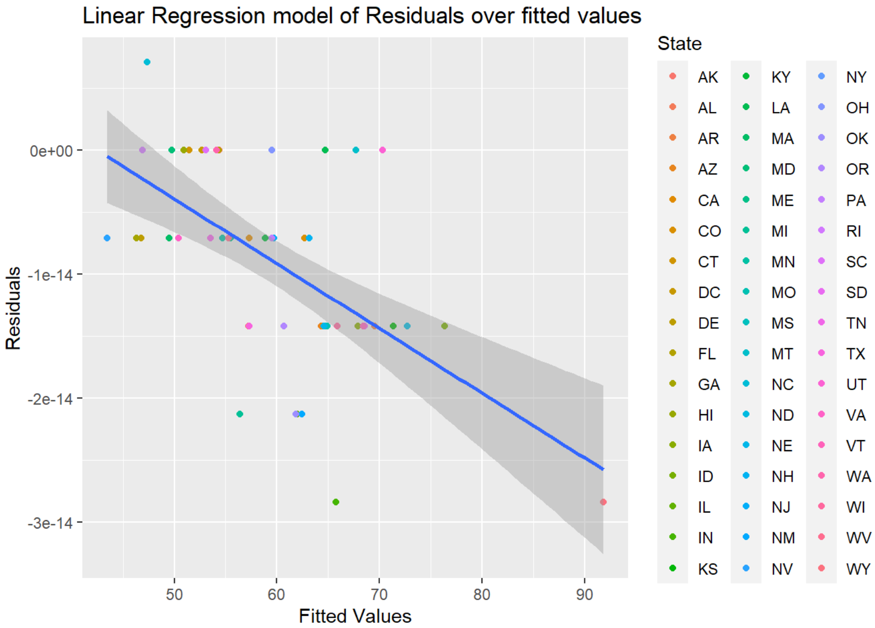
term	estimate	std.error	statistic	p.value
(Intercept)	0	0	-1.164000e+00	0.2502
Percentage_65_older	1	0	3.251182e+14	0.0000
Hospital_Bed_Availability	1	0	5.222330e+15	0.0000

Step 5.5:

We used the results of the logistic regression to graph the residuals versus the fitted values. Our residuals appear to skew a little below zero, but they are within a reasonable range of zero.

```
ggplot(lm_df, mapping=aes(x=lm_df$fitted.values, y=lm_df$residuals))+
    geom_point(mapping=aes(color = lm_df$data$State))+
    geom_smooth(method=lm)+
    labs(title="Linear Regression model of Residuals over fitted values",
        y = "Residuals",
         x = "Fitted Values",
        color = "State")
```

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




```
fit_df$fitted_values <- lm_df$fitted.values
```

===

Part 6: Visually Depicting the Tiers of Preparedness_Score

In order to assess each state's preparedness score with respect to one another, we decided to utilize the leaflet library. We started off by downloading a csv file with the latitudes for each state and then merged it with our data to create a new data frame (map_df). Using the preparedness score for each state, we color coded them in order to show visually which states could be at risk. As the preparedness score ranges roughly from 40-95, with 40 being considered least prepared, the color sequence follows that range as well with Red indicating a state is poorly equipped where as blue is an indication of strongly equipped state. An interesting feature of the map is the popup option, clicking a circle for any state will reveal details about it preparedness score.

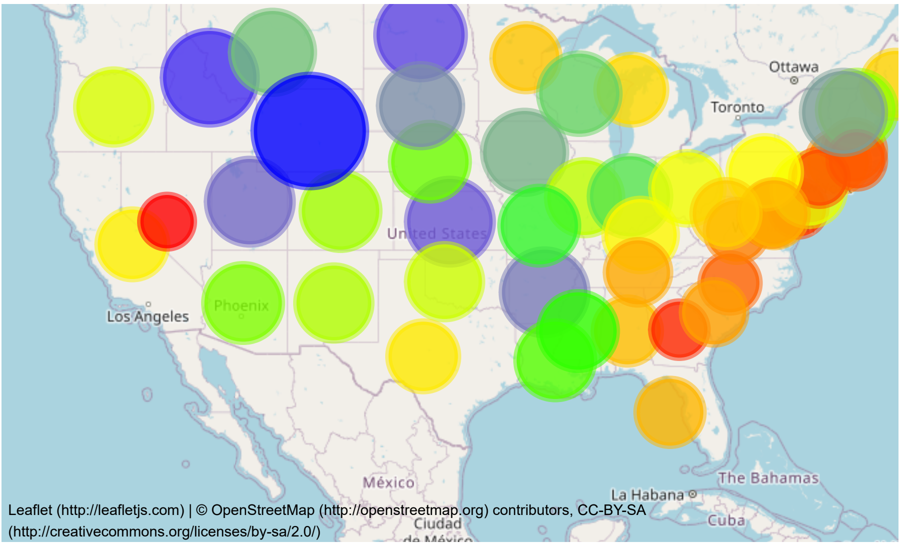
```
#https://www.kaggle.com/washimahmed/usa-latlong-for-state-abbreviations
csv_file <- "statelatlong.csv"
sll <- read_csv(csv_file)
```

```
## Parsed with column specification:
## cols(
##   State = col_character(),
##   Latitude = col_double(),
##   Longitude = col_double(),
##   City = col_character()
## )
```

```
map_df <- merge(sll, fit_df, by = "State")

map_df$Preparedness_Score = round(map_df$Preparedness_Score, digits = 2)
map_df$Preparedness_Score = as.character(map_df$Preparedness_Score)
map_df$Preparedness_Score = paste("Preparedness Score = ", map_df$Preparedness_Score)


pal <- colorFactor(c("red", "orange", "yellow", "green", "blue"), domain = c(map_df$fitted_values))

prep_map <- leaflet(map_df) %>%
  addTiles() %>%
  #Longitude and Latitude of Kansas was used as the default map co-ordinates.
  setView(lat=38.49, lng=-98.32, zoom=4) %>%
  addCircleMarkers(radius = ~map_df$fitted_values/2.2, popup = map_df$Preparedness_Score,
                  color = ~pal(map_df$fitted_values), fillOpacity = 0.8)
```

```
## Assuming "Longitude" and "Latitude" are longitude and latitude, respectively
```

```
prep_map
```



##
==

Part 7: References

Main dataset link = https://www.kaggle.com/mrmorj/hospital-bed-capacity-and-covid19 (https://www.kaggle.com/mrmorj/hospital-bed-capacity-and-covid19) Dataset link for longitudes and latitudes = https://www.kaggle.com/washimahmed/usa-latlong-for-state-abbreviations (https://www.kaggle.com/washimahmed/usa-latlong-for-state-abbreviations)