Examining the Relationship Between Behavior and Risk of Contracting COVID-19

Jake Campbell, Jin Lee, and Julie Grossman

4/5/2021

I. Introduction

In early 2020, the news began reporting on a new disease in China. In March of 2020, this disease, COVID-19, began spreading in the United States. Despite lockdowns and mask mandates, it morphed into a pandemic. Over a year later, the pandemic continues. One of the reasons the pandemic has continued, is that not everybody takes it seriously. We hypothesize that people who engage in behaviors traditionally associated with mitigating the risk of contracting COVID-19, such as handwashing, mask wearing, and avoiding social gatherings, are less likely to contract COVID.

This data, which examines people's behaviors in response to the COVID-19 pandemic, comes from Imperial College London. The data also includes information about COVID-19 test results. Data collection started in March 2020 and is ongoing, but the subset of the data being used for this project was collected between March 2020 and July 2020. The dataset description does not detail the sampling method, but it states that responses are nationally representative of the general public and that, "people with severe symptoms, people who have been hospitalized, and some other hard to reach group will be under-represented in the sample." No other sampling bias is suspected.

This data was collected through interviews and represents an observational study, as there is no direct intervention by the researchers. No bias is suspected in the questions as they were created by a well-respected institution and measure primarily objective factors, such as frequencies as counts. The measurements may be biased towards behaviors that are seen as socially acceptable as people may not feel comfortable admitting to a researcher that they don't do things like wear a mask or avoid large gatherings.

This data is very relevant as it is about a pandemic that is ongoing. It has been shown that individual behavior has a significant impact on the spread of COVID-19, so examining COVID-related behavior across the United States could potentially be used to identify areas that are likely to have outbreaks. It's also interesting to observe how COVID-related behavior varies across the United States.

The data was cleaned by first removing columns that were not of interest. The remaining columns were renamed to make them easier to work with. All of the data was read in as character data, so columns that contained factor or numeric data were transformed into their respective data type. The data cleaning code can be seen below.

```
#select columns of interest
data <- raw_data %>%
  select("endtime" : "i13_health", "gender" : "employment_status")

#display names before renaming
head(names(data), 10)
```

```
##  [1] "endtime"     "state"       "qweek"       "i1_health"   "i2_health"
##  [6] "i7a_health"  "i3_health"   "i4_health"   "i5_health_1" "i5_health_2"
```

```
#rename columns
new_names <- c("endtime",
               "state", "qweek",
               "hh_contact", "non_hh_contact",
               "n_leave_home", "tested",
               "hh_tested", "cough",
               "fever", "lost_smell",
               "lost_taste", "short_of_breath",
               "no_symp", "contact_symp_person",
               "isolate_after_symp", "doctor_after_symp",
               "travel_before_symp", "isolate_if_symp",
               "isolation_ease", "isolation_willingness",
               "mask", "hand_washing",
               "sanitizer", "cover_mouth",
               "avoid_symp_person", "avoid_going_out",
               "avoid_healthcare", "avoid_pub_transport",
               "avoid_work_not_home", "avoid_school_not_home",
               "avoid_guests", "avoid_s_gath",
               "avoid_m_gath", "avoid_l_gath",
               "avoid_crowds", "avoid_shopping",
               "stop_share_bedroom", "stop_share_meals",
               "surface_cleaning", "avoid_touching_public_objects",
               "n_wash_sanitize", "gender",
               "age", "hh_size",
               "hh_children", "emp_status"
               )

names(data) <- new_names

#display names after renaming
head(names(data), 10)
```

```
##  [1] "endtime"        "state"          "qweek"          "hh_contact"
##  [5] "non_hh_contact" "n_leave_home"   "tested"         "hh_tested"
##  [9] "cough"          "fever"
```

```
#display data summary before recoding
#note how all columns are character data
head(summary(data[13:20]))
```

```
##  short_of_breath      no_symp          contact_symp_person isolate_after_symp
##  Length:14031       Length:14031       Length:14031        Length:14031
##  Class :character   Class :character   Class :character    Class :character
##  Mode  :character   Mode  :character   Mode  :character    Mode  :character
##  doctor_after_symp  travel_before_symp isolate_if_symp    isolation_ease
##  Length:14031       Length:14031       Length:14031       Length:14031
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
```

```
#commonly used factor levels
yn_levels <- c("No", "Yes")
freq_levels <- c("Not at all", "Rarely", "Sometimes", "Frequently", "Always")

data_recode <- data %>%
  #convert from character to numeric
  mutate(hh_contact = as.numeric(hh_contact),
         non_hh_contact = as.numeric(non_hh_contact),
         n_leave_home = as.numeric(n_leave_home),
         age = as.numeric(age),
         hh_size = as.numeric(hh_size),
         hh_children = as.numeric(hh_children),
         n_wash_sanitize = as.numeric(n_wash_sanitize)
         ) %>%
  #convert from yes/no character to factor
  mutate(cough = factor(cough, levels = yn_levels),
         fever = factor(fever, levels = yn_levels),
         lost_smell = factor(lost_smell, levels = yn_levels),
         lost_taste = factor(lost_taste, levels = yn_levels),
         short_of_breath = factor(short_of_breath, levels = yn_levels),
         no_symp = factor(no_symp, levels = yn_levels),
         doctor_after_symp = factor(doctor_after_symp, levels = yn_levels),
         ) %>%
  #convert from frequency character to factor
  mutate(mask = factor(mask, levels = freq_levels),
         hand_washing = factor(hand_washing, levels = freq_levels),
         sanitizer = factor(sanitizer, levels = freq_levels),
         cover_mouth = factor(cover_mouth, levels = freq_levels),
         avoid_symp_person = factor(avoid_symp_person, levels = freq_levels),
         avoid_going_out = factor(avoid_going_out, levels = freq_levels),
         avoid_healthcare = factor(avoid_healthcare, levels = freq_levels),
         avoid_pub_transport = factor(avoid_pub_transport, levels = freq_levels),
         avoid_work_not_home = factor(avoid_work_not_home, levels = freq_levels),
         avoid_school_not_home = factor(avoid_school_not_home, levels = freq_levels),
         avoid_guests = factor(avoid_guests, levels = freq_levels),
         avoid_s_gath = factor(avoid_s_gath, levels = freq_levels),
         avoid_m_gath = factor(avoid_m_gath, levels = freq_levels),
         avoid_l_gath = factor(avoid_l_gath, levels = freq_levels),
         avoid_crowds = factor(avoid_crowds, levels = freq_levels),
         avoid_shopping = factor(avoid_shopping, levels = freq_levels),
         stop_share_bedroom = factor(stop_share_bedroom, levels = freq_levels),
         stop_share_meals = factor(stop_share_meals, levels = freq_levels),
         surface_cleaning = factor(surface_cleaning, levels = freq_levels),
         avoid_touching_public_objects = factor(avoid_touching_public_objects, levels = freq_lev
els),
         isolate_after_symp = factor(isolate_after_symp, levels = freq_levels)
         ) %>%
  #convert other characters to factor
  mutate(tested = factor(tested,
                         levels = c("No, I have not", "Yes, and I tested positive",
                                    "Yes, and I have not received my results from the te
st yet",                                              "Yes, and I tested negative"),
                         labels = c("N", "Y+", "Y~", "Y-")),
```

```
         hh_tested = factor(hh_tested,
                            levels = c("No, they have not", "Yes, and they tested positive",
                                                    "Yes, and they have not received their results
from the test yet", "Yes, and they tested negative"),
                            labels = c("N", "Y+", "Y~", "Y-")),
         gender = factor(gender, levels = c("Male", "Female")),
         contact_symp_person = factor(contact_symp_person,
                                      levels = c("Not sure", "No", "Yes")),
         travel_before_symp = factor(travel_before_symp,levels = c("Not sure", "No", "Yes")),
         isolate_if_symp = factor(isolate_if_symp, levels = c("Not sure", "No", "Yes")),
         isolation_ease = factor(isolation_ease,
                                 levels = c("Not sure", "Very difficult", "Somewhat difficult",
                                            "Neither easy nor difficult", "Somewhat easy",
                                            "Very easy")),
         isolation_willingness = factor(isolation_willingness,
                                        levels = c("Not sure", "Very unwilling",
                                                   "Somewhat unwilling",
                                                   "Neither willing nor unwilling",
                                                   "Somewhat willing", "Very willing")),
         emp_status = factor(emp_status,
                                    levels = c("Other", "Retired", "Not working", "Unemployed",
                                               "Full time student", "Part time employment",
                                               "Full time employment")),
         qweek = factor(qweek,
                        levels = c("week 1", "week 2", "week 3", "week 4", "week 5", "week 6",
"week 7", "week 8", "week 9", "week 10", "week 11", "week 12", "week 13"))



         )


#display data summary after recoding
head(summary(data_recode[13:20]))
```

```
##  short_of_breath no_symp      contact_symp_person  isolate_after_symp
##  No  :13137      No  : 1408   Not sure:  292      Not at all:  191
##  Yes :  472      Yes :12201   No      :  743      Rarely    :   85
##  NA's:  422      NA's:  422   Yes     :  375      Sometimes :  204
##                               NA's    :12621      Frequently:  412
##                                                   Always    :  518
##                                                   NA's      :12621
##  doctor_after_symp travel_before_symp isolate_if_symp
##  No  :  938        Not sure:  241     Not sure:1484
##  Yes :  400        No      :  852     No      :2734
##  NA's:12693        Yes     :  317     Yes     :9813
##                    NA's    :12621
##
##
##                     isolation_ease
##  Not sure                  : 736
##  Very difficult            : 804
##  Somewhat difficult        :1490
##  Neither easy nor difficult:1495
##  Somewhat easy             :3565
##  Very easy                 :5941
```

II. Data Visualizations

A. Test Results by Mask Wearing Frequency

A bar graph of positive and negative test results by mask usage demonstrates that the more often one wears a mask, the more likely they are to test negative for COVID-19. While there may be confounding variables due to other behaviors, type of mask, what one defines as “frequent”, and close contacts, it is demonstrated that the more often an individual wears a mask, the less likely they are to test positive for COVID-19.

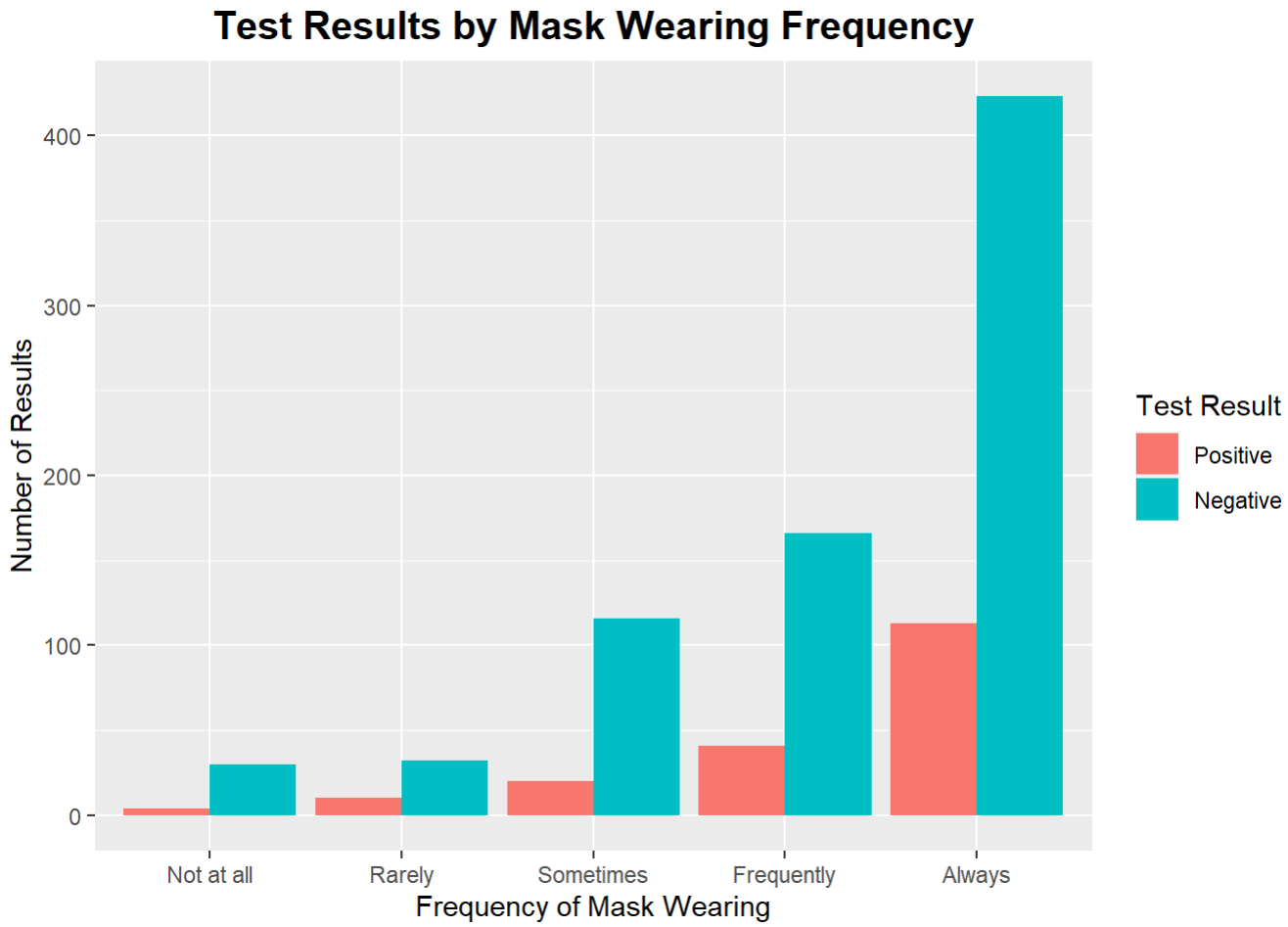
Of the 955 individuals, 3.56% never wore a mask, 4.4% rarely wore a mask, 14.24% sometimes wore a mask, 21.68% frequently wore a mask, and 56.13% always wore a mask. 188(19.69%) tested positive for COVID-19, while 767(80.31%) tested negative for COVID-19.

```
# data with applied filters
test_data <- data_recode %>% filter(tested == 'Y+' | tested == 'Y-')

# # Graph 1: positive tests v. mask wearing

# Tests Results v. Mask Usage
mask1 <- ggplot(data = test_data, mapping = aes(x = mask, fill = tested)) +
  geom_bar(position = 'dodge') +
  labs(title = "Test Results by Mask Wearing Frequency",
       x = "Frequency of Mask Wearing",
       y = "Number of Results",
       fill = "Test Result") +
  theme(plot.title = element_text(face = 'bold', size = 15, hjust = .5))+
  scale_fill_discrete(labels = (c("Positive", "Negative")))

mask1
```



B. Handwashing Density Plot

Density plots of hand washing and sanitizing habits compared to test results demonstrates that the more often one washes their hands, the more likely they are to test negative for COVID-19. While there may be confounding variables due to other behaviors, frequency of mask wearing, and close contacts, it is demonstrated that the more often an individual washes their hands, the less likely they are to test positive for COVID-19.

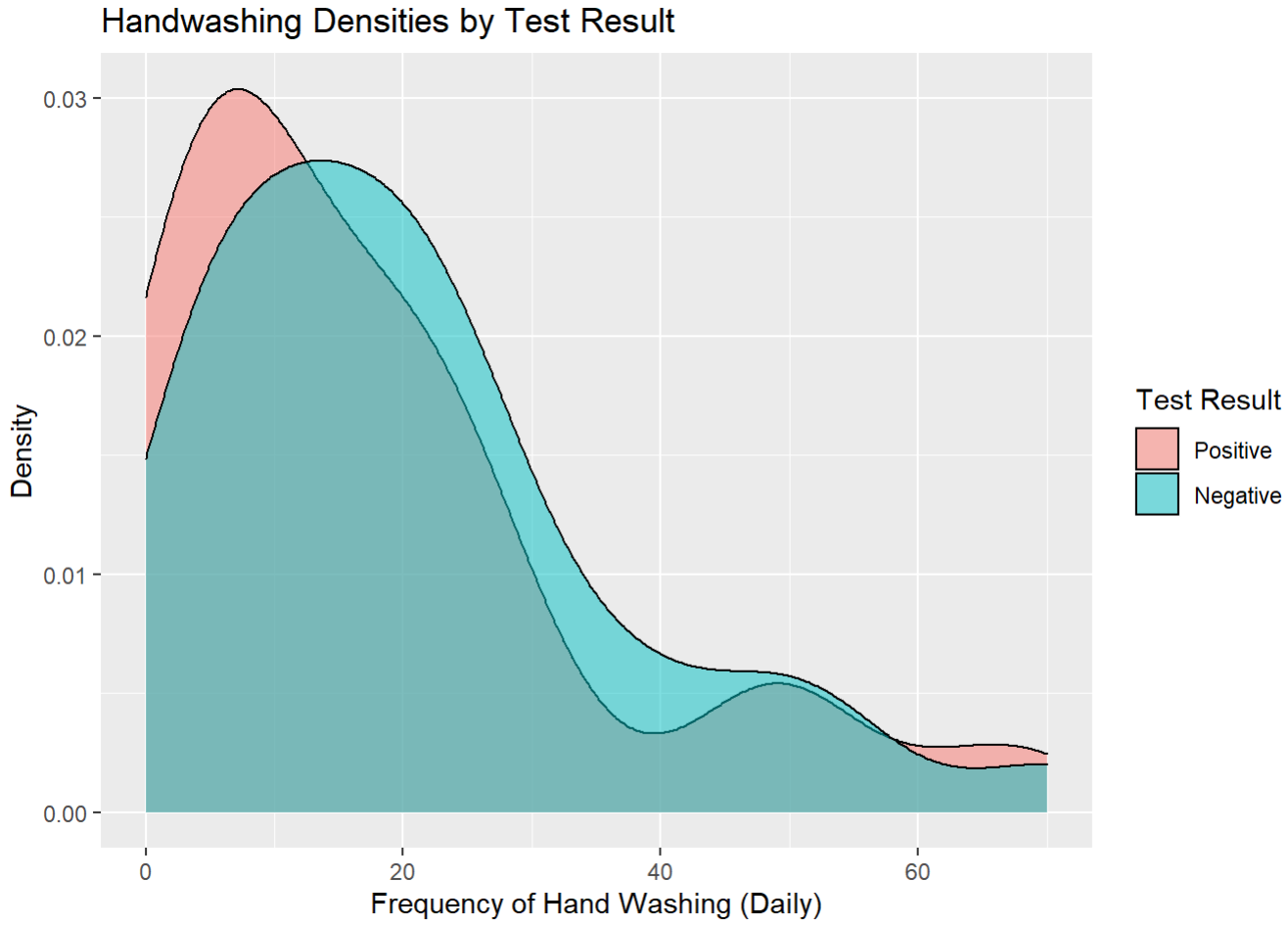
Of the sample size of 59, the median number of hand washing per day of those who tested positive for COVID-19 was 12, while it was 17.5 for those who tested negative.

```
# # Graph 2: positive tests v. number of times sanitizing per day

# Data modification
sanitizing_data <- test_data %>%
  filter(n_wash_sanitize < 75) %>%
  group_by(n_wash_sanitize, tested) %>%
  summarise(sampsize = n()) %>%
  ungroup()



# Test Results v. Number of Times of Hand Washing per Day
sanitizing1 <- ggplot(data = sanitizing_data,
                      mapping = aes(x = n_wash_sanitize, fill = tested)) +
  geom_density(alpha = .5, color = 'black') +
  labs(title = "Handwashing Densities by Test Result",
       x = "Frequency of Hand Washing (Daily)",
       y = "Density") +
  #theme(axis.text.y = element_blank(), axis.ticks.y = element_blank(),
        #plot.title = element_text(face = 'bold', size = 15, hjust = .5)) +
  guides(fill = guide_legend(title = "Test Result"))+
  scale_fill_discrete(labels = (c("Positive", "Negative")))
sanitizing1
```



C. Average Number of Handwashings by State

In order to visualize the frequency of handwashing in the United States, the data has been translated onto the United States map. The selected color of each state indicates the level of handwashing frequency. As the color of the state comes to be more of a blue color, this showcases a higher frequency of handwashing, which signifies a more overall handwashing performance as a state. As the color of the state comes to be more of a red color, this showcases a lower frequency of handwashing, which signifies a lesser overall handwashing performance as a state. The spectrum of color radiance from blue to red can assist in indicating the frequency of handwashing and the comparison from state to state.

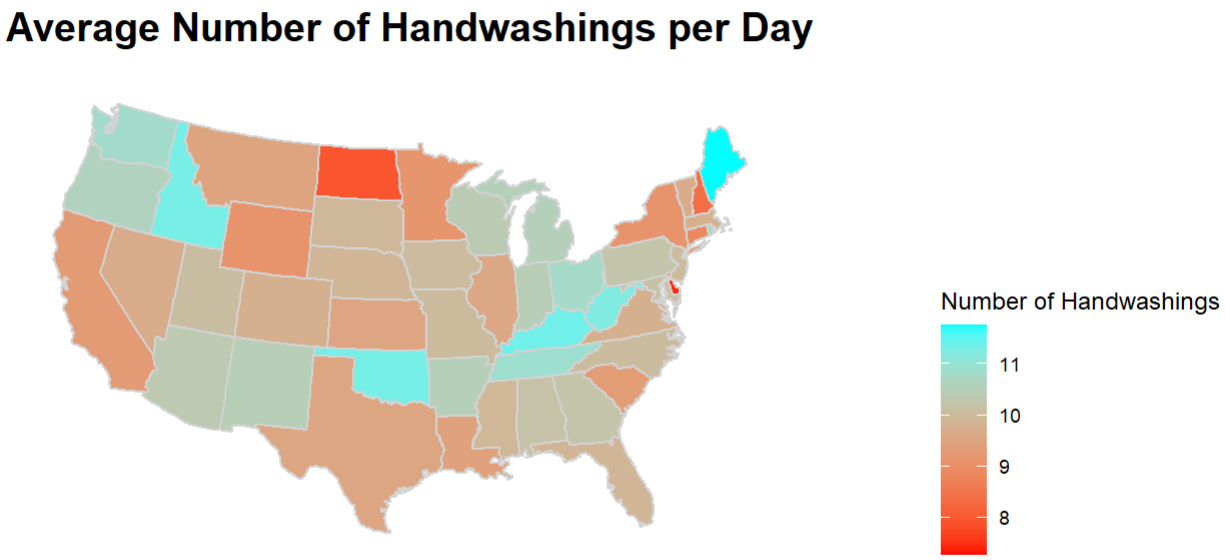
```
# merge data_recode with map_data
us_states <- map_data("state")
data_recode$region <- tolower(data_recode$state)

#summarize data by state
us_sanitize <- data_recode %>%
  filter(n_wash_sanitize < 100) %>%
  group_by(region) %>%
  summarize(mean_handwash = mean(n_wash_sanitize))
tibble(us_sanitize)
```

```
## # A tibble: 51 x 2
##    region               mean_handwash
##    <chr>                        <dbl>
##  1 alabama                      10.2
##  2 alaska                        8.47
##  3 arizona                      10.4
##  4 arkansas                     10.5
##  5 california                    9.30
##  6 colorado                      9.78
##  7 connecticut                   8.88
##  8 delaware                      7.44
##  9 district of columbia          7.28
## 10 florida                       9.91
## # ... with 41 more rows
```

```
us_states_recode <- us_states %>%
   left_join(us_sanitize, by = 'region')
```

```
# draw a map about Frequecy of handwashing by state
ggplot(data = us_states_recode,
   mapping = aes(x = long, y = lat,  group = group, fill = mean_handwash))  +
  geom_polygon(color = "lightgrey") +
  scale_fill_continuous(low = 'red', high = 'cyan') +
 coord_map(projection = "albers", lat0 = 39, lat1 = 45) +
  labs(title = "Average Number of Handwashings per Day") +
  labs(fill = "Number of Handwashings") +
  theme_map()+
   theme(plot.title = element_text(face = 'bold', size = 15, hjust = .5),
        legend.position = "right")
```



D. Average Mask Wearing Frequency by State

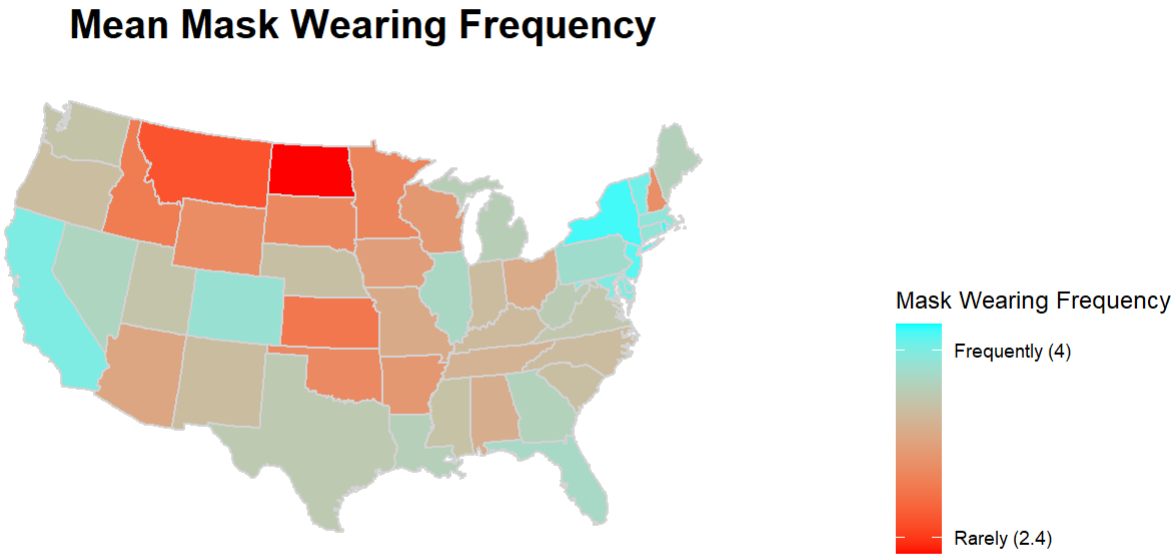
In order to visualize the medium of wearing masks in the United States, the data has been translated onto the United States map. The selected color of each state indicates the level of mask wearing medium. As the color of the state comes to be more of a blue color, this showcases a higher medium of wearing masks, which signifies a more overall mask-wearing performance as a state. For example, it is evident that New York (4.171958 medium), New Jersey (4.120181 medium) and Vermont (4.062500 medium) portrays a bluer color, signifying a higher medium of wearing masks. As the color of the state comes to be more of a red color, this showcases a lower medium of wearing mask, which signifies a lesser overall mask-wearing performance as a state. For example, it is evident that North Dakota (2.263158 medium), Montana (2.547170 medium) and Kansas (2.816327 medium) portrays a redder color, signifying a lower medium of wearing masks. The spectrum of color radiance from blue to red can assist in indicating the medium of wearing masks and the comparison from state to state.

```
# calculate state mask wearing averages
stats1 <- data_recode %>%
  group_by(state) %>%
  summarise( mmask = mean(as.numeric(mask), na.rm = TRUE))

#merge stats1 data with us_states
stats1$region <- tolower(stats1$state)

us_mask <- us_states %>%
   left_join(stats1, by = 'region')
```

```
# draw a map of mean mask wearing by state
ggplot(data = us_mask,
   mapping = aes(x = long, y = lat,  group = group, fill = mmask))  +
  geom_polygon(color = "lightgrey") +
  scale_fill_continuous(low = 'red', high = 'cyan',
                        breaks = c(2.4, 4),
                        labels = c("Rarely (2.4)", "Frequently (4)"))+
 coord_map(projection = "albers", lat0 = 39, lat1 = 45) +
  labs(title = "Mean Mask Wearing Frequency") +
  labs(fill = "Mask Wearing Frequency") +
  theme_map()+
  theme(plot.title = element_text(face = 'bold', size = 15, hjust = .5),
        legend.position = "right")
```



III. Decision Tree to Predict Positive Cases

A. Data Prep

Since most of the observations in the dataset don't have a positive or negative test result associated with them, the model was developed using the subset that did have associated test results. These observations were selected and features that were not associated with behavior were removed. The data was then split into a training set (with 800 observations) and a testing set (with 155 observations). The random seed was set for the data splitting as the results varied greatly when it was not.

```
test_results <- data_recode %>%
  #select only observations that have a positive or negative test result
  filter(tested %in% c("Y-", "Y+")) %>%
  #select columns to be used in training of model
  select(-endtime, -state, -qweek, -hh_tested, -cough, -fever, -short_of_breath, -lost_smell, -l
ost_taste,
         -short_of_breath, -no_symp, -contact_symp_person, -isolate_after_symp, -doctor_after_sy
mp,
         -travel_before_symp, -hh_size, -hh_children, -avoid_work_not_home, -avoid_school_not_ho
me, -age,
         -gender, -emp_status) %>%
  #refactors tested column so that only levels are positive and negative
  mutate(tested = droplevels(tested))

#check to ensure that there are no NA observations
sum(is.na(test_results))
```

```
## [1] 0
```

```
#generate random numbers for creating train and test set
set.seed(1)
test_sample <- sample(1:955, 800)

#create training data
test_results_train <- test_results[test_sample,]
test_results_train <- test_results_train %>%
  select(-region)
test_results_train_labels <- as.data.frame(test_results[test_sample,])


#create testing data
test_results_test <- test_results[-test_sample,]
test_results_test <- test_results_test %>%
  select(-region)
test_results_test_labels <- as.data.frame(test_results[-test_sample,])
```

B. Model Training

A decsion tree was used to predict which individuals would be likely to become infected with COVID-19. Decision trees work by splitting the data to minimize the entropy of each group and stopping the splitting once the splits no longer reduce entropy.

People who were predicted to test positive displayed behaviors like not washing their hands and being in a situation where it is difficult to isolate, which seem to make sense, However, both people who reported their frequency of avoiding medium gatherings as always or not at all were predicted to test positive. This is most likely due to the fact that the training dataset is relatively small compared to the full dataset. The accuracy of the model with the training data was 86%.

```
#create the model
test_results_model <- C5.0(tested ~ ., data = test_results_train)
#show information about the model
test_results_model
```

```
##
## Call:
## C5.0.formula(formula = tested ~ ., data = test_results_train)
##
## Classification Tree
## Number of samples: 800
## Number of predictors: 25
##
## Tree size: 14
##
## Non-standard options: attempt to group attributes
```

```
summary(test_results_model)
```

```
##
## Call:
## C5.0.formula(formula = tested ~ ., data = test_results_train)
##
##
## C5.0 [Release 2.07 GPL Edition]      Mon May 03 22:38:39 2021
## -------------------------------
##
## Class specified by attribute `outcome'
##
## Read 800 cases (26 attributes) from undefined.data
##
## Decision tree:
##
## n_wash_sanitize <= 1:
## :...avoid_guests in {Rarely,Always}: Y+ (16/2)
## :   avoid_guests in {Not at all,Sometimes,Frequently}:
## :   :...avoid_healthcare in {Not at all,Always}: Y+ (9/3)
## :       avoid_healthcare in {Rarely,Sometimes,Frequently}: Y- (23/2)
## n_wash_sanitize > 1:
## :...n_leave_home <= 2: Y- (527/57)
##     n_leave_home > 2:
##     :...stop_share_meals in {Not at all,Rarely,Sometimes}: Y- (82/13)
##         stop_share_meals in {Frequently,Always}:
##         :...avoid_symp_person in {Not at all,Sometimes}: Y- (24/3)
##             avoid_symp_person in {Rarely,Frequently,Always}:
##             :...avoid_going_out in {Not at all,Rarely}: Y- (6)
##                 avoid_going_out in {Sometimes,Frequently,Always}:
##                 :...isolate_if_symp in {Not sure,No}: Y- (31/9)
##                     isolate_if_symp = Yes: [S1]
##
## SubTree [S1]
##
## isolation_ease in {Not sure,Very difficult,Somewhat difficult}: Y+ (3/1)
## isolation_ease in {Neither easy nor difficult,Somewhat easy}: Y- (25/10)
## isolation_ease = Very easy:
## :...avoid_crowds in {Not at all,Rarely}: Y- (2)
##     avoid_crowds in {Sometimes,Frequently}: Y+ (20/5)
##     avoid_crowds = Always:
##     :...avoid_m_gath in {Not at all,Rarely,Sometimes,Always}: Y+ (27/6)
##         avoid_m_gath = Frequently: Y- (5/1)
##
##
## Evaluation on training data (800 cases):
##
##      Decision Tree
##    ----------------
##    Size      Errors
##
##      14  112(14.0%)   <<
##
##
##     (a)   (b)    <-classified as
```

```
##    ----  ----
##      58    95    (a): class Y+
##      17   630    (b): class Y-
##
##
##  Attribute usage:
##
##  100.00% n_wash_sanitize
##   94.00% n_leave_home
##   28.13% stop_share_meals
##   17.88% avoid_symp_person
##   14.88% avoid_going_out
##   14.13% isolate_if_symp
##   10.25% isolation_ease
##    6.75% avoid_crowds
##    6.00% avoid_guests
##    4.00% avoid_healthcare
##    4.00% avoid_m_gath
##
##
## Time: 0.0 secs
```

C. Model Testing and Assessment

The accuracy of the model on the testing data was approximately 81%. However, this is not an accurate representation of the effectiveness of the model. The model is very effective at predicting negative cases, but predicted almost 5 times as many false negative cases as it did true positive cases. This shows that the model is biased towards predicting negative cases. A cost matrix and boosted decision tree were initially used, but the boosted tree made interpretation too difficult and the cost matrix was ineffective at increasing the accuracy of the model.

```
#assess the accuracy of the model
test_results_predict <- predict(test_results_model, test_results_test[-4])

CrossTable(test_results_test$tested, test_results_predict,
           prop.chisq = FALSE, prop.c =FALSE,
           prop.r = FALSE,
           dnn = c('actual data', 'predicted data'))
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  155
##
##
##              | predicted data
##  actual data |        Y+ |        Y- | Row Total |
## -------------|-----------|-----------|-----------|
##           Y+ |         6 |        29 |        35 |
##              |     0.039 |     0.187 |           |
## -------------|-----------|-----------|-----------|
##           Y- |         7 |       113 |       120 |
##              |     0.045 |     0.729 |           |
## -------------|-----------|-----------|-----------|
## Column Total |        13 |       142 |       155 |
## -------------|-----------|-----------|-----------|
##
##
```

D. Predicting on the Full Dataset

After being trained on the observations that contained test results, the model was run on the entire dataset. This generated predictions of positive or negative tests for each observation.

```
data_recode_predict2 <- data_recode %>%
  select(state, hh_contact, non_hh_contact, n_leave_home, isolate_if_symp:avoid_pub_transport,
         avoid_guests:n_wash_sanitize)

data_recode_predict_no_state <- data_recode_predict2 %>%
  select(-state)

#add prediction column
data_recode_predict2$pred <- predict(test_results_model, data_recode_predict_no_state)

#check for NAs
sum(is.na(test_results))
```

```
## [1] 0
```

E. Map of Predictions

The below map shows the percentage of people sampled in a given state who were predicted to test positive for COVID-19. The states that show 0% of people in that state testing positive were often a result of not many people from that state being sampled.

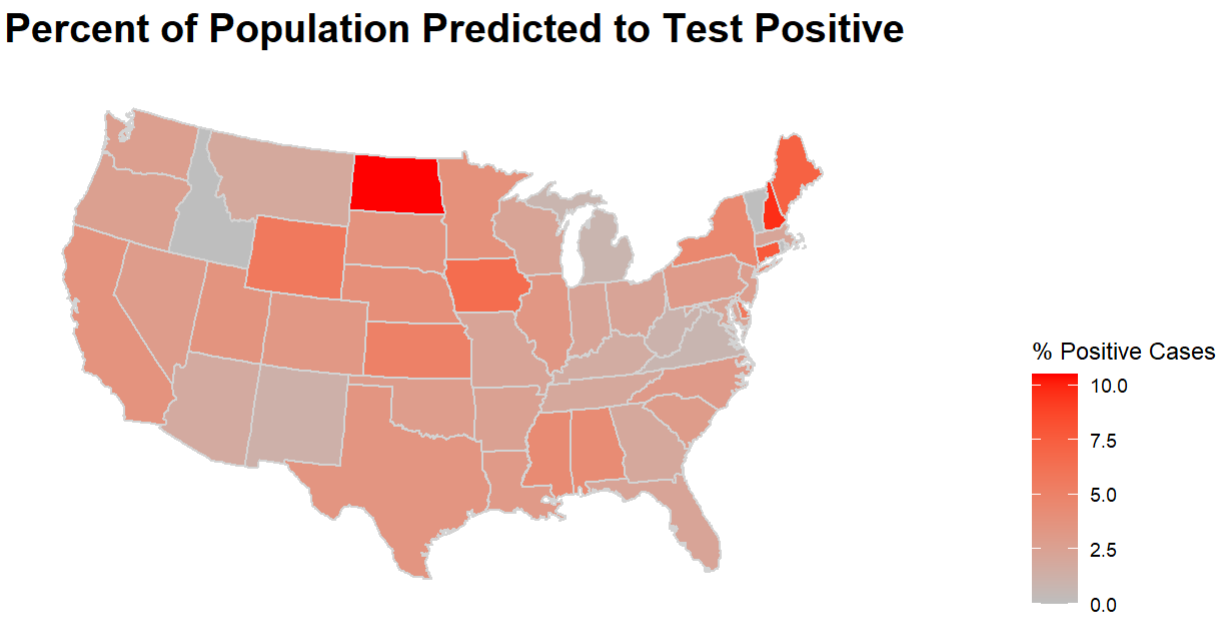
```
predict_map_data <- data_recode_predict2 %>%
  #make a column of the predictions as integers so positive cases can be counted
  mutate(pred_int = ifelse(pred == "Y+", 1, 0)) %>%
  #group by state and determine percent of positive cases
  group_by(state) %>%
  summarize(
    count = n(),
    positive_cases = sum(pred_int),
    percent_cases = positive_cases / count * 100
  ) %>%
  mutate(region = tolower(state)) %>%
  ungroup()

#table containing the data used to generate the map
head(predict_map_data, 20)
```

```
## # A tibble: 20 x 5
##    state                count positive_cases percent_cases region
##    <chr>                <int>          <dbl>         <dbl> <chr>
##  1 Alabama                256             11          4.30 alabama
##  2 Alaska                  59              6         10.2  alaska
##  3 Arizona                345              6          1.74 arizona
##  4 Arkansas               119              3          2.52 arkansas
##  5 California            1401             52          3.71 california
##  6 Colorado               244              8          3.28 colorado
##  7 Connecticut            141             11          7.80 connecticut
##  8 Delaware                52              3          5.77 delaware
##  9 District of Columbia    57              3          5.26 district of columbia
## 10 Florida               1061             24          2.26 florida
## 11 Georgia                457              9          1.97 georgia
## 12 Hawaii                  47              1          2.13 hawaii
## 13 Idaho                   90              0          0    idaho
## 14 Illinois               502             17          3.39 illinois
## 15 Indiana                267              6          2.25 indiana
## 16 Iowa                   107              7          6.54 iowa
## 17 Kansas                  98              5          5.10 kansas
## 18 Kentucky               245              4          1.63 kentucky
## 19 Louisiana              127              4          3.15 louisiana
## 20 Maine                   69              5          7.25 maine
```

```
us_states <- map_data("state")
predict_map_join <- us_states %>%
   left_join(predict_map_data, by = "region")
```

```
ggplot(data = predict_map_join,
   mapping = aes(x = long, y = lat,  group = group, fill = percent_cases))  +
  geom_polygon(color = "lightgrey") +
  scale_fill_continuous(low = "gray", high = "red") +
 coord_map(projection = "albers", lat0 = 39, lat1 = 45) +
  labs(title = "Percent of Population Predicted to Test Positive") +
  labs(fill = "% Positive Cases") +
  theme_map()+
   theme(plot.title = element_text(face = 'bold', size = 15, hjust = .5),
        legend.position = "right")
```



IV. Conclusions

Overall, it seems as if individual behaviors such as mask wearing and hand washing are associated with fewer positive cases when looked at in isolation. This can be seen in the graphs of mask wearing and handwashing. There are also strong regional patterns, with very frequent mask wearing in the northeast and infrequent mask wearing in the midwest. However, when these factors are looked at in combination with all the other behavior factors, the model generated to predict positve tests was not particularly accurate and some of its predictions seemed counterintuitive, such as predicting a positive test result for people who always avoid gatherings

V. Limitations/Recommendations

The inconsistencies in the model are likely a result of the fact that only 855 out of the 14,031 (~6%) observations in the dataset were associated with a positive or negative test result. While the entire dataset is representative of the United States, it's very unlikely that the small subset of observations that had test results associated with them is. Training a model on a small, unrepresentative dataset will not result in an effective model.

Despite the model's limited effectiveness, the graphs demonstrate that people who wash their hands or wear masks frequently are more likely to test negative for COVID-19 than people who do not. This is consistent with CDC guidelines so we recommend following CDC guidelines. We also recommend checking COVID-19 risk maps before traveling out of the state or county.