Exploring and Analyzing COVID-19 Confirmed Cases Growth and Hospital Capacity

Team 2: Yuran Zhu, Catherine King

05/01/2020

Contents

1. Data Summary **1**
Data source . . . 1
Graphical Summary . . . 2
Overview for nationwide . . . 2
Focus: California . . . 4

2. Methodology **7**
Logistic Growth Model . . . 7

3. Findings **7**
Confirmed Cases Growth in US . . . 7
Analyze and Predict on California . . . 7
Apply the Model to the Nationwide . . . 9
Hospitalizations in California . . . 12
Predict Hospitalizations for California . . . 13
Hospitalizations in Louisiana . . . 14
Predict Hospitalizations for Louisiana . . . 15
Compare Hospital Capacity with High Risk Proportions . . . 16

4. Conclusion and Discussion **17**

Appendix **17**
Gitlab Link for this Project . . . 17
Code . . . 17

This project explores pressing questions related to COVID-19. Currently, the United States is under intense pressure from the disease, with swiftly growing cases and high demand for medical resources. We use data on case statistics and hospitalization capacity, aiming to predict the case growth trend and hopitalization, including whether hospital capacity will be reached in certain locations.

1. Data Summary

Data source

- COVID-19 case data: from Johns Hopkins CSSE at https://github.com/CSSEGISandData/COVID-19 (up to April, 29, 2020)
- US hospital capacity data: from Harvard Global Health Institute at https://globalepidemics.org/our-data/hospital-capacity/

- High risk data: from the Dartmouth Institute at https://www.dartmouthatlas.org/covid-19/ (high risk is defined as age 65 and older with two or more chronic conditions)
- Hospitalization data: from the COVID Tracking Project, which is a volunteer organization organized by The Atlantic that publishes COVID-19 data. https://covidtracking.com. The data is broken up by state.
- HRR: from the Dartmouth Atlas Project project: https://www.dartmouthatlas.org/covid-19/hrr-mapping/. The Hospital Referral Regions are in a lot of the coronavirus data sets and are bigger regions than just counties, but much smaller than states.

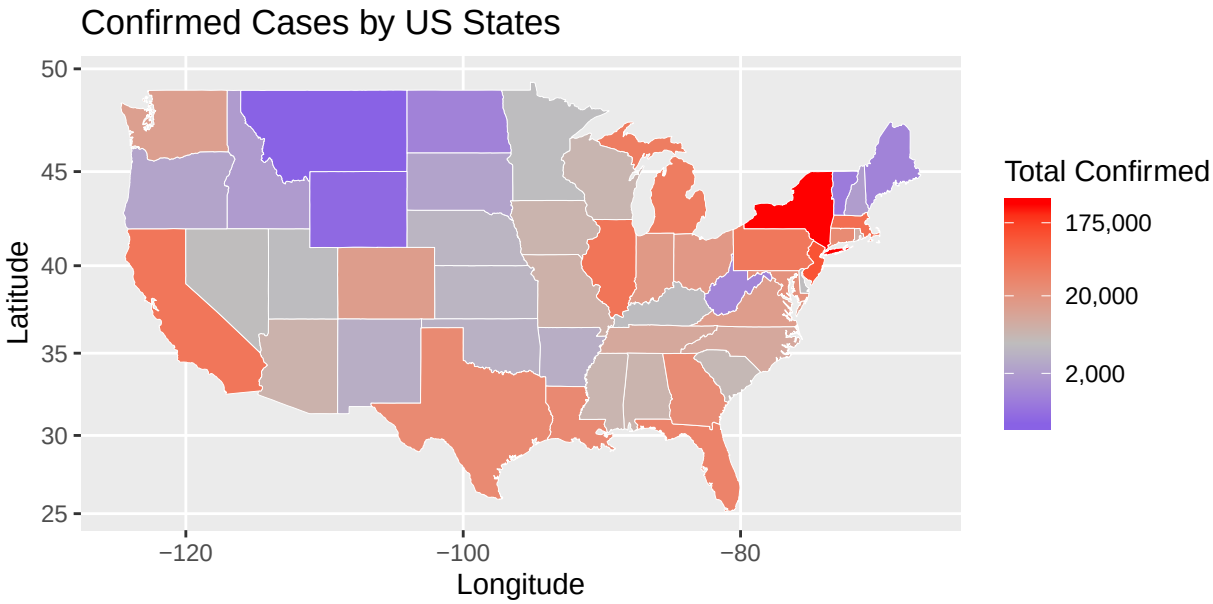
Our GitLab project that includes all of our data files can be found here: https://gitlab.com/cking412/coronavirus.

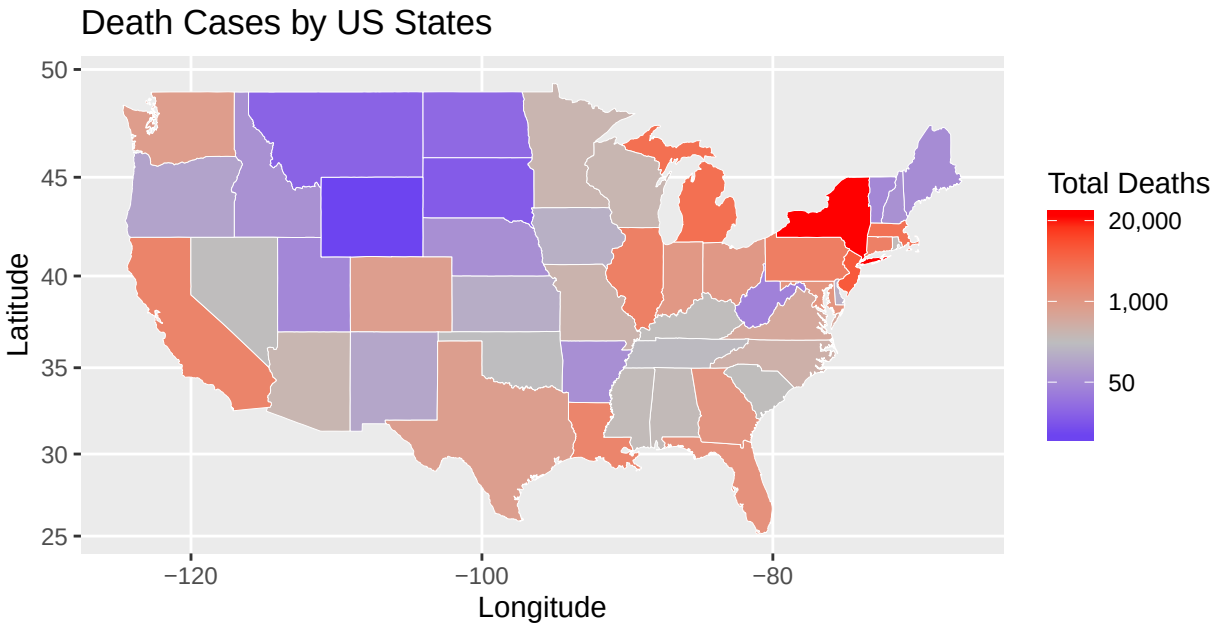
Graphical Summary

Overview for nationwide

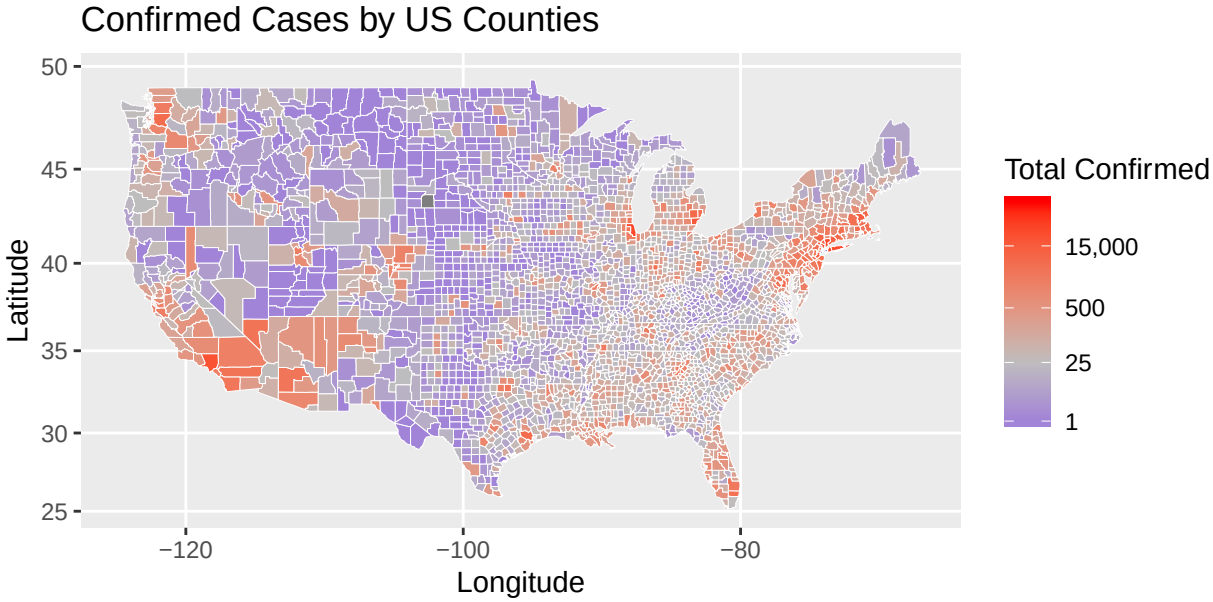
Using US map data, we first visualize US confirmed and death cases by states. It's clear that New York, New Jersey, Massachusetts, Illinois, and California are the five states undergoing the most severe pressure.

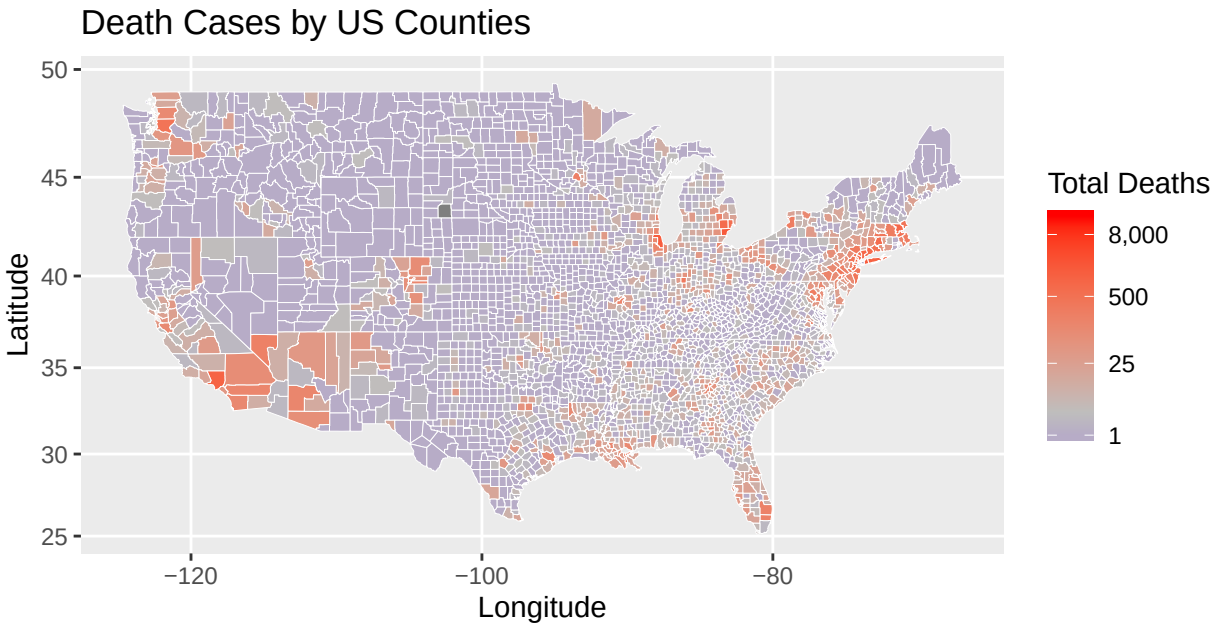
State	Total Confirmed	Total Deaths
New York	299,691	23,477
New Jersey	116,365	6,771
Massachusetts	60,265	3,405
Illinois	50,358	2,215
California	48,747	1,946





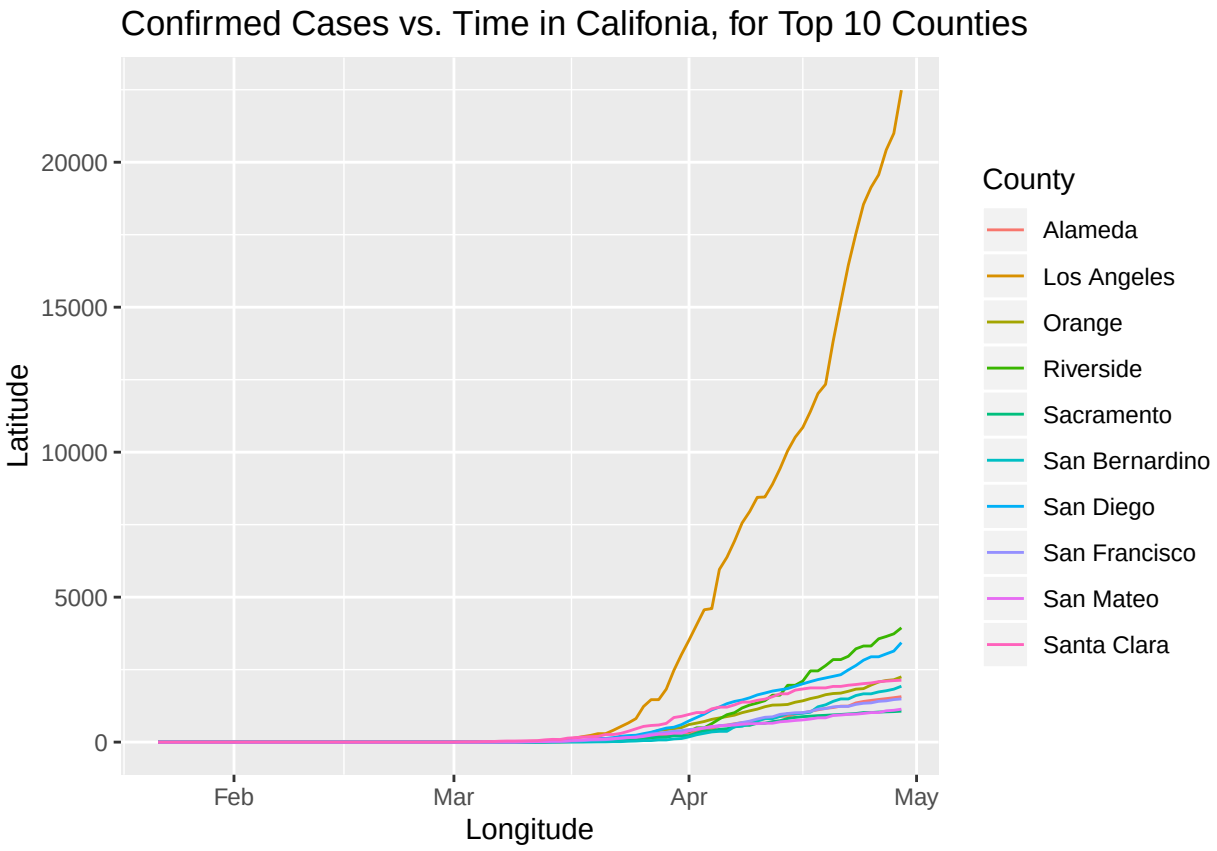
We also generate visuals by counties, and it turns out that counties located in the North West seem to have fewer cases. Here, one county has missing data: `Shanno, South Dakota` (fips code 46113).





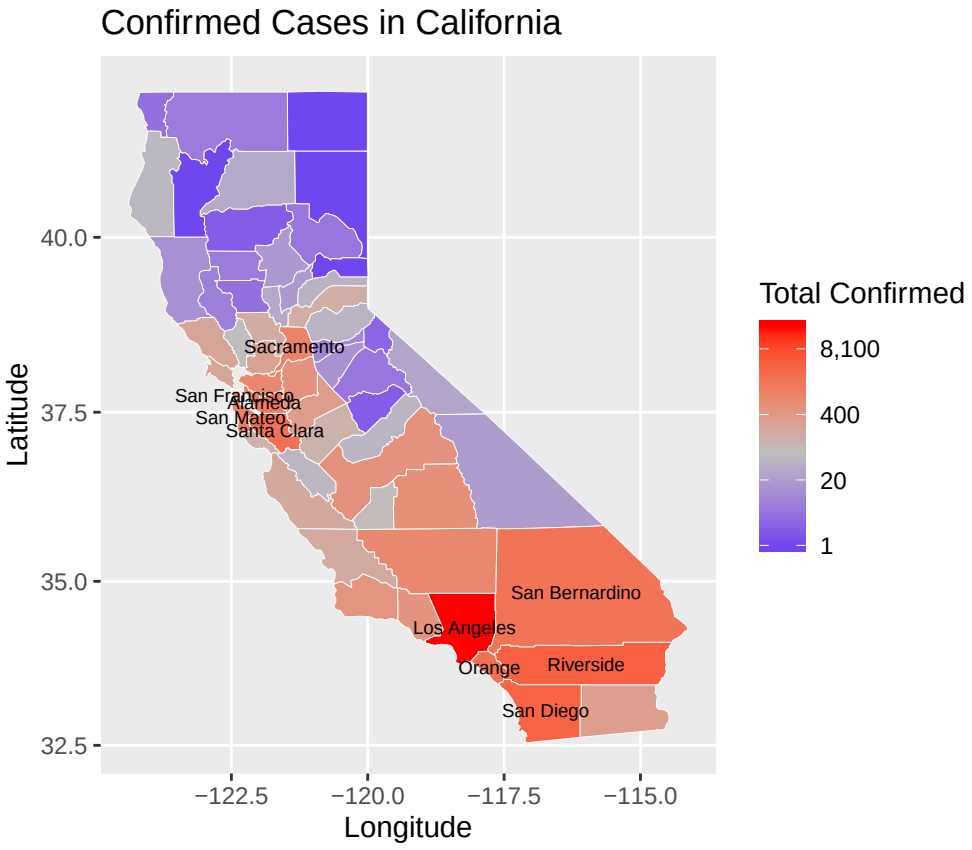
Focus: California

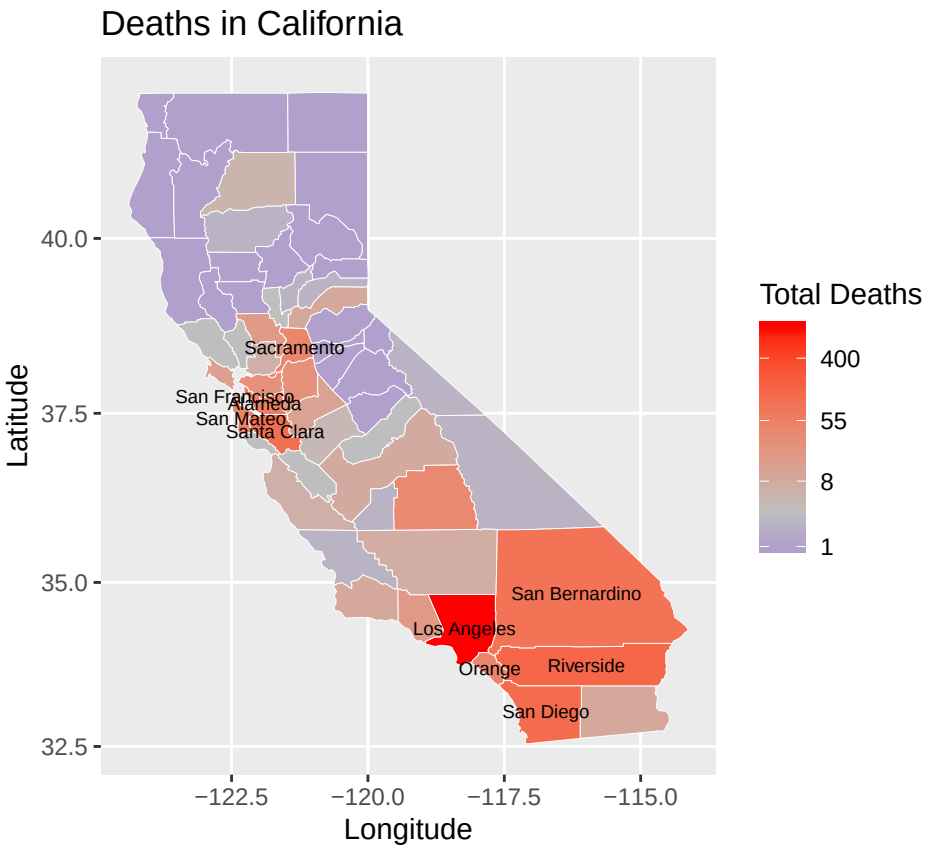
We selected one representative state, California, as the focus to further visualize and build the model on. From the plot of Califonia's confirmed cases, large discrepancies can be found. It's clear that Los Angeles is the most hard hit county in the whole state. After starting slowly in early March, the virus has now been spreading swiftly since late March.



To further explore the infectious conditions for different counties, plots are displayed based on California counties. The top 10 counties with the highest numbers case numbers are labelled. The following table summarizes those regions, with the population of the county also indicated. We can see that the hardest hit counties tend to have higher populations, likely due to the population density facilitating transmission throughout the communities.

County	Population	Confirmed	Deaths
Los Angeles	10,039,107	22,485	1,056
Riverside	2,470,546	3,942	143
San Diego	3,338,330	3,432	120
Orange	3,175,692	2,252	44
Santa Clara	1,927,852	2,134	107
San Bernardino	2,180,085	1,928	89
Alameda	1,671,329	1,568	57
San Francisco	881,549	1,490	23
San Mateo	766,573	1,136	48
Sacramento	1,552,058	1,068	42





We also find the following 4 counties haven't report any confirmed cases. Of note, those counties have lower populations than most other counties in California.

County	Population	Confirmed	Deaths
Lassen	30,573	0	0
Modoc	8,841	0	0
Sierra	3,005	0	0
Trinity	12,285	0	0

Meanwhile, we calculate the confirmed proportion of county population. It allows us to observe the infectious pattern more comprehensively, in combining case number and case rate. The figure below shows top 10 counties in California with the highest confirmed proportion.

County	Population	Confirmed	Proportion
Los Angeles	10,039,107	22,485	0.0022
Mono	14,444	26	0.0018
Alpine	1,129	2	0.0018
Imperial	181,215	319	0.0018
San Francisco	881,549	1,490	0.0017
Riverside	2,470,546	3,942	0.0016
San Mateo	766,573	1,136	0.0015
Tulare	466,195	626	0.0013
Santa Clara	1,927,852	2,134	0.0011
Santa Barbara	446,499	485	0.0011

To summarize, Southern California is suffering a lot from the disease. Los Angeles has the most confirmed cases with 22485, which is nearly 6 times that of the county with second most cases, Riverside.

2. Methodology

As for the data analytics part, we mainly investigate two questions:

- How does the case growth trend develop, and when will it possibly reach a steady state?
- As the viral infection continues to develop fast, will we reach the hospitalization capacity in certain states or regions in the near future?

Logistic Growth Model

The logistic growth model has been widely used to model population growth with limited resources and space. It can be used in epidemiology, to analyze pandemic dynamics, with a cumulative number of confirmed cases or deaths. The logistic growth model corresponds to the SI model in epidemic studies (S for Susceptible, I for Infectious). As now the primary method to control the pandemic in US is stay at home orders, we could use the logistic growth model to in the case of COVID-19 outbreak.

The differential form of the logistic model is as follows:

$$\frac{dN}{dt} = rN\frac{1-N}{K}$$

where K is the predicted maximum of confirmed cases (deaths), r is the grow rate, t is the number of days since the first case occured.

Then we can model the cumulative confirmed cases (deaths) N_t on day t with the equation:

$$N_t = \frac{K}{1 + (\frac{K}{N_0 - 1})e^{-rt}}$$

3. Findings

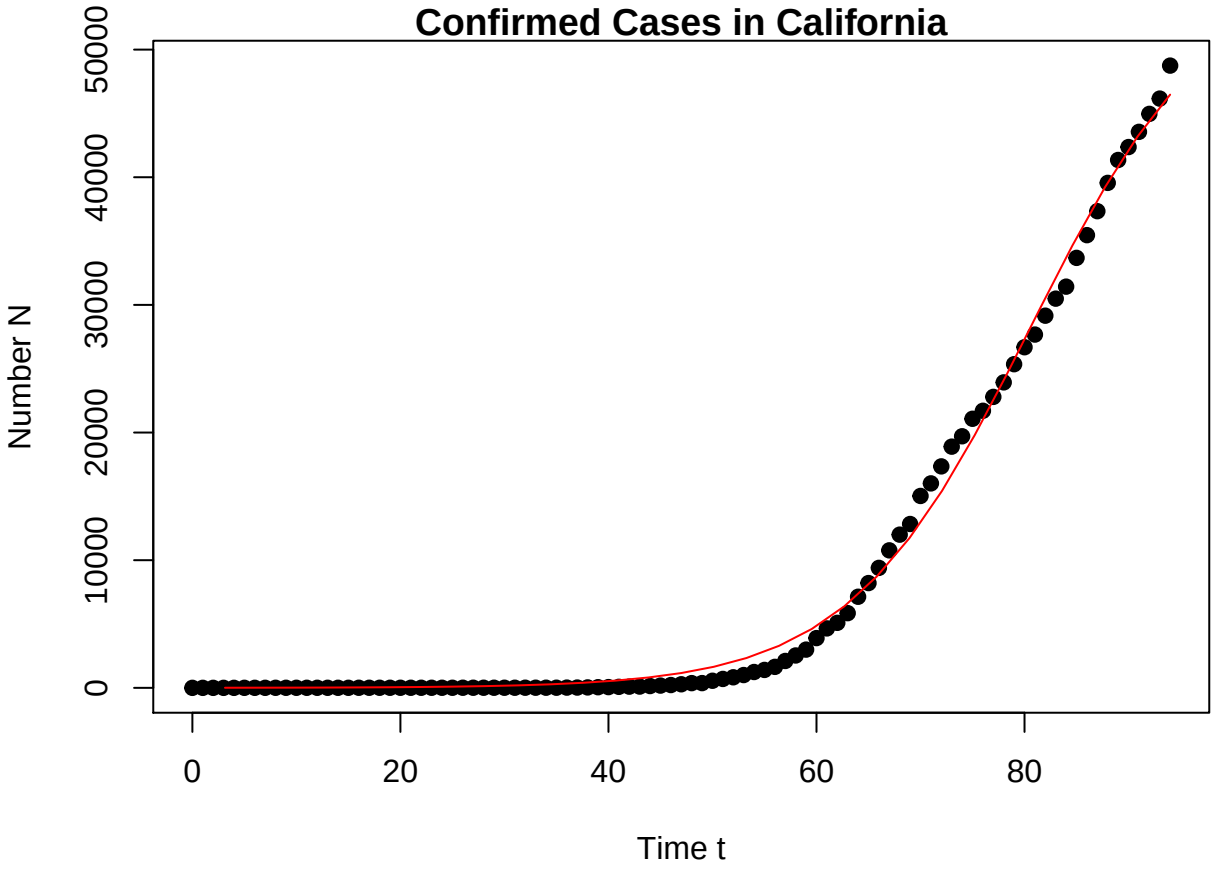
Confirmed Cases Growth in US

Analyze and Predict on California

First, we run the logistic growth model on California data, aiming to learn the growth rate and predict future days to reach a steady number of confirmed cases. Day 0 in the data is Jan, 26, 2020, when the first confirmed cases are reported. Here is the summary of results, indicating that the predicted maximum of confirmed cases may be around 56,318, assuming that logistic growth assumptions are met.

```
## Fit data to K / (1 + ((K - N0) / N0) * exp(-r * t)):
##      K   N0  r
##   val:   56318.027   5.271   0.115
##   Residual standard error: 911.5944 on 92 degrees of freedom
##
## Other useful metrics:
##   DT 1 / DT  auc_l   auc_e
##   6.02   1.7e-01 852635.48   833153.5
```

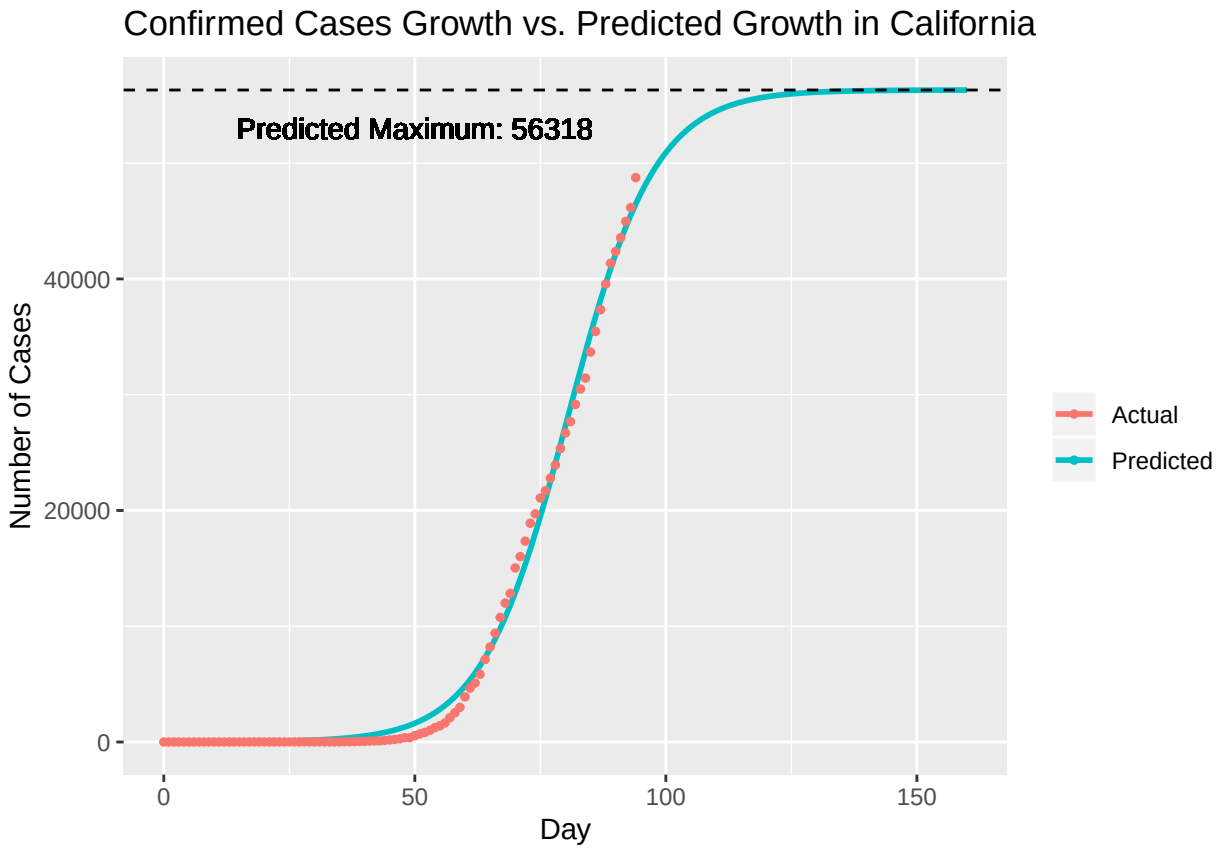
The plot shows that the fitted logistic curve seems to be a good fit to the raw data on number of confirmed cases.



Using the parameters retrieved, we can predict the future number of confirmed cases and we can find the approximate day of reaching the steady state. Based on the predicted values, we can calculate daily case increase. It turns out that the daily increase has reached the highest on day 82, i.e., April 17. The following table displays the predicted number of cases and daily increase from day 77 to day 87.

Day	Predicted Confirmed Cases	Increase per Day
77	22,515	1,537
78	24,089	1,573
79	25,688	1,600
80	27,304	1,616
81	28,925	1,621
82	30,542	1,616
83	32,142	1,601
84	33,717	1,575
85	35,257	1,540
86	36,752	1,495
87	38,196	1,444

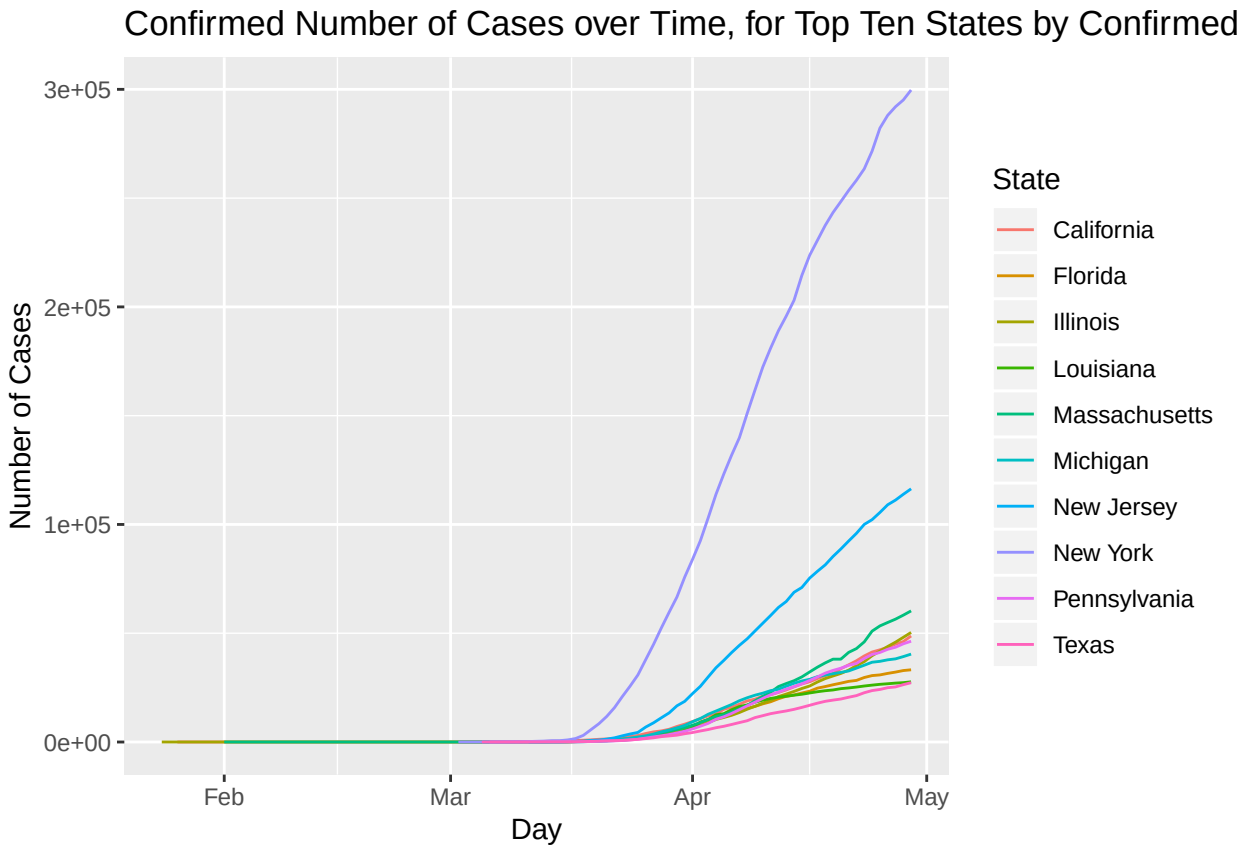
The figure shows the confirmed case growth vs. the predicted growth. We discover the estimated daily growth begins to be less than 10 starting on day 137 (June 11th) and less than 1 starting on day 158 (July 1st). Therefore, it seems that this COVID-19 outbreak will be under control in June. This does not make any predictions on potential future waves of the disease if and when social distancing guidelines are lifted.



Apply the Model to the Nationwide

Determine States with Current Highest Growth Rates/Capacity

We then apply the logistic growth model to the entire nationwide on an individual state basis, then predict the respective growth rate and maximum number of confirmed cases for each US state or territory.



The states that have the highest growth rate are shown below.

State	Growth Rate
Louisiana	0.213
Guam	0.211
Northern Mariana Islands	0.210
Virgin Islands	0.203
South Dakota	0.202
Idaho	0.200
Montana	0.200
Vermont	0.198
Hawaii	0.185
Puerto Rico	0.174

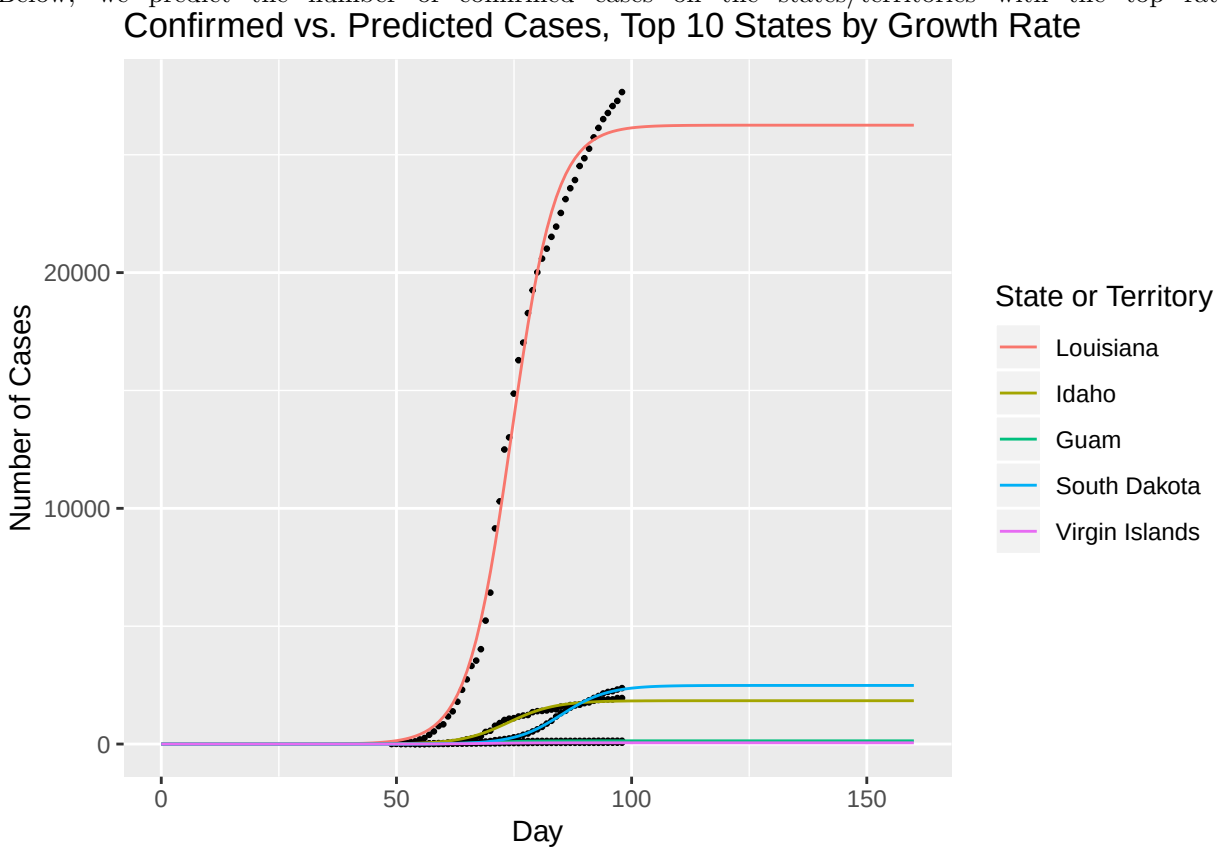
The states that have the highest capacities are shown below.

State	Capacity
New York	302,915
New Jersey	122,089
Massachusetts	74,254
Illinois	63,250
California	56,316
Pennsylvania	48,301
Michigan	39,119
Florida	33,639

State	Capacity
Iowa	32,988
Nebraska	30,108

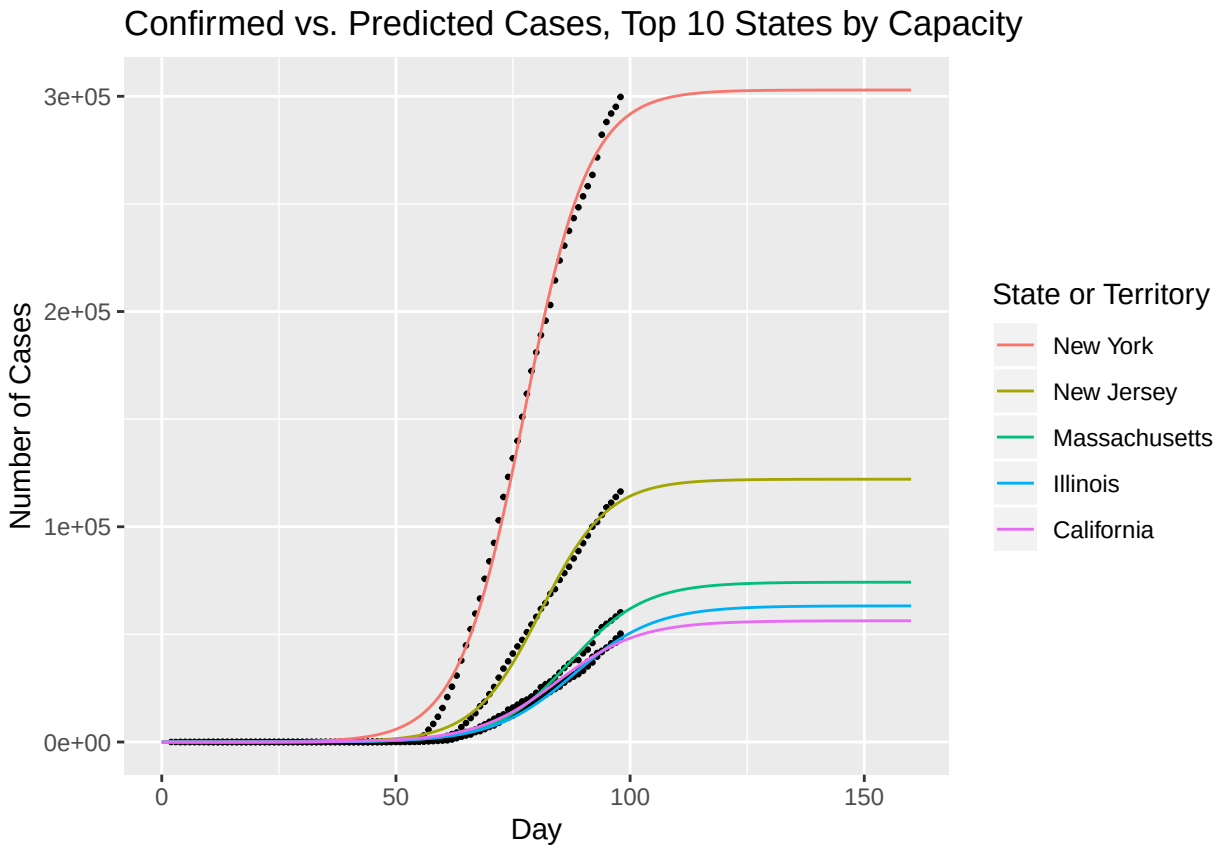
Predict on Representative States

Below, we predict the number of confirmed cases on the states/territories with the top rates.



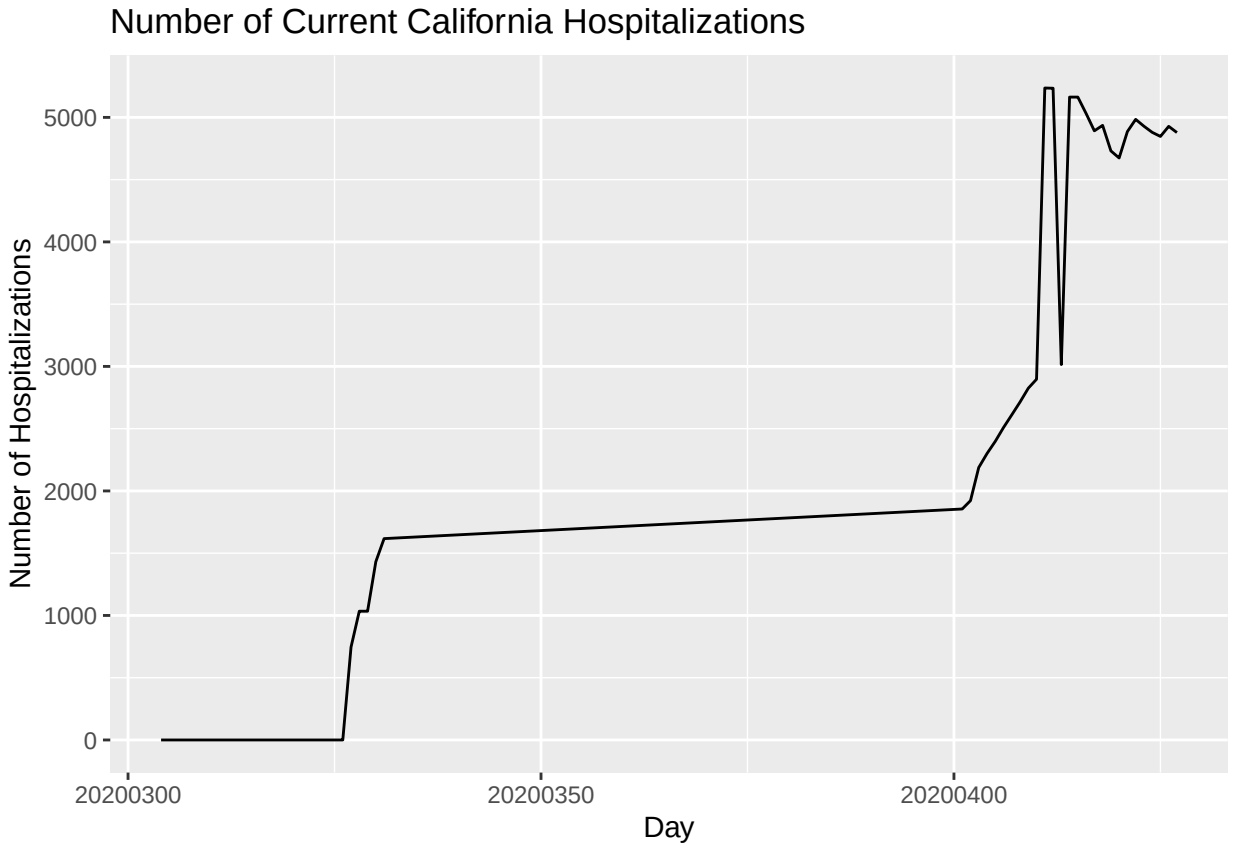
Next, we predict for those with higher capacity. The 5 states with the most highest maximum capacity is New York, New Jersey, Massachusetts, Illinois, and California. We visualize the predicted line vs. actual trend for those 5 states. It seems the number of confirmed cases in those states would reach the steady state in June, while New York may take longer time. These predictions seem to be better fits than those for areas with high rates, likely because there is more data for these more population-dense states.

State	Growth Rate	Predicted Maximum
New York	0.144	302,915
New Jersey	0.141	122,089
Massachusetts	0.126	74,254
Illinois	0.116	63,250
California	0.115	56,316



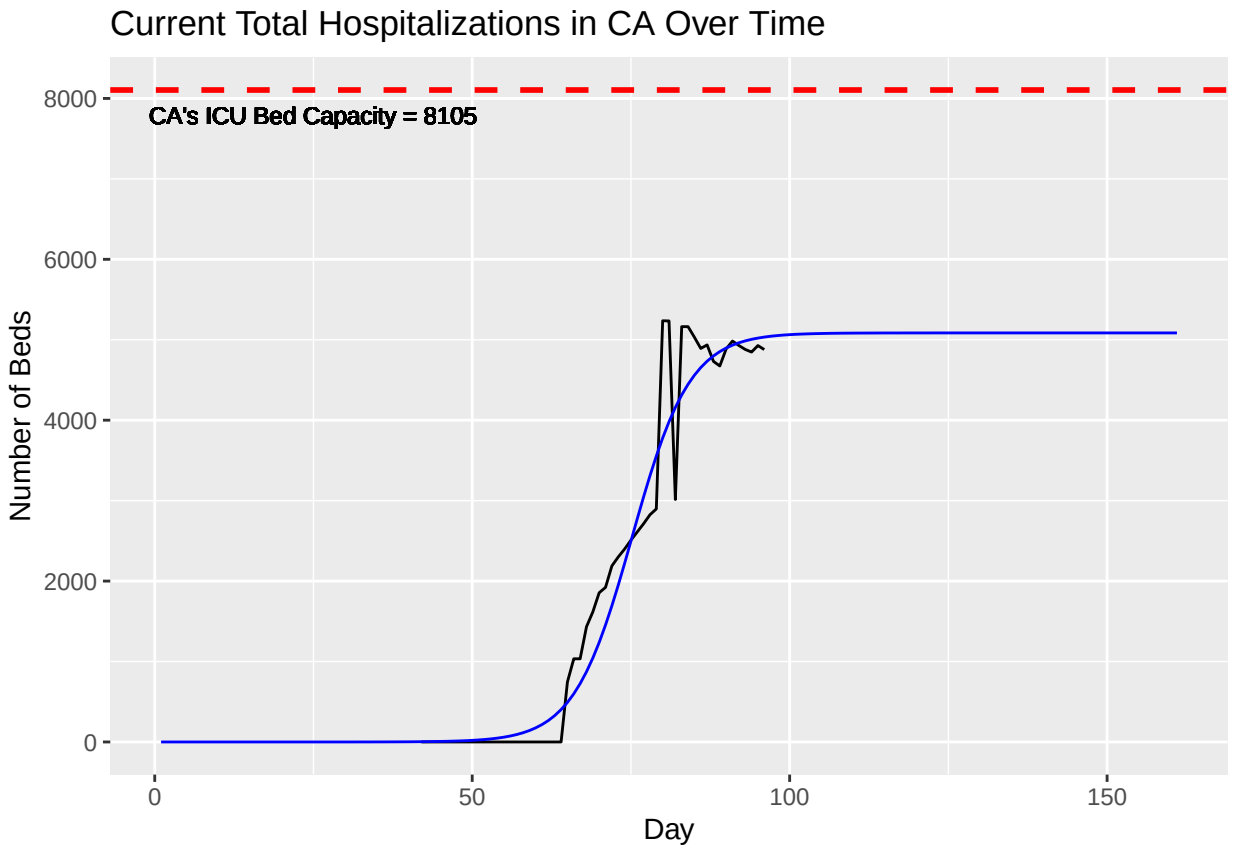
Hospitalizations in California

Next, we specifically looked at hospitalizations in our representative state: California. We investigated whether they would be reaching or exceeding their hospital bed capacity during this outbreak. The hospitalization data does not start until later March, so this estimate is likely less accurate.



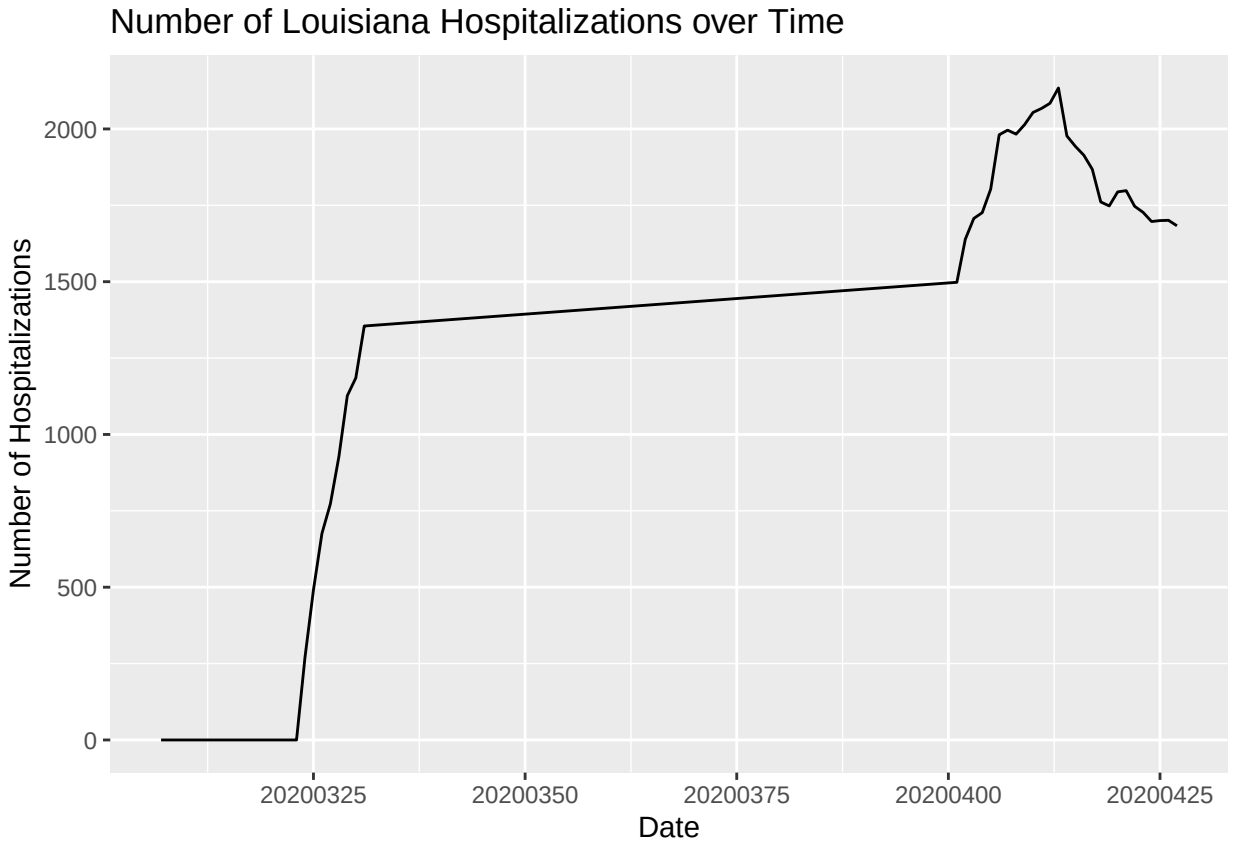
Predict Hospitalizations for California

Next, we predict future hospitalization needs for California as the outbreak continues, likely through early June as mentioned earlier.



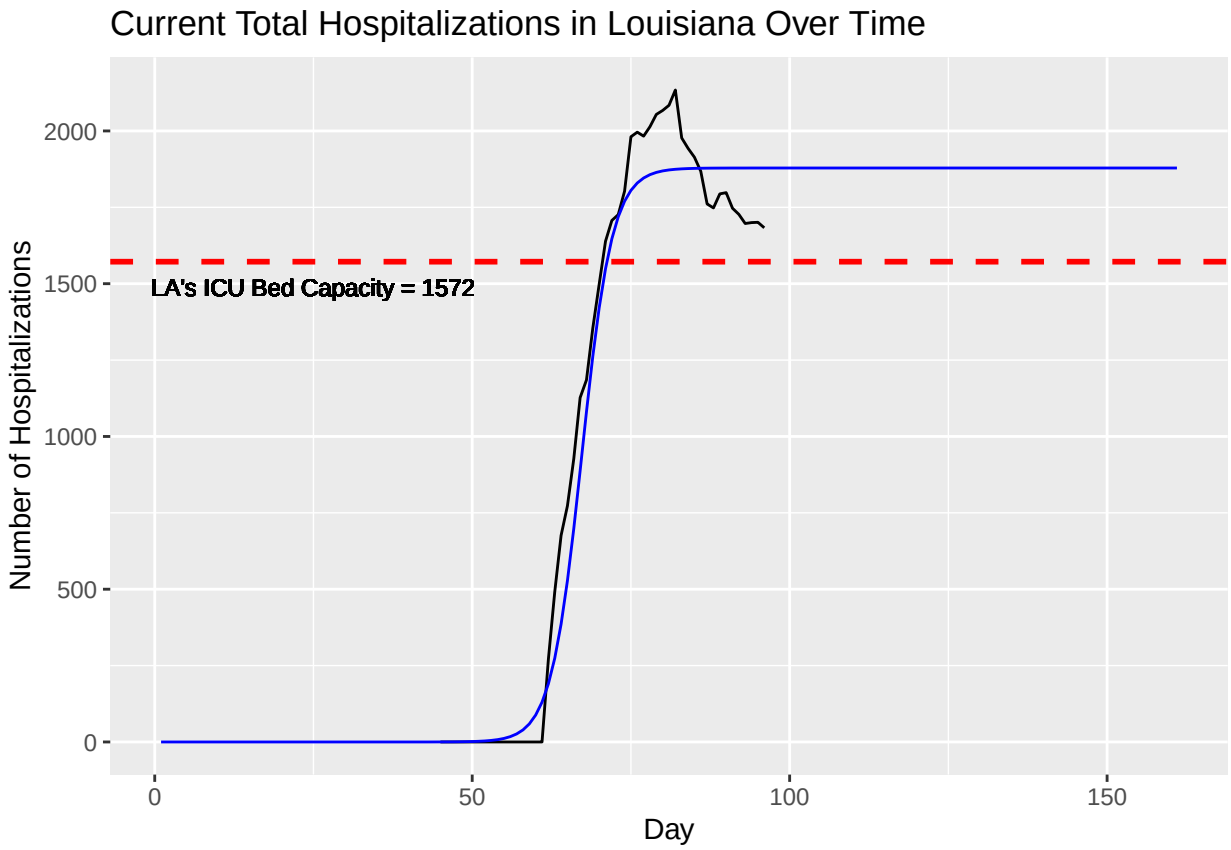
Hospitalizations in Louisiana

Next, we specifically looked at hospitalizations in Lousiana. It currently is the highest growth state and ranks 11th on highest possible capacity (in terms of highest number of possible cases).



Predict Hospitalizations for Louisiana

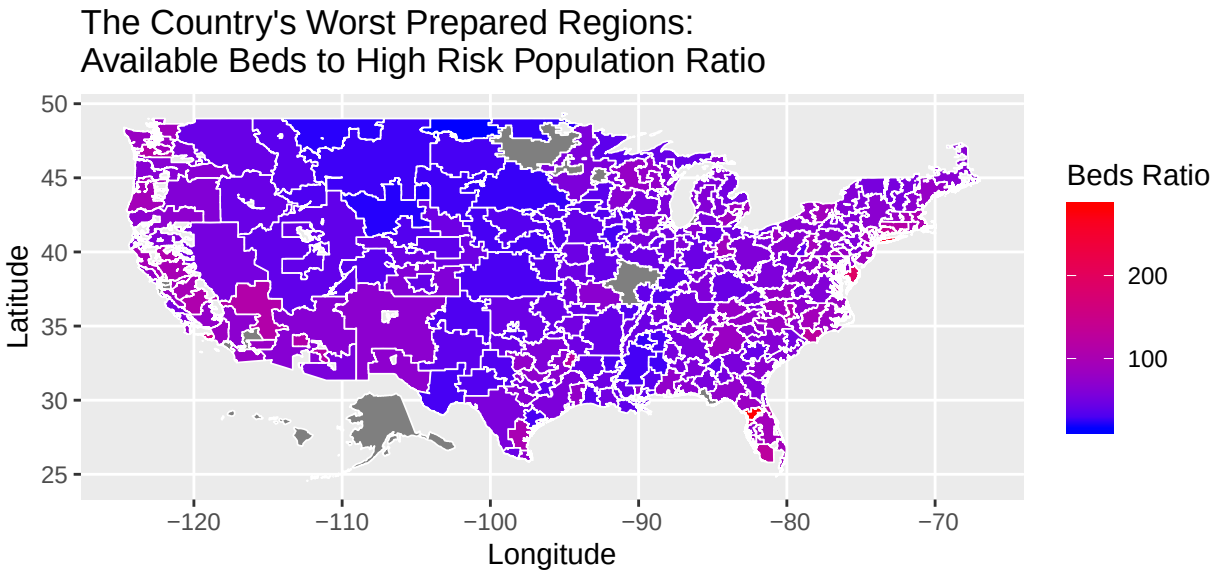
It does seem like Louisiana has already exceeded their ICU bed capacity. They likely will need to use other types of hospital beds. It is likely that not all hospitalizations need intensive care, but it is good to be aware of limitations.



Compare Hospital Capacity with High Risk Proportions

Next, we compared the general hospital capacity in Hospital Referral Regions with the proportion of the local population that is considered to be high risk for COVID-19. These are individuals over the age of 65 with at least 2 underlying medical conditions.

The following plot shows areas with worst bed to high risk population ratio. The red areas correspond with the Northeast and parts of California, where COVD-19 has hit hardest.



4. Conclusion and Discussion

The US has been deeply affected by the COVID-19 outbreak in terms of public health, the economy and many other aspects. Therefore, the predictions related to this epidemic disease is topical and aims to help policymakers determine approprate measures in advance. The project uses logistic growth model in predicting the confirmed cases in the near future. The results indicate that number of confirmed cases in US will continue increasing until late June and Early July. However, the model is based on the assumption of limited space and resources. If the current stay at home order is lifted earlier, it may take longer time to reach a steady state or hospital capacity will be exceeded.

In addition to using logistic modeling to predict the end of the COVID-19 outbreaks across the US, we also investigated hospital capacity. By implementing stay at home orders and thereby lowering the capacity in the model ("flattening the curve"), most states are well within their capacity in terms of potential number of hospitalizations. We did see that Louisiana may exceed their ICU capacity if everyone that is hospitalized with COVID-19 needs intensive care and no other people need intensive care. However, most COVID patients will not need intensive care, and because of flattening the curve states are well-equipped to deal with the hospital burdens. Overall, modeling is an effective planning tool for dealing with pandemics and pandemic response. They can help policymakers understand the effects of certain policies and help them prepare for potential shortfalls in various locations.

Appendix

Gitlab Link for this Project

Web link: https://gitlab.com/cking412/coronavirus

Code

```
knitr::opts_chunk$set(echo=FALSE, warning=FALSE, message=FALSE)
library(tidyverse)
library(ggplot2)
library(growthcurver)
library(maps)
library(mapproj)
library(sf)
library(ggrepel)
library(knitr)
library(stringr)
# Data Import
# US data on confirmed/deaths cases
us.confirmed <- read_csv("time_series_covid19_confirmed_us.csv")
us.deaths <- read_csv("time_series_covid19_deaths_us.csv")

# High risk proportion
high.risk <- read_csv("high-risk-from-covid-19-hsa-hrr-v3.csv", skip = 1)

# Hospital capacity
capacity <- read_csv("hrr_hospital_capacity.csv")

# Hospitalizations by State
hospitalizations <- read_csv("US_Hospitalizations_By_State.csv")

# Fips code table for county
fips.table <- read_csv("fips.table.csv")
```

```r
# Tidy the Data
# Spread table and modify column names
us.confirmed <- us.confirmed %>%
  pivot_longer(12:ncol(us.confirmed),
               names_to = "time", values_to = "confirmed")

us.deaths <- us.deaths %>%
  pivot_longer(13:ncol(us.deaths),
               names_to = "time", values_to = "deaths")

colnames(high.risk) <- c("hospital.area", "HSA.name", "HRR.id", "HRR.name", "population",
                         "high.risk.prop")

# Extract HRR State and County for `high.risk`
HRR_state <- rep(0, nrow(high.risk))
for (i in 1:length(HRR_state)) {
  HRR_state[i] = unlist(str_split(high.risk$HRR.name[i], pattern = "- "))[1]
}
HRR_region <- rep(0, nrow(high.risk))
for (i in 1:length(HRR_region)) {
  HRR_region[i] = unlist(str_split(high.risk$HRR.name[i], pattern = "- "))[2]
}
high.risk <- high.risk %>%
  mutate(state = HRR_state,
         region = HRR_region,
         high.risk.pop = round(population * high.risk.prop))

# Same for `capacity`
HRR_state.2 <- rep(0, nrow(capacity))
for (i in 1:length(HRR_state.2)) {
  HRR_state.2[i] = unlist(str_split(capacity$HRR[i], pattern = ", "))[2]
}
HRR_region.2 <- rep(0, nrow(capacity))
for (i in 1:length(HRR_region.2)) {
  HRR_region.2[i] = unlist(str_split(capacity$HRR[i], pattern = ", "))[1]
}
capacity <- capacity %>%
  mutate(state = HRR_state.2,
         region = HRR_region.2)

# Sum high risk by state
high.risk.by.state <-  high.risk %>%
  group_by(state) %>%
  summarize(sum.pop = sum(population),
            sum.risk.pop = sum(high.risk.pop))

# Combine the confirmed and deaths data
us.combined <- left_join(us.deaths, us.confirmed,
                         by = colnames(us.confirmed)[1:12])

# Convert time to Date format
us.combined$time <- as.Date(us.combined$time, "%m/%d/%y")
```

```r
# Graphical Summary
# Import US statey map data
usa <- map_data("state")

# Most recent case data, by State
us.total <- us.combined %>%
  filter(time == last(time)) %>%
  group_by(Province_State) %>%
  summarize(sum.confirmed = sum(confirmed, na.rm = TRUE),
            sum.deaths = sum(deaths, na.rm = TRUE))
us.total$Province_State = tolower(us.total$Province_State)

# Display top 5 states with highest confirmed cases and death
top.5 <- us.total %>%
  arrange(desc(sum.confirmed, sum.deaths)) %>%
  head(5)

top.5$Province_State <- top.5 %>% pull(Province_State) %>% str_to_title()
kable(top.5, col.names = c("State", "Total Confirmed", "Total Deaths"),format.args = list(big.mark = ",

# Join case statistics with map data
us.total$Province_State = tolower(us.total$Province_State)
us.total.map <- left_join(usa, us.total, by = c("region" = "Province_State"))

#To get integer labels on the graphs rather than continuous labels need to manually specify

confirmed_breaks <- c(2000, 20000, 175000)
confirmed_labels <- format(confirmed_breaks, big.mark = ",", trim = TRUE)
ggplot(data = us.total.map, aes(x = long, y = lat, group = group, fill = sum.confirmed)) +
  geom_polygon(color = "white", size=.1) +
  scale_fill_gradient2(low = "blue", mid = "grey", high = "red", trans="log", midpoint = log(median(us.t
  coord_map() +
  labs(title = 'Confirmed Cases by US States') + labs(fill = "Total Confirmed") +
  labs(x = "Longitude", y = "Latitude")


#To get integer labels on the graphs rather than continuous labels
death_breaks <- c(50,1000, 20000)
death_labels <- format(death_breaks, big.mark = ",", trim = TRUE)
ggplot(data = us.total.map, aes(x = long, y = lat, group = group, fill = sum.deaths)) +
  geom_polygon(color = "white", size=.1) +
  scale_fill_gradient2(low = "blue", mid = "grey", high = "red",
                       trans = "log", midpoint = log(median(us.total$sum.deaths)),
                       breaks = death_breaks, labels = death_labels) +
  coord_map() +
  labs(title = 'Death Cases by US States') + labs(fill = "Total Deaths") +
  labs(x = "Longitude", y = "Latitude")

# Import US county map data, join with fips code
usa.2 <- map_data("county") %>%
  left_join(fips.table, by = c("region" = "region", "subregion" = "subregion"))

# Most recent case data, by county
```

```r
us.sum <- us.combined %>%
  filter(time == last(time)) %>%
  mutate(add.confirmed = confirmed +1, # avoid -Inf when log transformation
         add.deaths = deaths +1)
us.sum.map <- left_join(usa.2, us.sum, by = c("fips" = "FIPS"))

# Plot for confirmed cases and deaths by US county
cases_breaks <- c(1, 25, 500, 15000)
cases_labels <- format(cases_breaks, big.mark = ",", trim = TRUE)
ggplot(data = us.sum.map, aes(x = long, y = lat, group = group, fill = add.confirmed)) +
  geom_polygon(color = "white", size=.1) +
  coord_map() +
  scale_fill_gradient2(low = "blue", mid = "grey", high = "red",
                       trans = "log", midpoint = log(median(us.sum.map$add.confirmed, na.rm=TRUE)),
                       breaks = cases_breaks, labels = cases_labels) +
  labs(title = 'Confirmed Cases by US Counties', fill = "Total Confirmed") +
  labs(x = "Longitude", y = "Latitude")

deaths_breaks <- c(1, 25, 500, 8000)
deaths_labels <- format(deaths_breaks, big.mark = ",", trim = TRUE)
ggplot(data = us.sum.map, aes(x = long, y = lat, group = group, fill = add.deaths)) +
  geom_polygon(color = "white", size=.1) +
  coord_map() +
  scale_fill_gradient2(low = "blue", mid = "grey", high = "red",
                       trans = "log", midpoint = 1,
                       breaks = deaths_breaks, labels = deaths_labels) +
  labs(title = 'Death Cases by US Counties', fill = "Total Deaths") +
  labs(x = "Longitude", y = "Latitude")

# Analyze on California
california <- us.combined %>%
  filter(Province_State == "California")
# Top 10 with highest number of cases
california.top.10 <- california %>%
  filter(time == last(time)) %>%
  arrange(desc(confirmed, deaths)) %>%
  head(10)

ggplot(data = california[california$Admin2 %in% california.top.10$Admin2,],
       aes(x = time, y = confirmed, color = Admin2)) +
  geom_line() +
  labs(title = "Confirmed Cases vs. Time in Califonia, for Top 10 Counties") +
  labs(x = "Longitude", y = "Latitude") + labs(color = "County")

# Nice output for top 10 counties
kable(california.top.10[, c("Admin2", "Population", "confirmed", "deaths")],
      col.names = c("County", "Population", "Confirmed", "Deaths"), format.args = list(big.mark = ","))

# CA map data
ca.county <- map_data("county", "california")

california.latest <- california %>%
  filter(time == last(time))  %>%
```

```
  mutate(add.confirmed = confirmed +1, # avoid -Inf when log transformation
         add.deaths = deaths +1)
california.latest$Admin2 = tolower(california.latest$Admin2)
ca.map <- left_join(ca.county, california.latest, by = c("subregion" = "Admin2"))

# Plots for CA cases
ca_cases_breaks <- c(1, 20, 400, 8100)
ca_cases_labels <- format(ca_cases_breaks, big.mark = ",", trim = TRUE)
ggplot(data = ca.map) +
  geom_polygon(aes(x=long, y = lat, group = group, fill = add.confirmed),
               color='white', size=.1) +
  scale_fill_gradient2(low = 'blue', mid = 'grey', high = 'red',
                trans='log',  midpoint = log(median(ca.map$add.confirmed)),
                breaks = ca_cases_breaks, labels = ca_cases_labels) +
  coord_map() +
  labs(title = 'Confirmed Cases in California') + labs(fill = "Total Confirmed") +
  labs(x = "Longitude", y = "Latitude") +
  geom_text(data = california.top.10,
            aes(x=Long_, y = Lat, label = Admin2),
             size = 2.5, col = "black")

ca_deaths_breaks <- c(1, 8, 55, 400)
ggplot(data = ca.map) +
  geom_polygon(aes(x=long, y = lat, group = group, fill = add.deaths),
               color='white', size=.1) +
  scale_fill_gradient2(low = 'blue', mid = 'grey', high = 'red',
                       trans = "log", midpoint = log(median(ca.map$add.deaths)),
                       breaks = ca_deaths_breaks, labels = ca_deaths_breaks) +
  coord_map() +
  geom_text(data = california.top.10,
            aes(x=Long_, y = Lat, label = Admin2),
             size = 2.5, col = "black") +
  labs(title = 'Deaths in California') + labs(fill = "Total Deaths") +
  labs(x = "Longitude", y = "Latitude")

# Counties have zero case
california.low <- california %>%
  filter(time == last(time)) %>%
  filter(confirmed == 0 & Population != 0)
kable(california.low[, c("Admin2", "Population", "confirmed", "deaths")],
      col.names = c("County", "Population", "Confirmed", "Deaths"), format.args = list(big.mark = ","))

# Caluculate comfirmed rate
ca.prop.top.10 <- california %>%
  filter(time == last(time)) %>%
  mutate(prop = confirmed/Population) %>%
  arrange(desc(prop)) %>%
  head(10)
kable(ca.prop.top.10[, c("Admin2", "Population", "confirmed", "prop")],
      col.names = c("County", "Population", "Confirmed", "Proportion"),
      digits = c(0,0,0,4),
      format.args = list(big.mark = ","))
```

```r
# Sum up confirmed cases for California
california.sum <- us.combined %>%
  filter(Province_State == "California") %>%
  group_by(time) %>%
  summarize(sum.confirmed = sum(confirmed),
            sum.deaths = sum(deaths))

# Remove 0 confirmed cases
california.sum <- california.sum %>%
  filter(sum.confirmed != 0)

# Assign variable day from 0
california.sum$day = 0:(nrow(california.sum)-1)

# Run the logistic growth model
gc.fit <- SummarizeGrowth(california.sum$day, california.sum$sum.confirmed)
gc.fit

# Plot the raw data vs. the fitted logistic curve
plot(gc.fit, main = "Confirmed Cases in California")

# Retrieve parameters
k <- unlist(gc.fit$vals['k'])
n0 <- unlist(gc.fit$vals['n0'])
r <- unlist(gc.fit$vals['r'])

# Make prediction until day 160
california.pred <- data.frame(day = 0:160) %>%
  mutate(pred = k/(1 + ((k - n0)/n0) * exp(-r * day)))

# Calculate daily increase
growth <- rep(0, nrow(california.pred))
growth[1] = NA
for (i in 2:nrow(california.pred)) {
  growth[i] = california.pred$pred[i] - california.pred$pred[i-1]
}
california.pred$growth = growth

# 5 days prior/after the day with max growth
# which.max(california.pred$growth)
ca.pred.partial <- california.pred %>%
  filter(day %in% c(77:87))
kable(ca.pred.partial,
      col.names = c("Day", "Predicted Confirmed Cases", "Increase per Day"),
      digits = c(0, 0, 0),
      format.args = list(big.mark = ","))

# Actual confirmed vs. predicted
ggplot(data = california.pred, mapping = aes(x = day, y = pred, color = "Predicted")) +
  geom_line(size = 1) +
  geom_point(data = california.sum,
             mapping = aes(x = day, y = sum.confirmed, color = "Actual"),
             size = 1) +
```

```
  labs(title = "Confirmed Cases Growth vs. Predicted Growth in California",
       y = "Number of Cases", x = "Day", color = "") +
  geom_hline(yintercept = 56318, linetype="dashed", size = .5) +
  geom_text(aes(x = 50, y = 53000, label = "Predicted Maximum: 56318"),color = "black")

# Sum confirmed cases for all states
state.sum <- us.combined %>%
#  filter(Province_State == "California") %>%
  group_by(time, Province_State) %>%
  summarize(sum.confirmed = sum(confirmed),
         sum.deaths = sum(deaths))

# Remove 0 confirmed cases
state.sum <- state.sum %>%
  filter(sum.confirmed != 0)

# Get top 10 states in terms of latest number of confirmed cases to plot
# (otherwise 50 states too many to see at once)
top.ten.states <- state.sum %>%
  group_by(Province_State) %>%
  arrange(desc(sum.confirmed)) %>%
  mutate (latest.confirmed = first(sum.confirmed)) %>%
  distinct(Province_State, latest.confirmed)
top.ten.states <- top.ten.states[1:10,1]

top.ten.sum <- state.sum %>%
  filter(Province_State %in% top.ten.states$Province_State) %>%
  group_by(Province_State)

ggplot(data = top.ten.sum, aes(x = time, y = sum.confirmed, color = Province_State)) +
  geom_line() + labs(x = "Day", y = "Number of Cases", color = "State") +
  labs(title = "Confirmed Number of Cases over Time, for Top Ten States by Confirmed Cases")

# Assign variable day from 0
state.sum <- state.sum %>% arrange(time)
state.sum$day = 0
state.sum$day <- as.numeric(difftime(as.character(state.sum$time),
                                     "2020-01-22",units = "days"))

# Run the model on all states
# number of unique US locations is 57, sorted alphabetically
loc <- sort(unique(state.sum$Province_State))
gc.fit.all <- vector("list", length(loc)) # list of all the fits
r.all <- data.frame()
k.all <- data.frame()
i <- 1  # index for the list
for (state in loc) {
  test <- state.sum %>% filter(Province_State == state)
  gc.fit.all[[i]] <- SummarizeGrowth(test$day, test$sum.confirmed)

  r.all[i, 1] <- state
  r.all[i, 2] <- unlist(gc.fit.all[[i]]$vals['r'])
  k.all[i, 1] <- state
```

```
  k.all[i, 2] <- unlist(gc.fit.all[[i]]$vals[['k']])

  i <- i + 1
}

# order states by top growth rate
ordered.states <- r.all[order(r.all$V2, decreasing = TRUE),]

# highest growth rate states
high <- ordered.states[1:10, ]
kable(high[, c("V1", "V2")],
      col.names = c("State", "Growth Rate"),
      row.names = FALSE,
      digits = c(0,3))

# order states by top capacity
ordered.states2 <- k.all[order(k.all$V2, decreasing = TRUE),]

# highest capacity states
high.cap <- ordered.states2[1:10, ]
kable(high.cap[, c("V1", "V2")],
      col.names = c("State", "Capacity"),
      row.names = FALSE,
      digits = c(0,0), format.args = list(big.mark = ","))

# List of highest growth rate states/territories
high.rate <- c("Louisiana", "Idaho", "Guam", "South Dakota", "Virgin Islands")
loc.high <- match(high.rate, loc)
state.pred <- vector("list",5)
j <- 1
for (state in loc.high) {
  k <- unlist(gc.fit.all[[state]]$vals['k'])
  n0 <- unlist(gc.fit.all[[state]]$vals['n0'])
  r <- unlist(gc.fit.all[[state]]$vals['r'])
  state.pred[[j]] <- data.frame(day = 0:160) %>%
    mutate(pred = k/(1 + ((k - n0)/n0) * exp(-r * day)))
  j <- j + 1
}
state.pred <- as.data.frame(state.pred) %>%
  select(c(Day = day, Louisiana = pred, Idaho = pred.1, Guam = pred.2,
           `South Dakota` = pred.3, `Virgin Islands` = pred.4)) %>%
  pivot_longer(high.rate, names_to = "State")

high.state.sum <- state.sum %>% filter(Province_State %in% high.rate)
state.pred$State <-  factor(state.pred$State, levels = high.rate )

# Plot
ggplot(data = high.state.sum,
       mapping = aes(x = day, y = sum.confirmed, color = Province_State)) +
  geom_point(color = "black", size = .5) +
  geom_line(data = state.pred, mapping = aes(x = Day, y = value, color = State)) +
  labs(title = "Confirmed vs. Predicted Cases, Top 10 States by Growth Rate") +
  labs(x = "Day", y = "Number of Cases", color = "State or Territory")
```

```r
# List of states with highest capacity
ordered.states.all <- left_join(r.all, k.all, by = "V1")
ordered.states.all <- ordered.states.all %>%
  arrange(desc(V2.y))
kable(head(ordered.states.all, 5),
      col.names = c("State", "Growth Rate", "Predicted Maximum"),
      digits = c(0,3,0), format.args = list(big.mark = ","))

high.capa <- c("New York", "New Jersey", "Massachusetts", "Illinois", "California")
loc.high.2 <- match(high.capa, loc)
state.pred.2 <- vector("list",5)
j <- 1
for (state in loc.high.2) {
  k <- unlist(gc.fit.all[[state]]$vals['k'])
  n0 <- unlist(gc.fit.all[[state]]$vals['n0'])
  r <- unlist(gc.fit.all[[state]]$vals['r'])
  state.pred.2[[j]] <- data.frame(day = 0:160) %>%
    mutate(pred = k/(1 + ((k - n0)/n0) * exp(-r * day)))
  j <- j + 1
}
state.pred.2 <- as.data.frame(state.pred.2) %>%
  select(c(Day = day, `New York` = pred, `New Jersey` = pred.1, Massachusetts = pred.2,
           Illinois = pred.3, California = pred.4)) %>%
  pivot_longer(2:6,
               names_to = "State")

high.capa.sum <- state.sum %>% filter(Province_State %in% high.capa)
state.pred.2$State <-  factor(state.pred.2$State, levels = high.capa )

# Plot for top 5
ggplot(data = high.capa.sum, mapping = aes(x = day, y = sum.confirmed)) +
  geom_point(color = "black", size = .5) +
  geom_line(data = state.pred.2, mapping = aes(x = Day, y = value, color = State)) +
  labs(title = "Confirmed vs. Predicted Cases, Top 10 States by Capacity",
       x = "Day", y = "Number of Cases", color = "State or Territory")

# Hospitalization
hosp.ca <- hospitalizations %>%
  filter(state == "CA")

# Remove 0 hospitalization days
hosp.ca <- hosp.ca %>%
 replace_na(list(hospitalizedCurrently = 0))
ggplot(data = hosp.ca, aes(x = date, y = hospitalizedCurrently)) +
  geom_line() +
  labs(x = "Day", y = "Number of Hospitalizations",
       title = "Number of Current California Hospitalizations")

# Make day variable for CA
hosp.ca$day <- round(as.numeric(difftime(as.Date(as.character(hosp.ca$date),
                                format = "%Y%m%d"), "2020-01-22",units = "days")))

# Run Logistic model
```

```r
gc.fit.ca <- SummarizeGrowth(hosp.ca$day, hosp.ca$hospitalizedCurrently)
# Retrieve parameters
k <- unlist(gc.fit.ca$vals['k'])
n0 <- unlist(gc.fit.ca$vals['n0'])
r <- unlist(gc.fit.ca$vals['r'])

# Get maximum number of hospital ICU beds
capacity.ca <- capacity %>%
  filter(state == "CA") %>%
  summarize(TotalICUBeds = sum(`Total ICU Beds`))

# Make prediction until day 160
ca.pred <- data.frame(day = 0:160) %>%
  mutate(pred = k/(1 + ((k - n0)/n0) * exp(-r * day)))

# Plot
ggplot(data = hosp.ca, mapping = aes(x = day, y = hospitalizedCurrently)) +
  geom_line(color = "black") + #geom_point(color = "red") +
  geom_line(data = ca.pred, mapping = aes(x = as.numeric(rownames(ca.pred)), y = pred),
            color = "blue") +
  geom_hline(yintercept=as.numeric(capacity.ca), linetype="dashed", color = "red", size = 1) +
  geom_text(aes(x = 25, y = as.numeric(capacity.ca),
                label = "CA's ICU Bed Capacity = 8105", vjust = 2), size = 3) +
  labs(x = "Day", y = "Number of Beds",
       title = "Current Total Hospitalizations in CA Over Time")
# Hospitalization
hosp.la <- hospitalizations %>%
  filter(state == "LA")

# Remove 0 hospitalization days
hosp.la <- hosp.la %>%
  replace_na(list(hospitalizedCurrently = 0))
ggplot(data = hosp.la, aes(x = date, y = hospitalizedCurrently)) +
  geom_line() +
  labs(x = "Date", y = "Number of Hospitalizations",
       title = "Number of Louisiana Hospitalizations over Time")

# Make day variable for LA
hosp.la$day <- round(as.numeric(difftime(as.Date(as.character(hosp.la$date),
                                format = "%Y%m%d"), "2020-01-22",units = "days")))

# Run Logistic model
gc.fit.la <- SummarizeGrowth(hosp.la$day, hosp.la$hospitalizedCurrently)
# Retrieve parameters
k <- unlist(gc.fit.la$vals['k'])
n0 <- unlist(gc.fit.la$vals['n0'])
r <- unlist(gc.fit.la$vals['r'])

# Get maximum number of ICU hospital beds
capacity.la <- capacity %>%
  filter(state == "LA") %>%
  summarize(TotalICUBeds = sum(`Total ICU Beds`))
capacity.la2 <- capacity %>%
```

```
  filter(state == "LA") %>%
  summarize(TotalHospBeds = sum(`Total Hospital Beds`))

# Make prediction until day 160
la.pred <- data.frame(day = 0:160) %>%
  mutate(pred = k/(1 + ((k - n0)/n0) * exp(-r * day)))

# Plot
ggplot(data = hosp.la, mapping = aes(x = day, y = hospitalizedCurrently)) +
  geom_line(color = "black") + #geom_point(color = "red") +
  geom_line(data = la.pred, mapping = aes(x = as.numeric(rownames(la.pred)), y = pred),
            color = "blue") +
  geom_hline(yintercept=as.numeric(capacity.la), linetype="dashed", color = "red", size = 1) +
  geom_text(aes(x = 25, y = as.numeric(capacity.la),
                label = "LA's ICU Bed Capacity = 1572", vjust = 2), size = 3) +
  labs(x = "Day", y = "Number of Hospitalizations",
       title = "Current Total Hospitalizations in Louisiana Over Time")
  #geom_hline(yintercept=as.numeric(capacity.la2), linetype="dashed", color = "red", size = 1) +
  #geom_text(aes(x = 25, y = as.numeric(capacity.la2),
  #              label = "LA's ICU Bed Capacity", vjust = 2), size = 3)
# High risk proportion

high.risk.by.HRR <- high.risk %>%
  group_by(state,region, HRR.name) %>%
  summarize(TotalPop = sum(population), TotalRiskPop = sum(high.risk.pop)) %>%
  mutate(PercentRisk = TotalRiskPop / TotalPop)
#Then join with capacity data. I think we need data on total number of hospitalizations not just confir
capacity$region <- tolower(capacity$region)
high.risk.by.HRR$region <- tolower(high.risk.by.HRR$region)

hospital.data <- full_join(high.risk.by.HRR, capacity, by = c("state", "region"))

hospital.data <- hospital.data %>% filter(!is.na(TotalRiskPop), !is.na(`Available Hospital Beds`)) %>%
  mutate(beds_ratio = TotalRiskPop/`Available Hospital Beds`) %>%
  arrange(beds_ratio)

#us.county <- map_data("county")
#us.cities <- map_data("us.cities")
HRR_boundary <- st_read("HRR_Bdry.shp")
HRR_boundary$HRRCITY <- tolower(HRR_boundary$HRRCITY)
hospital.data$HRR.name <- tolower(hospital.data$HRR.name)
us.hrr.map <- left_join(HRR_boundary, hospital.data, by = c("HRRCITY" = "HRR.name"))

ggplot(us.hrr.map) +
  geom_sf(size = 0.3, color = "white", aes(fill = beds_ratio)) +
  ggtitle("The Country's Worst Prepared Regions: \nAvailable Beds to High Risk Population Ratio") +
  scale_fill_gradient(low = "blue", high = "red") +
  labs(x = "Longitude", y = "Latitude", fill = "Beds Ratio")
```