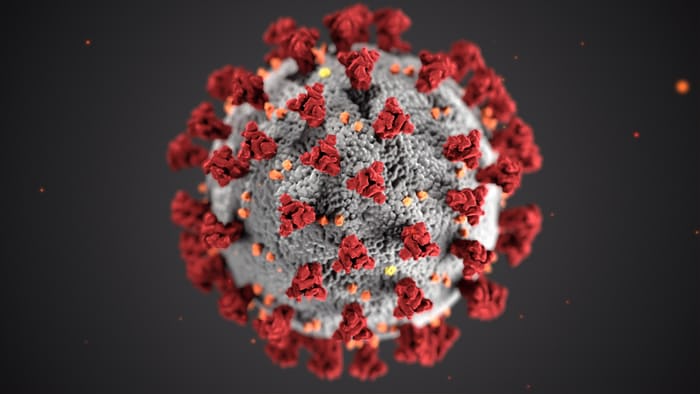
COVID-19: Current Climate and Deep Analysis



Kevin Pollard, Sean Dunn, Alexandra Kosmidis

IST 707 Project Report

12 June 2020

Table of Contents

[Introduction 1](#_Toc42808571)

[Motivation 1](#_Toc42808572)

[Problem Statement 1](#_Toc42808573)

[About the Data 1](#_Toc42808574)

[Data Prep 1](#_Toc42808575)

[EDA 1](#_Toc42808576)

[Covid Domestic 1](#_Toc42808577)

[COVID-19 Worldwide 1](#_Toc42808578)

[Twitter Sentiment 1](#_Toc42808579)

[Biomarker Modeling 1](#_Toc42808580)

[Modeling Overview 1](#_Toc42808581)

[Motivation 1](#_Toc42808582)

[Problem Statement 1](#_Toc42808583)

[About Model Data 1](#_Toc42808584)

[Data Loading 1](#_Toc42808585)

[View Loaded data 1](#_Toc42808586)

[Create Patient Master 1](#_Toc42808587)

[View Patient Master 1](#_Toc42808588)

[Aggregate Data 1](#_Toc42808589)

[Age and Gender Analysis 1](#_Toc42808590)

[Age Histogram 1](#_Toc42808591)

[Age Outliers 1](#_Toc42808592)

[Gender Analysis 1](#_Toc42808593)

[Correlation Analysis 1](#_Toc42808594)

[Kmeans Clustering 1](#_Toc42808595)

[Cluster Build 1](#_Toc42808596)

[Elbow Plot Result 1](#_Toc42808597)

[Principal Component Analysis 1](#_Toc42808598)

[Classification Models 1](#_Toc42808599)

[Data Splitting 1](#_Toc42808600)

[Model Controls 1](#_Toc42808601)

[Model Tuning 1](#_Toc42808602)

[Model Build 1](#_Toc42808603)

[Results 1](#_Toc42808604)

[Model Fit Results 1](#_Toc42808605)

[Model Prediction Results 1](#_Toc42808606)

[Conclusions 1](#_Toc42808607)

[References 1](#_Toc42808608)

### Introduction

Pandemics have plagued mankind since antiquity, with one of the first being the Antonine Plague. Also known as the Plague of Galen, the ancient pandemic lasted from 165 to 180 AD, and impacted Southern Europe, particularly the Roman Empire. Later significant plagues recorded in history include the Plague of Justinian from 541 to 750 AD, which was the first recorded outbreak of bubonic plague, killing 10,000 people a day at its peak. The famous Black Death from 1331 to 1353 impacted Asia and Europe, killing an estimated 75 to 200 million worldwide, with multiple outbreaks and waves. The 1918 flu pandemic, otherwise known as the Spanish Flu, killed 20 to 100 million worldwide, impacting multiple continents and causing deaths in young, healthy individuals, breaking historical trends recorded of other pandemics. The impacts of pandemics are undoubtedly drastic and perplexing, especially when considering the rates at which infection and death occur.

Humanity is no stranger to pandemics, but the outbreak of a new strain of viral infections always proves to be devastating for healthcare workers, researchers, and everyday citizens all over the world. While coronaviruses are not uncommon, they are amongst the most difficult to treat, causing more severe disease contractions such as Middle East Respiratory Syndrome (MERS) and Severe Acute Respiratory Syndrome (SARS). With the recent outbreak of a new coronavirus, COVID-19, new challenges have been posed in attempts to eradicate the spread and contraction of the illness. Pressure is placed on vaccine development, while the economic, psychological, and social implications will be felt for generations to come.

There have been over 5.5 million confirmed cases worldwide of COVID-19, and nearly 350,000 deaths, having been declared a pandemic on March 11, 2020. The response of government and health officials has been proven to be crucial in the rate at which transmission spreads, as well as how quickly citizens recover and adhere to protective measures and procedures put into place. While there is still much to learn and discover about the disease and its long-term impact, there are many useful patterns to observe through the analysis of official collected case data all over the world.

### Motivation

Efforts to completely contain COVID-19 have failed, and the disease continues to spread. The rapid growth rate of COVID-19 infections continues to fill hospitals in certain areas to capacity, which overwhelms the healthcare system. While it’s clear COVID-19 will continue to infect many more, our analysis aims to learn more about what clusters have formed worldwide, investigate why they formed, and highlight parallels among the clustering to the US. Our entire team is interested in this analysis as we each one of us has a stake in maintaining health and helping decision-makers prepare and respond to surges in COVID-19.

### Problem Statement

Our analysis involved a clustering and classification problem, which predicts whether given attributes in a location will likely develop into one of three disease cluster classes: a mild cluster, moderate cluster, severe cluster. The aspirational outcome would be to enable decision-makers, such as lawmakers, government officials, and citizens, lead-time to take action to minimize the extent of the outbreak and to prepare for a surge response.

### About the Data

Data has been collected through multiple sources, in order to address different analysis questions and angles.

Johns Hopkins Data

The Johns Hopkins dataset is synced hourly from the Johns Hopkins Center for Systems Science and Engineering. It contains a current count of confirmed global cases, as well as deaths, and death rate. The rows are per each country affected. There are 5 columns in total, with 147 rows as of May 26, 2020. A secondary set used includes the Johns Hopkins COVID-19 Time Series Data, which includes latitude and longitude coordinates, as well as date intervals which confirmed cases are observed. This also includes identically structured sets for both deaths and recovered numbers.

New York Times Data

The New York Times dataset is a released series of date files with cumulative counts of coronavirus present in the United States, broken down by state and county level. This has been collected since the first reported coronavirus case on January 21, 2020. There are three files present, US cumulative Data, US state-level data, and US county-level data. The US data has three columns labeled as date, cases, and deaths. The US state-level data has five columns labeled as date, state, FIPS, cases, and deaths. The US county-level data has six columns labeled as date, county, state, FIPS, cases, and deaths. The live files present have further broken-down case and death fields, identified as cases, deaths, confirmed\_cases, confirmed\_deaths, probable\_cases, and probable\_deaths.

European Center for Disease Control

The European Center for Disease Control dataset is similar to the previous two sources, with 11 columns identified as dateRep, day, month, year, cases, deaths, country, geographical ID, country territory code, population data, and continentExp. This gives a further picture of cases relative to population numbers present at the time of present cases and deaths, as well as separate columns for day and month, which prove useful in prospective data analyses and model training. Both the ECDC and Johns Hopkins datasets provide a collective overview of worldwide impact of the COVID-19 impact, particularly when looking at recovery rates and predictive methods.

#### Data Loading

***Johns Hopkins***  Data time series worldwide data update 9AM daily

<https://data.world/shad/covid-19-time-series-data> <https://github.com/CSSEGISandData/COVID-19>

***EDC******Data***

<https://www.ecdc.europa.eu/en/publications-data/download-todays-data-geographic-distribution-covid-19-cases-worldwide>

***NY Times******Data***

<https://github.com/nytimes/covid-19-data/blob/master/README.md>

A tibble is created to see total cases and deaths at current time:

## # A tibble: 1 x 2

## cases deaths

## <int> <int>

## 1 7069278 405587

### Data Prep

Collectively, data was checked for tidiness and completeness across all three dataset sources. This included checking for completeness, missing values, structure of data, and duplicates present.

#### Check Complete Cases

## a

## 1 Complete Cases? FALSE Incomplete Count: 2328

## a

## 1 Complete Cases? TRUE Incomplete Count: 0

## a

## 1 Complete Cases? TRUE Incomplete Count: 0

## a

## 1 Complete Cases? TRUE Incomplete Count: 0

#### Check NA Status

| **columns** | **percent\_missing** |
| --- | --- |
| date | 0.00000000 |
| county | 0.00000000 |
| state | 0.00000000 |
| fips | 0.01064827 |
| cases | 0.00000000 |
| deaths | 0.00000000 |
| **columns** | **percent\_missing** |
| date | 0 |
| state | 0 |
| fips | 0 |
| cases | 0 |
| deaths | 0 |

| **columns** | **percent\_missing** |
| --- | --- |
| date | 0 |
| cases | 0 |
| deaths | 0 |
| **columns** | **percent\_missing** |
| date | 0 |
| cases | 0 |
| deaths | 0 |
| country | 0 |
| id | 0 |

After evaluation of the dataset, there was less than .01 percent missing in the FIPS category for the first dataset. Given the fact that geospatial and map-related plots will not be heavily relied on during this particular analysis, this is ignored during the cleaning and prep process. The rest of the datasets show 0 percent missing in all of the subsequent related columns and the attached occurrences.

#### Check Duplicates

After evaluation of the dataset, there were no duplicate data values or inputs found. Below is a table generated with the duplicate counts present in the datasets.

| **d** |
| --- |
| Duplicate Count: 0 |
| **d** |
| Duplicate Count: 0 |

| **d** |
| --- |
| Duplicate Count: 0 |
| **d** |
| Duplicate Count: 0 |

#### Data Structure

Data structure shows multiple data categories related to their related occurrences and columns as present in data frames. The first data set consists of six different variables, with date, county, and state being identified as factors, and the FIPS, cases, and deaths being shown as integers. This identification is consistent amongst all the data, with the exception being that the date in the last data set is not identified as a factor, but as “date” with a specific format. This is to be noted if conversion is needed during analytical processes.

## 'data.frame': 218627 obs. of 6 variables:

## $ date : Factor w/ 140 levels "2020-01-21","2020-01-22",..: 1 2 3 4 4 5 5 5 6 6 ...

## $ county: Factor w/ 1774 levels "Abbeville","Acadia",..: 1468 1468 1468 379 1468 1178 379 1468 979 942 ...

## $ state : Factor w/ 55 levels "Alabama","Alaska",..: 52 52 52 15 52 5 15 52 3 5 ...

## $ fips : int 53061 53061 53061 17031 53061 6059 17031 53061 4013 6037 ...

## $ cases : int 1 1 1 1 1 1 1 1 1 1 ...

## $ deaths: int 0 0 0 0 0 0 0 0 0 0 ...

## 'data.frame': 5404 obs. of 5 variables:

## $ date : Factor w/ 140 levels "2020-01-21","2020-01-22",..: 1 2 3 4 4 5 5 5 6 6 ...

## $ state : Factor w/ 55 levels "Alabama","Alaska",..: 52 52 52 15 52 5 15 52 3 5 ...

## $ fips : int 53 53 53 17 53 6 17 53 4 6 ...

## $ cases : int 1 1 1 1 1 1 1 1 1 2 ...

## $ deaths: int 0 0 0 0 0 0 0 0 0 0 ...

## 'data.frame': 140 obs. of 3 variables:

## $ date : Factor w/ 140 levels "2020-01-21","2020-01-22",..: 1 2 3 4 5 6 7 8 9 10 ...

## $ cases : int 1 1 1 2 3 5 5 5 5 6 ...

## $ deaths: int 0 0 0 0 0 0 0 0 0 0 ...

## 'data.frame': 22383 obs. of 5 variables:

## $ date : Date, format: "2020-06-09" "2020-06-08" ...

## $ cases : int 575 791 582 915 787 758 759 545 680 866 ...

## $ deaths : int 12 30 18 9 6 24 5 8 8 3 ...

## $ country: Factor w/ 210 levels "Afghanistan",..: 1 1 1 1 1 1 1 1 1 1 ...

## $ id : Factor w/ 207 levels "","ABW","AFG",..: 3 3 3 3 3 3 3 3 3 3 ...

## [1] ""

### EDA

EDA includes an understanding of statistical summaries, rankings, trends, and map visuals to provide an overall picture of the data for further analysis. This is understood on both a domestic (USA) and worldwide level to provide a micro and macro scope for analytical foundations.

#### Statistical Summary

## [1] 218627 6

## [1] "fips" "cases" "deaths"

## [1] "==============================================================="

## [1] "---------------------------------------------------------------"

## [1] "fips"

## [1] "---------------------------------------------------------------"

## [1] "mean: 30072.9616641778"

## [1] "median: 29055"

## [1] "min: 1001"

## [1] "max: 56045"

## [1] "range: 55044"

## [1] "sd: 15350.0274083435"

## [1] "quantile: 1001" "quantile: 18079" "quantile: 29055" "quantile: 45041"

## [5] "quantile: 56045"

## [1] "IQR: 26962"

## [1] "---------------------------------------------------------------"

## [1] "cases"

## [1] "---------------------------------------------------------------"

## [1] "mean: 371.947678008663"

## [1] "median: 20"

## [1] "min: 0"

## [1] "max: 212122"

## [1] "range: 212122"

## [1] "sd: 3560.53234758282"

## [1] "quantile: 0" "quantile: 5" "quantile: 20"

## [4] "quantile: 94" "quantile: 212122"

## [1] "IQR: 89"

## [1] "---------------------------------------------------------------"

## [1] "deaths"

## [1] "---------------------------------------------------------------"

## [1] "mean: 20.9091877947372"

## [1] "median: 0"

## [1] "min: 0"

## [1] "max: 21356"

## [1] "range: 21356"

## [1] "sd: 321.345923502159"

## [1] "quantile: 0" "quantile: 0" "quantile: 0" "quantile: 3"

## [5] "quantile: 21356"

## [1] "IQR: 3"

## [1] 5404 5

## [1] "fips" "cases" "deaths"

## [1] "==============================================================="

## [1] "---------------------------------------------------------------"

## [1] "fips"

## [1] "---------------------------------------------------------------"

## [1] "mean: 31.6260177646188"

## [1] "median: 31"

## [1] "min: 1"

## [1] "max: 78"

## [1] "range: 77"

## [1] "sd: 18.4859705992254"

## [1] "quantile: 1" "quantile: 17" "quantile: 31" "quantile: 46"

## [5] "quantile: 78"

## [1] "IQR: 29"

## [1] "---------------------------------------------------------------"

## [1] "cases"

## [1] "---------------------------------------------------------------"

## [1] "mean: 15047.618430792"

## [1] "median: 2378"

## [1] "min: 1"

## [1] "max: 383591"

## [1] "range: 383590"

## [1] "sd: 39855.1197068709"

## [1] "quantile: 1" "quantile: 216" "quantile: 2378"

## [4] "quantile: 12284.75" "quantile: 383591"

## [1] "IQR: 12068.75"

## [1] "---------------------------------------------------------------"

## [1] "deaths"

## [1] "---------------------------------------------------------------"

## [1] "mean: 846.285344189489"

## [1] "median: 66"

## [1] "min: 0"

## [1] "max: 30239"

## [1] "range: 30239"

## [1] "sd: 2920.49567536409"

## [1] "quantile: 0" "quantile: 3" "quantile: 66"

## [4] "quantile: 449.25" "quantile: 30239"

## [1] "IQR: 446.25"

## [1] 140 3

## [1] "cases" "deaths"

## [1] "==============================================================="

## [1] "---------------------------------------------------------------"

## [1] "cases"

## [1] "---------------------------------------------------------------"

## [1] "mean: 580838.071428571"

## [1] "median: 176148"

## [1] "min: 1"

## [1] "max: 1970613"

## [1] "range: 1970612"

## [1] "sd: 680581.819202428"

## [1] "quantile: 1" "quantile: 44.5" "quantile: 176148"

## [4] "quantile: 1192891" "quantile: 1970613"

## [1] "IQR: 1192846.5"

## [1] "---------------------------------------------------------------"

## [1] "deaths"

## [1] "---------------------------------------------------------------"

## [1] "mean: 32666.6142857143"

## [1] "median: 3833"

## [1] "min: 0"

## [1] "max: 110966"

## [1] "range: 110966"

## [1] "sd: 40298.0976344798"

## [1] "quantile: 0" "quantile: 0" "quantile: 3833"

## [4] "quantile: 69463.5" "quantile: 110966"

## [1] "IQR: 69463.5"

## [1] 22383 5

## [1] "cases" "deaths"

## [1] "==============================================================="

## [1] "---------------------------------------------------------------"

## [1] "cases"

## [1] "---------------------------------------------------------------"

## [1] "mean: 315.832462136443"

## [1] "median: 3"

## [1] "min: -2461"

## [1] "max: 48529"

## [1] "range: 50990"

## [1] "sd: 1838.03575600202"

## [1] "quantile: -2461" "quantile: 0" "quantile: 3" "quantile: 54"

## [5] "quantile: 48529"

## [1] "IQR: 54"

## [1] "---------------------------------------------------------------"

## [1] "deaths"

## [1] "---------------------------------------------------------------"

## [1] "mean: 18.1203145244158"

## [1] "median: 0"

## [1] "min: -1918"

## [1] "max: 4928"

## [1] "range: 6846"

## [1] "sd: 122.37288689534"

## [1] "quantile: -1918" "quantile: 0" "quantile: 0" "quantile: 1"

## [5] "quantile: 4928"

## [1] "IQR: 1"

### Covid Domestic

#### Top 10 Ranking Cities

For domestic (USA) cases, a table is created to show the top-ranking states for number of COVID-19 cases. The first ranked state is New York, followed by Illinois, California, and so-on.

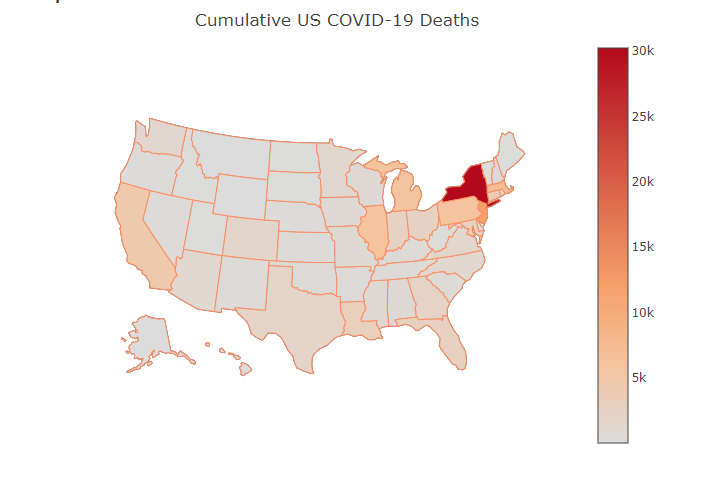
| **state** | **Ranking** | **Ranked.state** |
| --- | --- | --- |
| New York | 01 | 01 -> New York |
| Illinois | 02 | 02 -> Illinois |
| California | 03 | 03 -> California |
| Pennsylvania | 04 | 04 -> Pennsylvania |
| Massachusetts | 05 | 05 -> Massachusetts |
| Michigan | 06 | 06 -> Michigan |
| Florida | 07 | 07 -> Florida |
| New Jersey | 08 | 08 -> New Jersey |
| Maryland | 09 | 09 -> Maryland |
| Connecticut | 10 | 10 -> Connecticut |

#### CRF USA Top 10 States

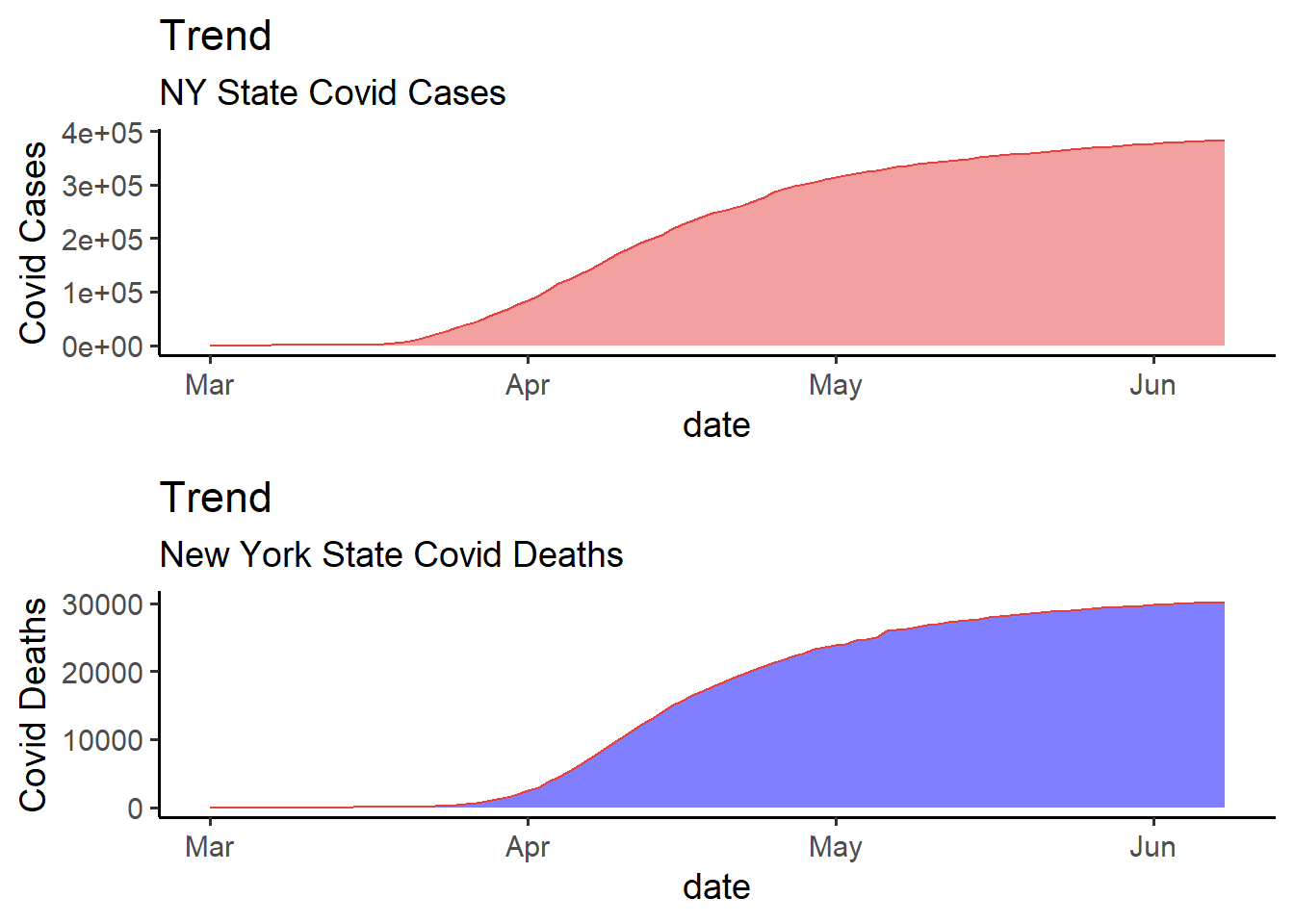
Going one step further for domestic (USA) preliminary visualization and analysis, the top-ranking states for case fatality rate (CFR) is generated as table. The top-ranking state is Connecticut, followed by Michigan, Virgin Islands, and so-on. Note that this data set also includes USA territories.

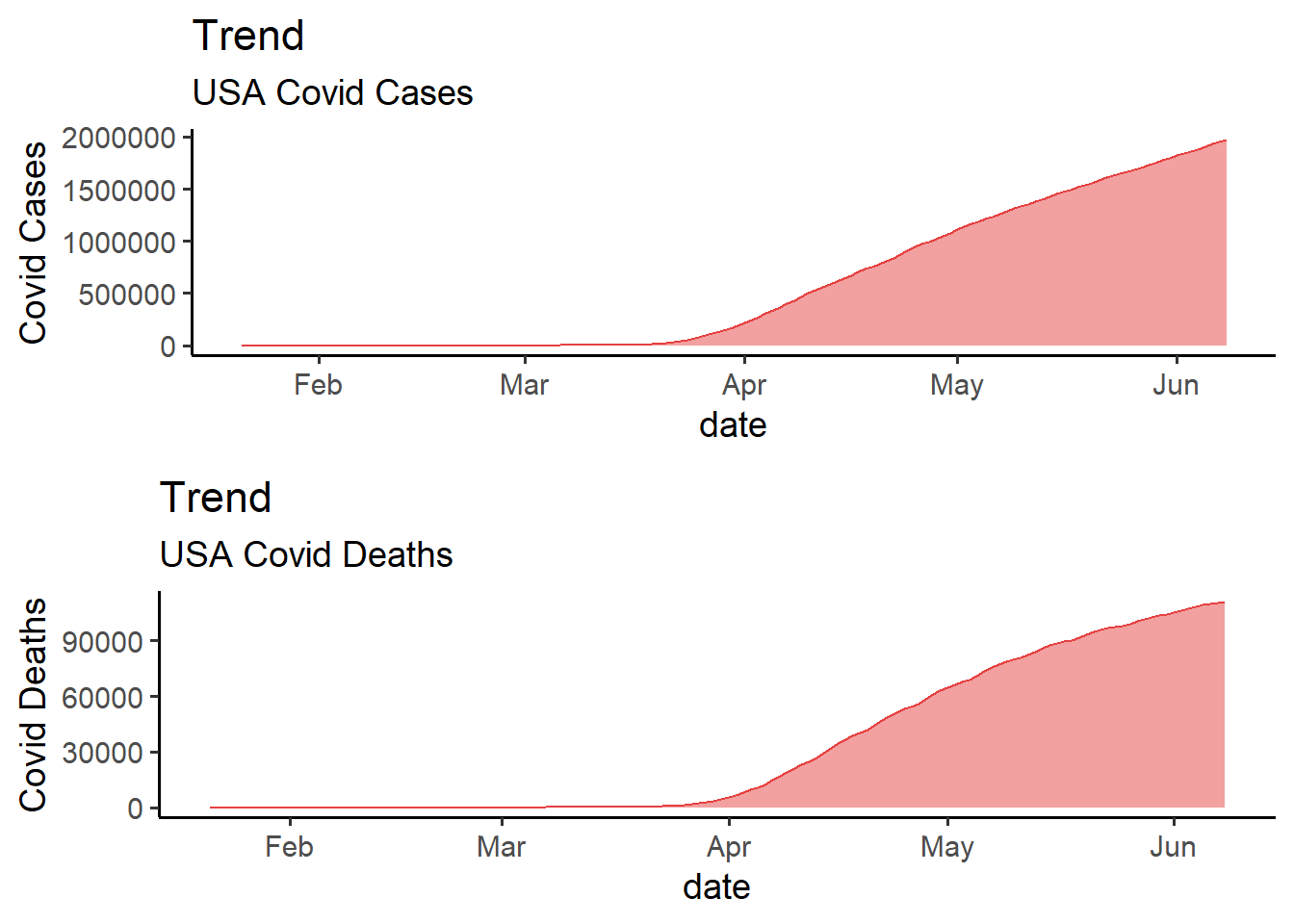
| **state** | **cases** | **deaths** | **mortality** | **Std** |
| --- | --- | --- | --- | --- |
| Connecticut | 44092 | 4084 | 0.09262451 | 0.02000629 |
| Michigan | 64911 | 5916 | 0.09114018 | 0.01803117 |
| Virgin Islands | 71 | 6 | 0.08450704 | 0.09582071 |
| New York | 383591 | 30239 | 0.07883136 | 0.01082809 |
| Pennsylvania | 80432 | 6007 | 0.07468421 | 0.01560997 |
| New Jersey | 164497 | 12214 | 0.07425059 | 0.01301839 |
| Northern Mariana Islands | 28 | 2 | 0.07142857 | 0.11195774 |
| Massachusetts | 103626 | 7353 | 0.07095710 | 0.01431030 |
| Louisiana | 43163 | 2944 | 0.06820657 | 0.01749021 |
| Ohio | 38837 | 2404 | 0.06189973 | 0.01716555 |

#### US Map Deaths



A map is generated to understand the density and prevalence of COVID-19 cases in the United States as a whole. New York state remains the top in the number of cases present. Given that New York is the top-ranking state for number of COVID-19 cases, a trend graph is produced below as an example to show the spread of COVID-19 and number of cases from March to June 2020. This trend is then compared directly with the calculated COVID-19 deaths as recorded in New York state from March to June 2020.





### COVID-19 Worldwide

#### Top Eight Countries

The top eight countries worldwide are generated into a table by number of cases. The top-ranking country by number of COVID-19 cases is noted as the USA, followed by Brazil, Russia, and so-on.

| **country** | **cases** |
| --- | --- |
| US | 1960897 |
| Brazil | 707412 |
| Russia | 476043 |
| United Kingdom | 288834 |
| India | 265928 |
| Spain | 241717 |
| Italy | 235278 |
| Peru | 199696 |

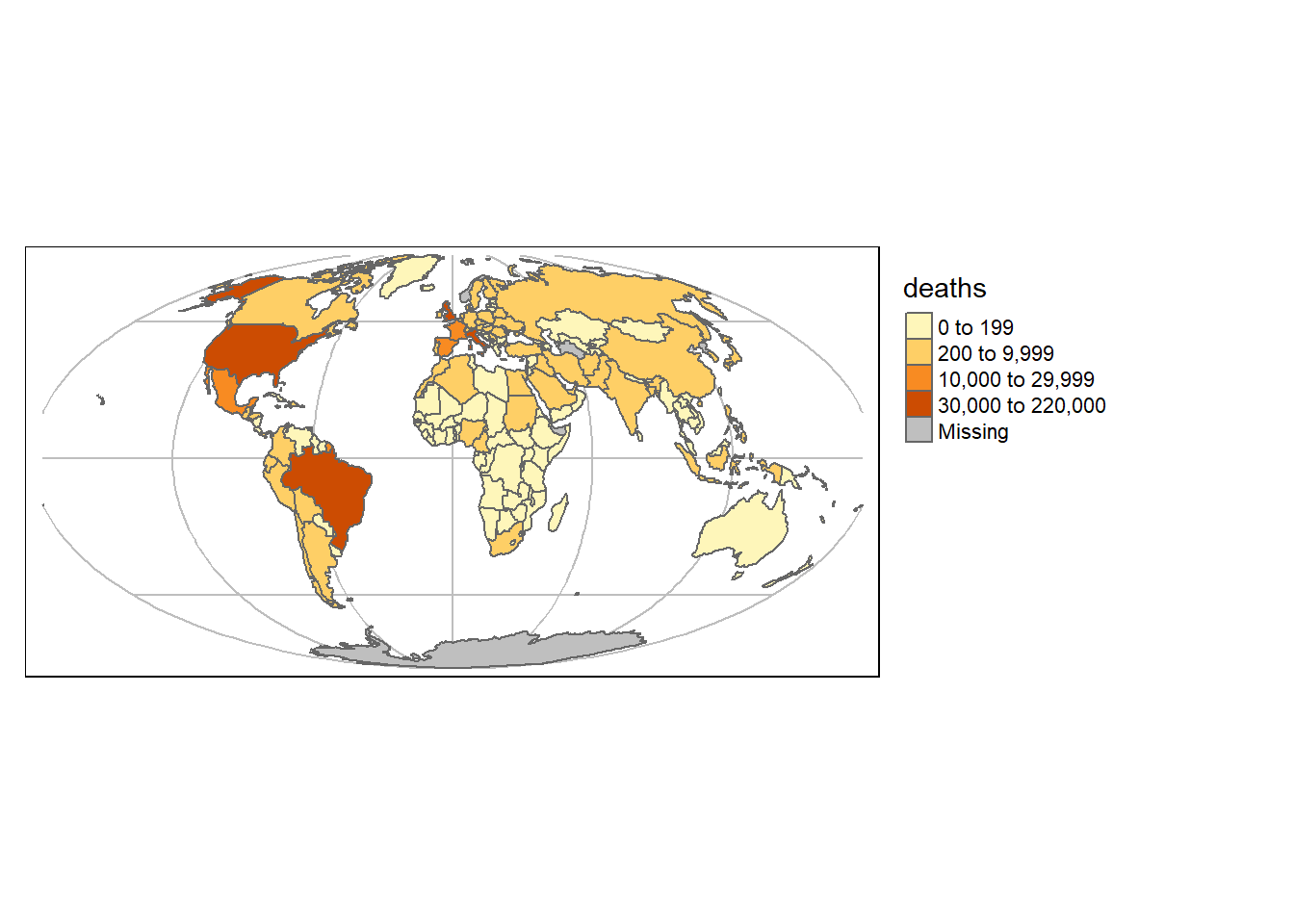
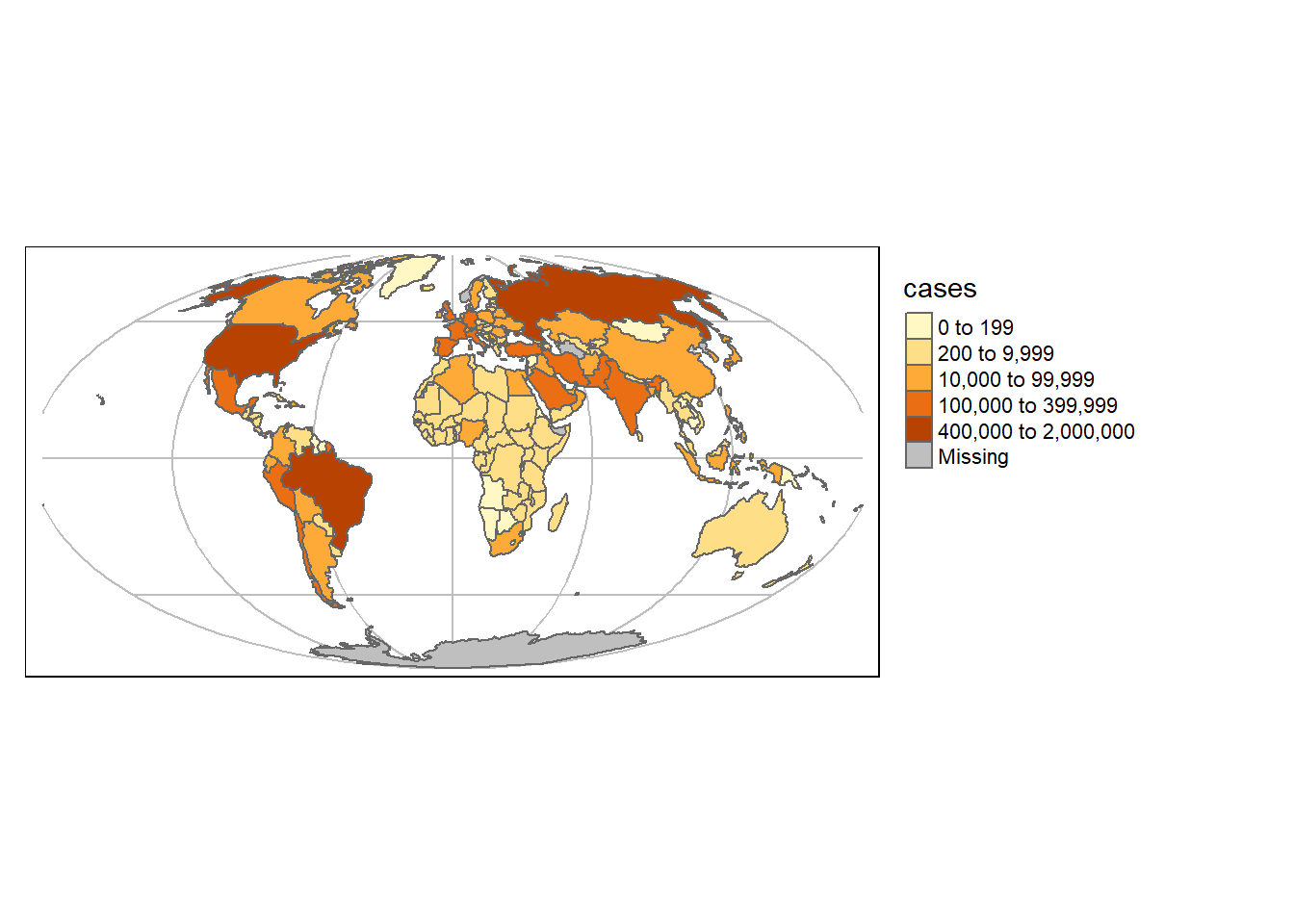
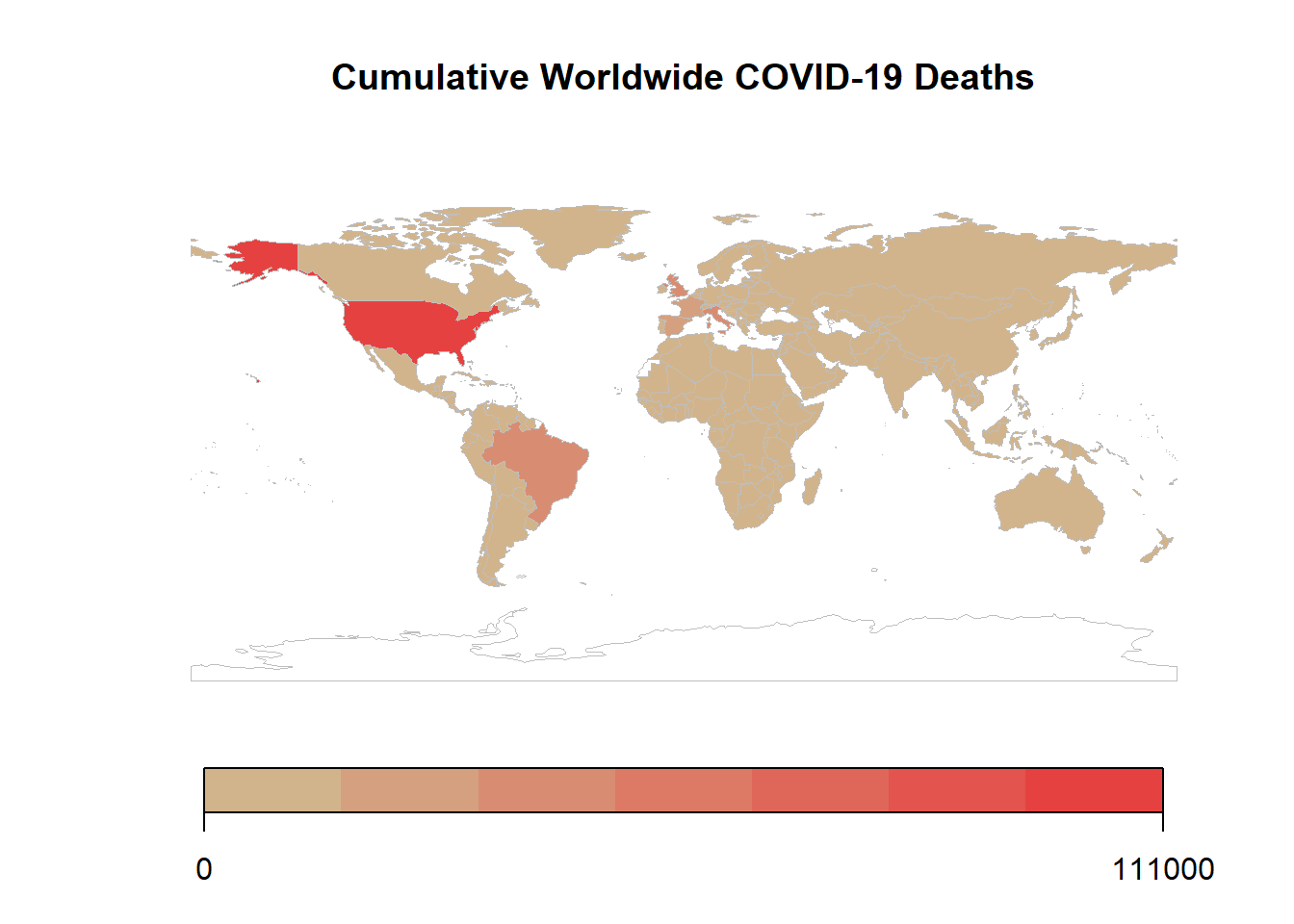
Again, the top eight countries worldwide are generated into a table by number of deaths by COVID-19. The top-ranking country by number of COVID-19 deaths is noted as the USA, followed by the United Kingdom, Brazil, Italy, and so-on.

| **country** | **death** |
| --- | --- |
| US | 110990 |
| United Kingdom | 40680 |
| Brazil | 37134 |
| Italy | 33964 |
| France | 29212 |
| Spain | 27136 |
| Mexico | 14053 |
| Belgium | 9606 |

#### Global Maps

#### Maps are generated for worldwide COVID-19 cases much like previously generated for cases in the USA. This is further broken down by showing the highest number of deaths, as shown by color-coded density.

#### C:\Users\knjpo\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\E4E43DA.tmp



#### Global Cases and Deaths

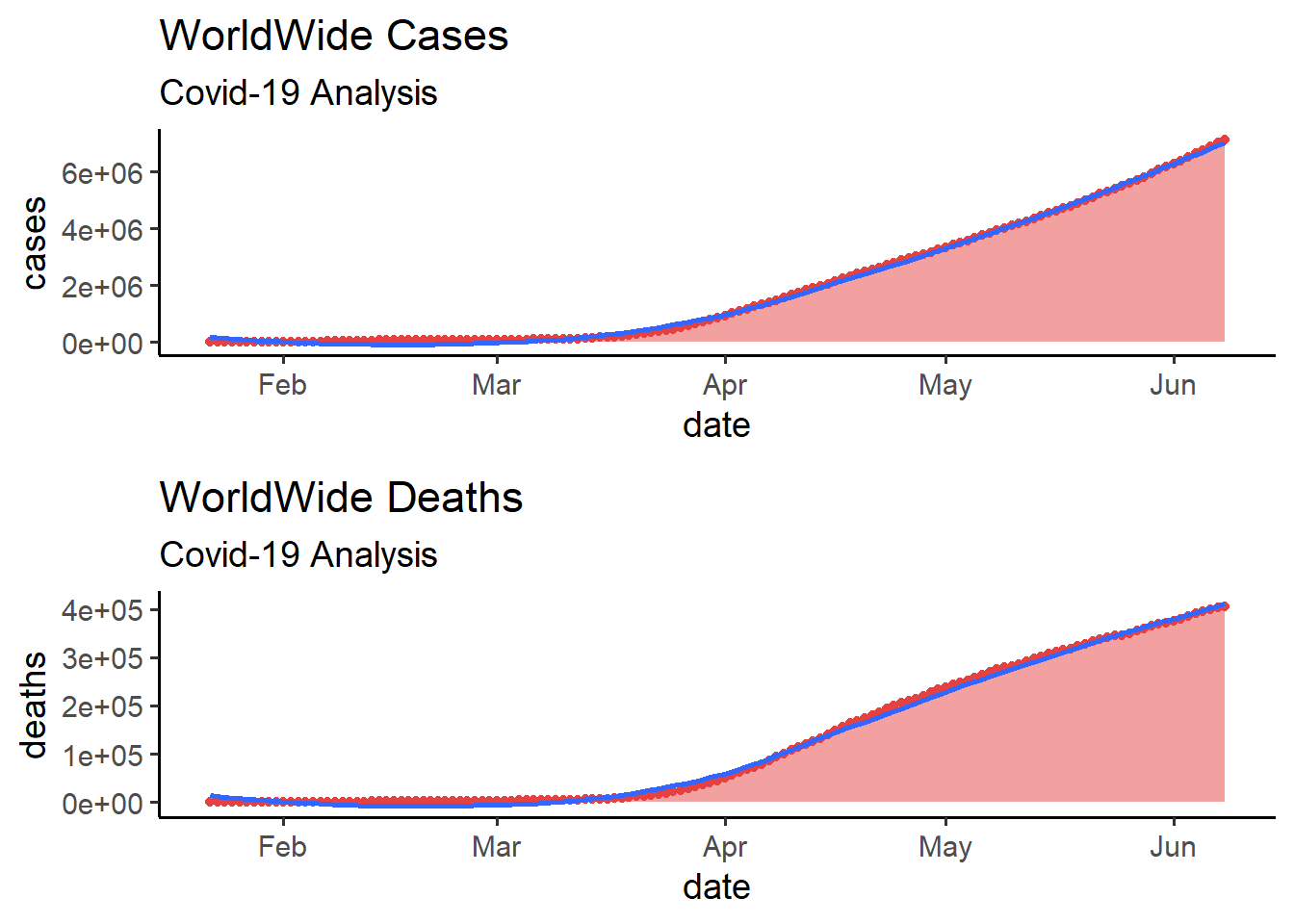
After understanding the prevalence of cases by country, it is important to understand the timeline of reported cases worldwide to further analyze questions or problems related to transmission rates in communities. This is shown in a generated table documenting cases from June 3, 2020 to June 8, 2020.

| **date** | **cases** |
| --- | --- |
| 2020-06-08 | 7118471 |
| 2020-06-07 | 7010349 |
| 2020-06-06 | 6891213 |
| 2020-06-05 | 6764918 |
| 2020-06-04 | 6632985 |
| 2020-06-03 | 6508635 |

Going one step further, deaths are added into the table themselves to further understand the timeline of cases that result in deaths. It is noted that as of June 8, 2020, there were over 400,000 deaths worldwide. Again, this is shown in a generated table documenting cases from June 3, 2020 to June 8, 2020.

| **date** | **cases** | **deaths** |
| --- | --- | --- |
| 2020-06-08 | 3798 | 406522 |
| 2020-06-07 | 3021 | 402724 |
| 2020-06-06 | 3837 | 399703 |
| 2020-06-05 | 4744 | 395866 |
| 2020-06-04 | 5189 | 391122 |
| 2020-06-03 | 5684 | 385933 |
|  |  |  |

Much like the New York state trends graphed from the domestic (USA) data, the same is completed for worldwide cases of COVID-19 followed by a trend analysis of worldwide deaths from COVID-19. These are compared to one another, encompassing cases and deaths from February 2020 to June 2020.



#### Global cumulative COVID-19

#### A tibble is developed to look at the overall confirmed, dead, and recovered cases worldwide, as well as a calculated case fatality rate based on recorded date.

## # A tibble: 2 x 8

## variable date confirmed death recovered cfr\_today cfr\_cumulative label

## <chr> <date> <dbl> <dbl> <dbl> <dbl> <dbl> <chr>

## 1 10000 2020-02-01 2111 46 62 0.0218 0.0215 " 10,0~

## 2 100000 2020-03-06 3918 112 2069 0.0286 0.0340 "100,0~

#### Plot CFR trends

#### To visualization case fatality rate further, plotly is used to understand the observed case fatality rates for worldwide COVID-19 cases. This is observed from January to June 2020. COVID-19 is seen with a declining fatality rate starting at the end of May to present day in June.

#### A close up of a map Description automatically generated

#### Country Specific

#### Country CFR trends

The top eight countries as shown in previous tables are observed for their specific case fatality rates using plotly, once again. The countries noted are Belgium, Brazil, France, Italy, Mexico, Spain, United Kingdom, and the USA. This is important to observe and analyze in regard to specific policies, documentation methods, laws, and orders put into place by governments and administrations unique to each of the leading countries of COVID-19 cases. This can also allude to an understanding of citizen response and adherence to procedures.

![A close up of a map

Description automatically generated](data:image/jpeg;base64,/9j/4AAQSkZJRgABAQEAYABgAAD/4RD0RXhpZgAATU0AKgAAAAgABAE7AAIAAAAOAAAISodpAAQAAAABAAAIWJydAAEAAAAcAAAQ0OocAAcAAAgMAAAAPgAAAAAc6gAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAEtldmluIFBvbGxhcmQAAAWQAwACAAAAFAAAEKaQBAACAAAAFAAAELqSkQACAAAAAzk1AACSkgACAAAAAzk1AADqHAAHAAAIDAAACJoAAAAAHOoAAAAIAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAyMDIwOjA2OjEwIDIxOjUyOjA2ADIwMjA6MDY6MTAgMjE6NTI6MDYAAABLAGUAdgBpAG4AIABQAG8AbABsAGEAcgBkAAAA/+ELIGh0dHA6Ly9ucy5hZG9iZS5jb20veGFwLzEuMC8APD94cGFja2V0IGJlZ2luPSfvu78nIGlkPSdXNU0wTXBDZWhpSHpyZVN6TlRjemtjOWQnPz4NCjx4OnhtcG1ldGEgeG1sbnM6eD0iYWRvYmU6bnM6bWV0YS8iPjxyZGY6UkRGIHhtbG5zOnJkZj0iaHR0cDovL3d3dy53My5vcmcvMTk5OS8wMi8yMi1yZGYtc3ludGF4LW5zIyI+PHJkZjpEZXNjcmlwdGlvbiByZGY6YWJvdXQ9InV1aWQ6ZmFmNWJkZDUtYmEzZC0xMWRhLWFkMzEtZDMzZDc1MTgyZjFiIiB4bWxuczpkYz0iaHR0cDovL3B1cmwub3JnL2RjL2VsZW1lbnRzLzEuMS8iLz48cmRmOkRlc2NyaXB0aW9uIHJkZjphYm91dD0idXVpZDpmYWY1YmRkNS1iYTNkLTExZGEtYWQzMS1kMzNkNzUxODJmMWIiIHhtbG5zOnhtcD0iaHR0cDovL25zLmFkb2JlLmNvbS94YXAvMS4wLyI+PHhtcDpDcmVhdGVEYXRlPjIwMjAtMDYtMTBUMjE6NTI6MDYuOTUwPC94bXA6Q3JlYXRlRGF0ZT48L3JkZjpEZXNjcmlwdGlvbj48cmRmOkRlc2NyaXB0aW9uIHJkZjphYm91dD0idXVpZDpmYWY1YmRkNS1iYTNkLTExZGEtYWQzMS1kMzNkNzUxODJmMWIiIHhtbG5zOmRjPSJodHRwOi8vcHVybC5vcmcvZGMvZWxlbWVudHMvMS4xLyI+PGRjOmNyZWF0b3I+PHJkZjpTZXEgeG1sbnM6cmRmPSJodHRwOi8vd3d3LnczLm9yZy8xOTk5LzAyLzIyLXJkZi1zeW50YXgtbnMjIj48cmRmOmxpPktldmluIFBvbGxhcmQ8L3JkZjpsaT48L3JkZjpTZXE+DQoJCQk8L2RjOmNyZWF0b3I+PC9yZGY6RGVzY3JpcHRpb24+PC9yZGY6UkRGPjwveDp4bXBtZXRhPg0KICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICA8P3hwYWNrZXQgZW5kPSd3Jz8+/9sAQwAHBQUGBQQHBgUGCAcHCAoRCwoJCQoVDxAMERgVGhkYFRgXGx4nIRsdJR0XGCIuIiUoKSssKxogLzMvKjInKisq/9sAQwEHCAgKCQoUCwsUKhwYHCoqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioq/8AAEQgB9QLOAwEiAAIRAQMRAf/EAB8AAAEFAQEBAQEBAAAAAAAAAAABAgMEBQYHCAkKC//EALUQAAIBAwMCBAMFBQQEAAABfQECAwAEEQUSITFBBhNRYQcicRQygZGhCCNCscEVUtHwJDNicoIJChYXGBkaJSYnKCkqNDU2Nzg5OkNERUZHSElKU1RVVldYWVpjZGVmZ2hpanN0dXZ3eHl6g4SFhoeIiYqSk5SVlpeYmZqio6Slpqeoqaqys7S1tre4ubrCw8TFxsfIycrS09TV1tfY2drh4uPk5ebn6Onq8fLz9PX29/j5+v/EAB8BAAMBAQEBAQEBAQEAAAAAAAABAgMEBQYHCAkKC//EALURAAIBAgQEAwQHBQQEAAECdwABAgMRBAUhMQYSQVEHYXETIjKBCBRCkaGxwQkjM1LwFWJy0QoWJDThJfEXGBkaJicoKSo1Njc4OTpDREVGR0hJSlNUVVZXWFlaY2RlZmdoaWpzdHV2d3h5eoKDhIWGh4iJipKTlJWWl5iZmqKjpKWmp6ipqrKztLW2t7i5usLDxMXGx8jJytLT1NXW19jZ2uLj5OXm5+jp6vLz9PX29/j5+v/aAAwDAQACEQMRAD8A+kaKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAorgfiMJNW8QeFvC013cWmm6xcTm9a3lMbTLFFuEO8cgMTzjkhTWV4n8L6b8LPhzrl74FFzpbXCRQkG6kljg3yhDKodjtYBycg9h6UtRnqdFeQa74G0f4bLo/iHwlLe2+pSapaW108l7LKNRSWQI4kVmIJwSwIAwRXU+CCP+E18ejIyNXiJH/bpDTWvy/wCB/mTe1vP/AIP+R21FeDXMH2r4IvAJpITL4vKCWJsOmdSI3KexHUV0V34O0rwF8RPCl94bN5DPq97Laak015LN9sXyHcF97H5gyA5GKUXdJ97fik/1G9G12v8Ag3/keg6F4jtPEE2qx2ccyHS757GbzVA3SKqsSuCcrhh1wevFa1eMfDL4X+EE8UeIdTGj4vdH8QyxWMn2mb9yipGyjG/DcsT8wPWp/hz8PtB1/TpNZ1yO6vri1129ltIpLuRYbZkunwVjUhSSRk7gc/TimteXzSf5f5jlo32vb8/8j2CivFG1q78NeGdZ+G9lKV1gX66do5J+b7Lc5ZJB7Rp5gz22CvX9I0y30XRbPTLFNlvZwJBGP9lRgfyoWqv/AF3/AA0/pCejt/X9PUuUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUVz/AI+uZ7P4beJbqzmkguIdJupIpYnKvGwhYhlI5BBGQRQB0FFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAY3ijwppfi/S1sdXSXEUgmgngkMctvIOjo45UisrRPhzpelafqFpf6hq/iBdRh8i4fW75rlmj5+QdAByegz7111FKyA4bRPhNomjaxbX8mpa3qosW32FrqeoNPBZnGAY0OMYHAznFTa58MNI1zxJLrP9o61ps1yqpew6bqDW8V6FGAJQvJwOOCOK7OinuG2xx9r8MtDsvB8Hhq1lvI9Pt9QW/jxIu9XWbzguduNu7jGM4755rb1fw7aazqekX11JMkukXJuYBGwCsxjZMNkHIwx6Y5xWrRR/X9fcByJ+HOnR+Nn8S2Oq61YTTTLPc2VrfFLW6kAA3SR4+bgDuK2fDnh208MaZJY2Ek0kUlzNckzMC26WQuw4A4yxx7etatFC0Vl/X9WDd3ODHh5dZ+Nz+ILnTHhi0PTltbe6kiK/aJpSWYqTwwRTjI7uw7V3lFFHRL+u4PV3/rsFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFc38R/wDklniv/sC3n/oh66Sub+I//JLPFf8A2Bbz/wBEPQB0lFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFc1Z3uv37XNzYPZPam4eOJbgMCFU4BBXqKmUuU2p0XUTd0rdzpaKwxqms2p8u+0f7Q2eJbOQFT+DcihtevZFMdtod755OF85QsY9y2an2kS/qtTpb71/mblFYnl+J25M+mJx0WNzz+Jo87xLF9+1064H/TKR0J/Ojn8mH1ftNff/mbdFYv9qa4FOfD2W9r2PBpv9vXtuM6lod1Cv8AegZZwPrjpR7SP9Jh9VqPaz/7ei/1NyisT/hLdJD7XkmQ991u/H6Up8XaODzcSBehcwOFH6U/aQ7h9UxH8j+5m1RWL/wmGhbtv28E+0Tkfnipf+En0X/oIw/maPaQ7oTwtdbwf3M1aKx5PFmhxkBr9D/uIzfyFSReJdGm+5qMA/322/zo9pDuJ4auldwf3M1KKjguIbmISW0scyHo0bBh+YqSrMWmnZhRRRQIKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigArm/iP/wAks8V/9gW8/wDRD10lc38R/wDklniv/sC3n/oh6AOkooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKbLLHDGZJnWNF5LOcAfjQG46iufXUtU1a7mfQjbLZwnYJLhSRM3crjsPWpBpmvH5jryqzfeUWikL/ALuf61nz32VzqeH5dJzSfbX9Ey9rV+mm6Pc3LOEZYyI8nGWxwB+NJodn9g0KztiMMkQLD/aPJ/UmoLbw9BHdLdX1xPqFwnKNcNlUP+yvQVrU0m3zMVSUI0/Zwd9bt/l+oUUUVZzBRRWDq2p6hJrUWkaGYFnMJmnmmUsIlzgYA7k+tVGLk7IxrVo0Y80vSy3bN6iuZsfFtvaQyWviWeO01C2bZKCDiQdnXHYipv8AhOfDu/YNR3N6LBI38lq/Y1L6I545hhWk5VEvJtJrya7nQUEZGDyK53/hNdNkBFlBfXj9khtWJP54po8Sajb3lqdX0f7BZXMnlCVpw7Kx+7uAHGaPYz6oP7Qw3SV/NJtL1a0XzOjwAuABjGMVF9itQ2RbQ5znPlipqKysd6bWw1URM7FVc9cDFRS2VrP/AK62hk/34wanopWQKTTumRwW0FrF5drDHDHnOyNQoz9BUlFFMG23dhRRRQIKKKx5f+El85/I/srytx2b/M3bc8Zx3qW7GkIc/VL1NiisT/iqf+oP/wCRaP8Aiqf+oP8A+RaXP5M1+r/3195t1U1PVtO0WyN5rF/a6fbBgpnupliQE9BuYgZqnB/wkf2iP7T/AGX5O4eZ5fmbtvfGeM1ifEtbl9I0ZbGWKG4OuWYikmiMiK3mcEqGUke24fWjmvbTdpfe7GVSHs09U9G9PI6LSfEGja/HJJoWrWOppEQsjWdykwQnoCVJxWhXmOp6hqngzxdf6xrb2mr3t1oE8kDWVu1qkS2hMmxkZ5Cdxm+/u424285q74bv/Gcur6ZNd2uszWd2Cb5r8aclvCpQsGg8iQy/e2gBy/ynk5Ga0Wu39b/5f1uZtNK7/rb/AD/rY7bTNUs9Z09L3TZvOt3ZlV9pXJVircEA9QRUt1e2tjGsl7cw2yPIsatNIEDOxwqgnuSQAO5rzHwrf6jLovhzw9pl5Jpov31G4nvYo0eVEiuD8iB1ZAWMgyWVuAcDJBFifxDr2maXrMTavLdy2Piax0+K4mghDmCT7LvUhUC5PmvzgHnjGBgprntb+tUvzYnorv8Arf8AyPTKK4e58RapHoPxCuEusS6PLMti3lr+5C2UUo4x83zsx+bPXHTilS51vWvENnp8Wu3Gm276JDeSva28LStKzkZBkRlAx1G09sY7rql3t+Kb/JFNWV/X8Gl+p29FeZ2+r+JYfCum+IbjX5LiRdSjsJrP7LCkE8f2v7OXbCFxIQd2VdV3AfLjIPplO2l/6/rUQUUUUgCiiigAooooAKKKKACiiigCG7uorGynu7ltsMEbSSNjoqjJP5CsWy8R3cl3py6lpiWlvqgJtJEuDIwO3eFkXYAhKgngsMgjPStDXrB9V8O6jYRELJdW0kSE9AWUgVzyS3GtXfhy3WwvbeTTpfPvWntnjSIrEybQzAByWbgoWGATn1Fv9363+4Ht9/8AwDrbidLW3eaQSMkYyRFG0jH6KoJP0ArM0DXl17SZb9LaS3VJ5YhHLkMdjFckEAqTjoelWIdWSSG9lks76BLOZojvtmLS4AO+NVyXU54IHODxWF4Nu9ml6lHJZX6SC8ubkRzWUsRkRpGZdpdQCSD0zn1xS7+n+Q/8/wDMuaBres63Y2WoHSrGCyukEm4ag7SKp/2fJAJ9t341H8R/+SWeK/8AsC3n/oh6y4bHT5/EGkz+FNFm0uSKVnvZv7NezUwFDmNtyqJCW24AzjGeOtO+Jul3cvgDxVdJruoQwjSLpvsaR25iIEDZXJiL4OOfmzycEcYoR21FFFIAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACgkAEk4A6moXvLdLtLV541nkBKRlvmYD2rK1+WS5lttGtnKSXhJmZeqQj7x/HpUykkrm1Ok5yUXp1+XcSfxVYNZs2myC7uWfyooF4Zm7cenfNFv4dW5YXOvyG+uSc+WSfKj9lX/GtKHS7GCWKWG1iWSFPLjcLyq9MZq1UqDfxmsq0YK1C68+vp6DY40hjEcSKiKMBVGAPwp1FI7rGheRgiqMlmOAK0ORvqxaK5u41u/wBYuJLPwqIysRxLqE3Man+6g/iPv0pI/EuplPs58O3kt/H8soXCw59RIex6/jWvsZHB9fo36272bT8lbf8Aq2zOlrP1XXLDRo0N9Nh5OI4kUs7n2UVmnUfFUilU0K1hYjh5LwMF+oAyataPoIsZnvr+c3upzDElww4Uf3UHZaORR1k/uB4ipV92hFrzkmkvk7Nv8O7KcA8Savuulu00aBj+5t5LYSybfVskYJ9Kv6Lo0mmS3dzeXf228u3DST+Xs+UDAUDJ6c/nWrRSdRtWWiNKeEhGSnJuUl1be/psvkiNreF5PMeKNnxjcVBOPTNSAADAGAKKKzOqyWwVW1HT7fVNPmsrxd0Mq4YDqPce9WaKabTuhSjGcXGSumc0dP1zQpVk0q4m1m2YbXtb2cCRT2KuRj8D/wDqbc3XijVIvscOmLo5kB33bXKy7F/2dvc/59a6eitfa9Wlf+vkcTwStyxqSUeya/NpyXyfocXaa5rGtPbaVYSfY762Df2jO8QcIV+UAA8Hcea0/wDirbY9NLvkz/txOf5it5IIo5HkjiRHkILsqgFsdMnvT6JVFfSKsTTwc0v3lWTl3Ttp001XrpuVdOkvpLMNqkMUNxk5SJ9wA7c1aoorJu7O+MeWKV7hRRRSKCiiigAooooAKztb0Kw8Q2KWmqxyvEkqTIYbiSB0dTlWDxsrAg+hrRooAx9M8K6TpM809vFcTzzR+U819eTXcnl9dgaZ2IXPJUEAntUOmeCtE0e8huNPivI/s5YwW7ahcPbw5BHyQs5jQAEgBVGAcDFb1FAHI674QT+x9PstA0m0mW0uHlQTancWUkO/cWaO4iDSKSWOR0IJHtTfDvgO1stDubXWYIpHvNSXUpIYZ5HWOVCnl/vGw8hHlIWdsF23Ejkiuwooj7u39bf5IHqrf1/Wpz+qeBtA1m6u59QtJn+2x+XdRR3k0UNwNu3MkSOEdscBiCRheflGNO30extL1Lu3g2TpbLaK+9jiJTkLgnHXv1q7RRsD13Mv/hG9J/slNM+y/wChpcC5WLzH4kEvmhs5z9/nGcdunFalFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAVzfxH/5JZ4r/wCwLef+iHrpK5v4j/8AJLPFf/YFvP8A0Q9AHSUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUANd1ijaSRgqKCzMegArm/D/ia41rX7uBoBFZ+SJrUsuGdd23cfYmrHjCeT+xlsLY4uNSlW1Q+gY/Mfptz+dQ28CWfxAighAWFdGCIvf5ZcD9DXRCK5G3u/0PKxFap9ZhGDtGLV/Nyvp8t/mjpKgvrtLCwnupfuwoXI9cdqnrC14/2hf2WjJyszefc+0SnofqePwrlm7LQ9uhBTqJPbd+i3Ktn4cF3oaXE5MeqzEXH2n+JH6qPoAQMVqaXpD2dxLeXt013ezKEaQqFCqOdqgdB/hWnRUxpxRpUxVWaab0f9W9PIKKR3WNC7sFVRksTgAVhR+NNCe6aE3oQZ2rM6kRuR1w3TitowlLZHn1K9Kk0qkkr92O1nVbxb6PSNFjVr+aPzGlk+5bx5xvPqc9BUMHgfR9qyajE1/d53SXE0jZkb3AOMe1J4aYajq2r6yCHjlmFtbsDkGOMYyPYk5ro61lJ0/djp3OKjRp4u9asuZNvlT1SV9Gl3e99yO3t4bWBYbaJIYkGFSNQoH4CpKKKwPSSSVkFFFRvcQxqxkljQL94swGKaTewySiseXxLZeZ5VgsuoTf3LZNwHuW6YpP7T1mX/AFGh7B/emuVH6Dmtvq9TqrerS/M09nLqbNFYu3xJP1ewtF9VDSN+vFL/AGRqkv8Ax869Nj0hhWPH40eyivimvxf5IORdWjZorG/sCdvlfW9RKf7MgU/nij/hGLYkNJeahIw/ia5bNHJS6z/D/hg5Y9zZpjzxRf6yVE/3mArJ/wCEU0tv9cs83/XS4c/1qSPwvosf3dPiP+8S38zRy0f5n93/AAQtDu/u/wCCWJNZ0yL/AFmoWq+3nLn+dVX8VaLGcG/Qn/ZRm/kKtR6PpsP+r0+1U+ohXP54q2kaRDEaKg9FGKL0F0b+aX6MPc8yGyvoNQt/PtWZo8kAspXp7GrFFFYu19CHa+gUUUUhBRRWHNP4nE7iCz04xbjsLzMCVzxnjrWVSoqa1TforibsblFYH2jxZ/z5aZ/3+f8Awo+0eLP+fLTP+/z/AOFY/Wl/LL7mFzfqpqeradotkbzWL+10+2DBTPdTLEgJ6DcxAzWfbzeJjcxC6tNOWEuPMZJWLBc84GOtZHxLW5fSNGWxlihuDrlmIpJojIit5nBKhlJHtuH1rWFXntZNapaq27sF9G+yf4I6LSfEGja/HJJoWrWOppEQsjWdykwQnoCVJxWhXm9y2reHvHN1davd2l9f6noUwtriytTbrbC2O/DRu8m7cZvvbv4QNvOael/q8PgbSdT1LxVqj6jrEdssMFhYWpZpWTcUhV4yFLAHLSMVXBPyjptdPby/Ftfp/lcbunb1/C3+f/DHotQXl7a6fatc6hcw2sCkBpZ5AigkgDJPHJIH1NcN4f1bV7m28SWHiDVNU0ldNMUyX96lkt1DEybm3GMPAQNp525weRkZrOTVPFen+ANb8RXGtXskU0kR0hL62t1mjg3qvmOEiQZfJIUjKrtzznB/X9f13FfS/wDWh6NBqlnc6pd6dBNuurNY2nj2kbA4JXkjBztPSrdcnoX/ACU/xZ/1wsf/AEGSuspLUAooopjCiiigAooooAKKKKACiiigAooooAKKKKACiiigArm/iP8A8ks8V/8AYFvP/RD10lc38R/+SWeK/wDsC3n/AKIegDpKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAOe1D/SfHukw9fsttNcEf72EBouvk+IVg23/WWMqZ+jA/5+tGn/6T491abr9ltobcH/ey5FGq4TxzoBycyR3KY+iA11dVH+7+lzxZawlU71F+ElD9Doaw9I/0nxFrF4RkLIlsh9No+YfmavX+s2WmXNvDey+UbjdsY9BjHX061T8JoToK3D/fu5ZJ2+pY/wBAK4G05pdv6/U+lhGUKEptb2S++/6G1VPVNUttIsTdXZbbkKqouWdj0UDuTVmeeK2haW4kWONRlmc4ArjdT1k6zr+mxafZzXMVq7XQTG3zcfKrDPRQx6muyjRlUd7aI8+v7RU26e+iV/N/puXotIvvEbi58R7oLTIMWmo5Ax2Mh7n2rofsdqbVbU28Rt1GBEUG0D6dKy8+I7ngLY2St1bLSMv4dCaP7P12b5LjV444x/FBAA7fn0/CtJQvvNL7/wBLhSwdOkm27t7t6t/122NS2tbexthDaRRwQrkhEG1Rk5PFVb/XLHT2RJpS8j/djiG9seuB2qsPCunOd16bi9f+/cTMT+mBV2x0mw00sbG2SIt95hyT+JqP3K1bb/D8bv8AI6IxpxVl/kUf+EljlXFlp99cSHoohKj8SelILXXr0b7i9i09T/yxgjDsB6Fj3+lbdFHtYx+CNvXX/gfgPnS+FfqYv/CMwS/8f17fXeeqyzkL+QxUi+F9FRlI0+PK9Mkn88nmtail9Yq/zMXtJ9xkUMcEYSGNY0HRUUAD8KfRRWG5AUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABWdrehWHiGxS01WOV4klSZDDcSQOjqcqweNlYEH0NaNFAGJY+ENHsPtLRxXM8l1CYJJr29nupfLPVA8rsyqeuFIGeetS3XhjSbzRLXSZrZxaWYT7N5U8kckBQYUpIrB1IHGQ2SCQeprWooDY5mb4eeGbjS7/T57GaSDUjE14z3s5knMRyhaTfvJGBznJwAc4qdPBejCwurKb+0bu2ugoljvNVurgfK24Y8yRtvI/hxnvW/RQKytYqQaXZ22qXeowQ7bq8WNZ5NxO8ICF4JwMbj0q3RRQMKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigArm/iP/wAks8V/9gW8/wDRD10lc38R/wDklniv/sC3n/oh6AOkooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigArN1/VW0bR5byOETyKyokZbG4swHX8c1pVzfjSULZ6dGVZw+oRF0UZZlXJOB3PAH41rSjz1FE58S6ioy9l8VtPV7Fvw9pt5ZC9udTMP2q9uPNYQklVGAAuT6c1BruU8UeHZAQAJpk5/2o6l+3a1fcWOnrZRn/AJa3jfN/3wP61j6/aatb3mkSz6hHcyNeCOHMARY3Knk46j/CuiNNud5SSeuny8rmdbCxpYdQUkrNPV/3k+g7WrmK58VT2xtReyLbLBDbkcF2O4sT2AGOf/11pRaPqclvHBd6itnbRqAsVgChAH+22TWRaSNYeDdW1tpN93P5gS5x8xGdqkenzc/l6U9dcuX+G4eRnbUZCbHDH5jKWKfnjmopRSipUrO73t+XZHXjsyjh6rwz3hG/3afe3cq2mr+GpYprq7kvLiSB/wB1BdyGRp8nClF6Nk5H8639BsLx7651nVY1hnukWOK3XnyIhyFJ9T1P+QLlnoNjaw2GbeN5rGIRxSleRxyfx6/ia0qKtbm0V/n+hw0qWJqOM8VK7jslpr3fpey6dd9iiiiuU7wooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKwpr/xKs8iw6NbSRhiEc3YBYZ4OMcVUYuRjVrRpJNpv0Tf5G7RXPf2h4o/6Adr/wCBg/wo/tDxR/0A7X/wMH+FX7J9196MPrtP+WX/AIDL/I6Gqmp6tp2i2RvNYv7XT7YMFM91MsSAnoNzEDNZ1vfeI3uoludGt4oWcCR1uwxVc8nGOayfiWty+kaMtjLFDcHXLMRSTRGRFbzOCVDKSPbcPrWc04tLu0vvdjenWjUi5JNWvumul+qN/SPE2g6+8qaDrenam0IBkFldxzFAemdpOM4NadcT4qvvFOheDJbh9X099Re+tIYZ7bTWjRFknjRgyPK+7IY9CvHTB5qHXn1XTtQstMtPEniPUL6ZZrhrbT7TT/OMeUAYvLGsSRqcgA/OxfqdtLT+vvN7aXZ3lFeYW3ibxJqHhzwxGb97C+vNcn0y8ne2haRo4xOMlRujEn7tTlcqG5wV+U2r/wASa34dt9esP7QOpXFrdWMFpe3sMYaP7UyoTIsQRWCElhgLnofWj/O35f5g9Gk/63/yPRaK4bxFPrvg7Q2nh1q/1ya9ubazgW6gtFeBnk2syFViQkg4Ac43BecZBteEpvE/9r3MGtW2rHTfIDxXGsfYRMJd2Cg+yttK7cHlQQQeTkAC1/ryuJ6HX0UUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABXN/Ef/klniv/ALAt5/6Ieukrm/iP/wAks8V/9gW8/wDRD0AdJRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFYmqXkt9d/2Ppkm2Vhm6nX/AJYJ6f7x/wA+o0p03N2/pFRjzMe+vPNcvDpNhLf+WxWSQOEQEdgx4JrEmhv5/GOiyaowV5nmkFspyIURcge5JI59q620tIbG0jtrZNkUYwo/z3rEl/ffEaBO1vprSfQtJj+QrohUirqmraPXr/XocuMqWhGMNLyivO10396R0Ncp8Q3aHQ7OeNirw30bqR64b/Gurrh/HN882YFAFpp8kM1yTyWLNgL+WTXNTfvpd9AxUb4ec+kVd/It6zafZ/DGi6Pg5ubmCGQe2dzH8xUdnoN5/wAJzcNPGRpcM7XsJI4eZ1Ufocn/APXVyyivvEOsWuq3cBs7C0LNawSD95KSMb29B6CulrXmdKKguxwuhHHVniZ3+JNeaX6NtvzsFFFFc56wUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFVNQ0uz1VIFv4fNW3uEuYhuK7ZEOVbgjOD2PFW6KAKep6VZ6zZi01KHzoBLHLs3FfnRw6nIIPDKD+FVdX8NaZrlxBcXyXCXECskdxaXk1rKFbBK74mVipKg7ScZAOOK1qKAMa18I6HZW9lBa2CxRWF297bIsj4jmffubrznzH4PHPTgVPc+HdJvP7R+12Uc41SNYrxZCWWVVBCggnAwCemK0qKHruHW5gxeCtDjsbuzmt7i9gvEVJl1C+nuyVU5UAyuxXBORtIwcHqKsaR4a0/RLiSeza+llkUIXvdRuLsquc4XznbaPXGM4GegrWooDpYKKKKACiiigAooooAKKKKACiiigAornvGnir/hE9FhuYrX7beXl3FZWVsZPLEs0jYUFsHA6knB4FcXD8UtbtfAviDXdZ07TxPoGuf2dcxWruUaINGrMpbkkGTI4GcdBTinLb+tv80Oz/r5/wCTPVaKQHcoI6EZpaQgooooAK5v4j/8ks8V/wDYFvP/AEQ9dJXN/Ef/AJJZ4r/7At5/6IegDpKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiikLKpAJAJ6ZPWgBaKKKACiiigAooooAKKKKACiiigAooooAo6zf/wBm6RPcLzIF2xj1c8D9aTR9OXTNOjixmZhumfqXc9ST35qnff8AEx8TWlkOYbMfapvduiD+tbddE/cpqHfV/p/n8zR+7FLuFc74d/03Wta1Q8q9wLWI/wCzGMHHsSa1Na1AaXol3ek4MMRZc926KPzxUPhqwOmeHLO2fPmCPfJnrvb5m/U1EdIN99P6/A86p+8xUIfypy/Rf+3fcalYfh1Rdf2lfOAwurpguRkFF+Vf61qahObbTLqdeDFC7g/RSaqeHIBbeG7CMDGYVcj3b5j/ADrmes0etHShJ92l+r/Q06KKK0OYKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKyZdR1hJnWLQvMRWIV/taDcOxx2q4Qc9rfNpfmUot7GtRWN/aet/8AQv8A/k6n+FH9p63/ANC//wCTqf4Vp7Cfdf8AgUf8yvZvy+9GzRWTBqGryXEaTaJ5UbMA0n2tG2jucd6xfinp1lqPw71EahZ290IQkkYniV9jbgNwyODgkZ9zWVSLp72+9P8AIiS5dzsKK4K6h1q1+KFjpXh3ULHTdLi0Ustk9i7xKFmVeESVFBwQAccDIwc8VPEPj3U9E1O9kt72wv7SzukjktLbR7uUopKhlkvEYxRyAEnay8fLnGc1K1Sa63/B2Jbte/T/ACuehx3dvLdS20VxE88AUyxK4LRhs7Sw6jODjPXFS153fajqGkeKfG+paTBHcz2dtp87wupJkiUOZAuCPn2BtvbdjNdHYeIJtX8WG10vyJNJt7FJ57jBLPJLzEqEHGAgLHIP3kx3pJ3/AK/rsHn/AF/Wp0NFFFMYUUUUAFFFFABRRRQAUUUUAFFFFAHI/EiHQrjw1BH4j1dNFUXsUllqD4AguUJeNiTxj5TnOARxkZFcfo3ws1i/hu4dY8UaffaBqmrHWLmPT7QqbxyVITeXYLHlFOBknpmvRvE+u6doGkedq0FxdRTv5CW1taPcPO5BIQIoOcgHrx715nB4O8Rapqi33gnR/wDhXFuzh5Jnudzzr72KEwg/7xzThpLT+tv8l92w5fDb+uv+bPZKKBkAZOT60UhBRRRQAVzfxH/5JZ4r/wCwLef+iHrpK5v4j/8AJLPFf/YFvP8A0Q9AHSUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFc14okEeueHc9TekAfhj+tdLXJeL3VfEnhhWOCbw4/NB/UVtRV5/f+R5+ZS5cM35x/wDSkdbRRRWJ6AUUUUAFFFFABRRRQAUUUUAFVNU1BNM06W6kG7YMKg6sx4A/OrdYd3/xNPE0FmOYLAC4m9DIfuD+ta0YqUry2Wr/AK89i4JN67FrRdPltLeSe9bfe3TeZO3YHso9gOK0qKKmc3OXMyZNyd2c74pH2y90fST9y7ut8g/vJGNxH8q6KueuP9I+IlnH2tLB5voXYL/IV0NVPSMUcOG96rVn52+SS/W5keKpfJ8L3zese3/vogf1rTtovItYov8AnmgX8hisjxV8+m21v/z8XkMX1y2f6Vt1zL42etLTDxXdv9P+CFFFFaHMFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAVkeJ/Dy+J9FfTJdRvLCGVgZHsxFucDnafMRwBnB4APHXrWvRRuBgXfhT7Vf2GoJrWpW2o2dubZryEQbrmMlWKyK0RTkqDlVUjnGAaoXnw7s7uG+tBrGrW+mX07XMunwSRLEJWcOzBvL8zlxkqXK8kYxxXXUUf1/Xz19RNXVn/AF/X5GXFo0Fjq2qaxAJp7m/ijWSEsu0+WGChc4wTuOcnH0rL+Hvhj/hFfCUVrLAILu4ka5uYhJ5gjdukYbuEQJGPZBXUUULS9hhRRRQAUUUUAFFFFABRRRQAUUUUAFFFFAHDeMf7X064aRvHj6JBqt/a2mnRppEc5hkKsDHkg7vMbDbmA27cZ5rK1zSfF3hrR5tW1r4t3FvY25XzZToFq23cwUcBSepFL8UfBUWpiHXdT8cX2iWFhewXjpI8fkQbOMx5XIkJPBJYZJ+U9K4+71v4Z6vGlrr3xY1zVrHzEeSyuWAim2kEK+2AEjIHGacLNrvf8NP+CVK2j8v6/Q9w1bUBpWg3moEeZ9lt3mx03bVJx+OKwYLvVtNu9Dl1DUnvE1VvKngeKNVhkMZkUxlVBwCpXDFuCOcjnodQsYtU0m5sZSRDdQtExHUKy44/OsK20XWLm60hNYNktvpLb1kgld3uXCFFYqVAjGGJIy3OBn1S3+79bkfZXz/Sx0dw0627m0jjlmA+RJZCisfQsASPyNc3oevXx8Ianqms+XJPZTXe5IvuhYmbCg4BIAGMkZNbMLausN6biOxlkEzfY0SR0Vo8DaJGIOGznOAR0rH0LQ9Si0fUtM1uCzWC9kuJN9tctIcTMxKkNGvQN1yc+gqXfW3b/ItW0v3ILe+1bT20G81DUmuk1aRYZ7Zoo1SFnjLqYyqhsArt+ZmyD61a+I//ACSzxX/2Bbz/ANEPUdroerzSaPbasbP7JpDiRZoZXaS5ZUKISpUBOGJI3Nzx71R+JvhzQ7jwB4q1O40bT5b8aRdOLt7VGlDLA207yM5GBg54wK0dun9IhX0v/TO2oooqRhRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABXE+NXz4s8LJ6XefzeP/Cu2rgfHbt/wl3h9R/DIrZHbMi/4V0YbWp9/5HkZxK2DfrH/ANKR31FFFc564UUUUAFFFFABRRRQAUUUUAUNU1RdOjjVImuLmdtsMCHlz/QDuabothLZW0sl4Va7uZDLMV5AJ6AewFVNGX+09SudYk+ZNxgtPQIDgsPqc1uV01LU4+zW/X/L5fmaS91cq+YUUUVzGZz2k/6R411y46iFYbdD/wABLN+pFdDXPeEf3sOqXh/5etRlZT/sghR/I10Na1fit2scWB1oc38zb+9t/kYmufvdY0W2HOblpiP9xc5/WtusO3/03xndS9UsbdYV/wB5/mJ/LityuaGrbPWr+7GEOy/PX8mgooorQ5gooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAOP+I/h6/wDEdhocGnQidbXXLS6uomZQGgR8vncRnGc46nHArz7xHpHiGwuPE3hvSdA1Ce41rXIdW0rVLeIG3t23Rl2kfI2FTG3Xk5rsfFOly6O82qav8TdT0OynnIjRxarFGWyQiloyegOMkniubGt6DuH/ABfW4PPTzbLn/wAhUU9Grd//AJH/AORRUm7W/rr/AJs9jGdoz1xzS0i/dHOeOvrS0ELYKKKKBhXN/Ef/AJJZ4r/7At5/6Ieukrm/iP8A8ks8V/8AYFvP/RD0AdJRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAV5945d18XaaVGRHFG/8A5GH/ANavQa8/8anb4naRvuR2MBx9bof4V04X+IeLnX+6P1X4anoFFFFcx7QUUUUAFFFFABRRRQAVk6/eSRWi2dpzd3p8qIf3R/E30ArWJwMngVh6Kp1O+m1qUfI+YbRT/DGDy31Jreikr1JbL8+hpBfafQ1bK0jsbGG1h+5EgUH196noorFtt3Zm3fUKiu5xa2c056RRs5/AZqWsXxhcfZvB+pP6wmP/AL6O3+tOC5pJGOIqeyozqdk39yE8GwG38H6erfeeMyknvuJb+tbE8yW9vJNKdqRqXY+gAyajsLf7Jp1tbDjyYlj/ACAFZvitm/4R+SFCQ08kcXHuwz+lKrPWUjTA0LRpUH5L9A8MQuNKa8nGJr6Vrhx6Bvuj6Yx+dbNNRFjjVEG1VAAA7CnVEVyqxtVqe0m59woooqjMKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigDI8S6zp+h6Yl1q1rcXMDShAltZvcsGIJztQE44POK8z8ceIrHxXoMOm+DfDurPrv2yCWymfRpYEt2WQEu7uoAXbuB9QcV2nxJ8Q6r4d8O2j6F5Ed3f6jb2AubhN8dqJW2+Yy5GccfiRUEfhfx2FXzPiKGb+IrocAB/DcaI737P8rMb0+7/ADR2q52jPXHOKWgZAGTk+tFBK2CiiigYVzfxH/5JZ4r/AOwLef8Aoh66Sub+I/8AySzxX/2Bbz/0Q9AHSUUUUAFFFFABRRRQAUUUUAFFFFABRRXOXGvau+tX1jpOkxXaWZQPI1yE5ZQ3Qj6/lVxg5bGFavCik531dlZNva+y9Do6K53+0/FP/QvW/wD4Gr/hR/afin/oXrf/AMDV/wAKr2T7r70YfXqf8sv/AACX+R0VefeMEVtX1WQt/qrK24+s4/pWxe6/4j0+ylu7rQIVhhXc5F4DgfQCsPxOVuz4jljP/LlaH8C4b+WK6KEHGd3/AFqjys0xNOtQcIp3V3Zpr7Mu6XY9FoooriPowooooAKKKKACiiqmq366Zpc92w3eWvyr/eY8AfnVRi5NRW7Gk27IreI7p7bRJlh/11wRBEPVm4/lmr1larZWMNtH92JAg98DrWPbRX2qa1Fc6nZG0is0zHEXDhpG/iyOuBW9W1T3IKn83/X9bly91KIUUUVzmYVz3jT95ocNt/z83kMWPXLg/wBK6Gue8T/vdR0C2/vagsuP9xSf61rS+NM4sfrhpR76fe7fqdDWJ4j/AHkmkwf37+NiPUKCTW3WJqn7zxRosXZTNI34Jgfqa5qnwnrYX+Lfsm/uTNuiiitDmCiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAPOvjg5Pw/ht57pbTTrrUrWDUZ8jdHbNIN7LkHkHBz2AJpieDPCoVdvjvXiABj/iqJef/H66bxrbJdaHGkvhKPxWvnqfsMnk4Xg/vP3pC8dPX5q821L+xtFjgl1P4GWltHPOlukjDTyokc4UEhvlyeMnAyQO4ohvbz/y/r5jlsn5f5ntQ+6MHPHWlpF+6OMcdPSloJWwUUUUDCub+I//ACSzxX/2Bbz/ANEPXSVzfxH/AOSWeK/+wLef+iHoA6SiiigAooooAKKKKACiiigAooooAK53QP8AkavEf/XaH/0CuirA0Bg2v+ICv/P0g/KMCtYfDL0/VHFiP41H/E//AEmRv0UUVkdpR1yH7T4f1CHGS9tIo+u0155HMLzw74glAJdbCwUn/gAz/KvUGUMpVhkEYIryrQkaPwf4oR/vIqRk/wC7kV24Z+6/Vfmj5vOI/voecZ/hF/5nqcDFreNmOSVBJ/Cn1DZP5lhbvjG6JTj8Kmrje59FF3imFFFFIoKKKKACsPU2Gpa7ZaanzR27fabn0GPuA/U9q17q4S0tJbiU4SJC7fQCs3w5bPHpn2u4H+lXp8+Vj7/dH0AxxXRS9yLqfJer/wAv8jSOicjXooornMwooooAK57Vf3vjjQou0MdxKR9VCiuhrnh+/wDiOxHK22mgH2ZpM/yFa092/JnFjNYwj3lH8Hf9DoaxH/e+Ool7Q2Bf8WfH8hW3WJof+l6pqupHlXm+zxH/AGUGMj2JNc09WketQ0jOflb79P8AM26KKK0OYKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAxfFeoappejC50Y6SsqyqHbVrpoIQhyPvKp+bO0AY71w3i3S/HPinRU0TxPN4W0TSry4iWe5gvJnmbDhgkYdFG8leOa3firpOo6n4XsptKtZL59M1S21CWyjxuuY4nyyKD1PcDviuZ8ceKYfiJ4ZHhjwrpesSale3EG6S402WBLAJIrmSR3UAEBTjBOTRDVrvdfpr+f3Den3f5nrYGFA9BS0ijCgE5OOtLQStgooooGFc38R/wDklniv/sC3n/oh66Sub+I//JLPFf8A2Bbz/wBEPQB0lFFFABRRRQAUUUUAFFFFABRRRQAVz3htSNa8QNjg3oA/75H+NdDXP+GX36hrxxj/AImLD8lUVrD4Jf11OKvb29H1f/pLOgooorI7QrhF0+ay0PxgJ4HjjkllkiLqQGXBIIz1+td3WZ4lTf4W1QHj/RJT+Sk1tSm4u3e35nn46gqkOfrFS/FNFnTGD6TZsvRoEI/75FWqpaN/yAbD/r2j/wDQRV2s5fEzrou9OL8kFFFFSahUc88dtbyTztsjjUszHsBUlYviRjPBa6Yh+a+nVGA6+WOWP6VpShzzUWVGPNKxHrlzHqVhYWtrJvj1GZRuHGYx8zH9BW6AFUBRgAYAHaub0jR7q18QOJlP2KzV/shP+2cn8hkV0tbYjljaEHdb/f8A8CxdSytFBRRRXKZBRRRQAVz2hfv/ABT4gus5Alit19tic/qa33kSPHmOq5OBuOMmsDwZ+90m6vP+fy+mnB9Rux/7LWsdISfy/r7jire9iKUO139yt/7caes3/wDZuj3N0PvomEHqx4X9SKNFsf7N0W2tT99Ey/8AvHk/qTWYzv4j1VFhUDTLGbc8h/5byL0A/wBketdDXNH3pcx69Reypqm93q/0X5/eFFFFaHKFFFFABRRRQAUViX/iRbTU3srWwur94UDTm1UMYs9BjuT1xUEnizzkMWmaVqE92eBFJAUCH/bY8AVyyxlCLactV6/h3+QuZHRUVzyjxdAgYnS7okZKYdCD6A9KBp3iib97JrtvbuefIisw6D23Md1L6zJ7U5fh+rSFfyOhorl18R6paSPp13pEl1qanMf2f5YpU/v7j90dj71OvjbSFTbdme1uhw1rJA28H04GDSWOw/2pW9dPlrpfyDmR0NFc+PFE8oxaeHtWdu3mwiIH8SaZ9m8QazMZbiZtDhjH7qGJlldz6semPan9bg9Kacn5L9XZfiHN2OjormzqmraCca7GL6y/5/rZMMnu6f1Fb1rd297As1pNHNGwyGRsitKdeFR8uz7Pf+vNaDTuTUUUVuMKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACisDxd4mfwtZafdCxN3DdajBZTP5vli3SVtvmk4OQCRxxnPUVv0dL/1/WobHP+Hf+Rg8Vf8AYTj/APSO3roKz9O0v7BqGq3PneZ/aF0txt248vEMceM55/1ec8dfatCm+novyKk7sKKKKRIVzfxH/wCSWeK/+wLef+iHrpK5v4j/APJLPFf/AGBbz/0Q9AHSUUUUAFFFFABRRRQAUUUUAFFFFABXPeFkxda6/rqkg/IL/jXQ1z3hJi7a4T/0F5x+QUVrH4JfI4q1vrFL5/kdDRRRWR2hWfr6l/DepqvVrSUD/vg1oVS1n/kA3/8A17Sf+gmqj8SMa6vSkvJ/kR+Hf+RX0v8A684f/QBWjWX4YYt4V0wt/wA+sY/JQK1Kc/iYsM70IPyX5BRRRUG5HcTpbW0s8pwkSF2PsBmsfw/YB4E1a9UyX1yC+9yT5anoq+gxT/FDM2kLbKSDdTxwcdcFuf5VsKqogVBhVGAB2FdCbhR0+0/wX/D/AIGnww9RaKKK5zMKKKKACiis3XNat9F09pZWBnYEQQjlpX7AAc9cU4xcnZGdSpClBzm7JGRc2sPinxVLb3S+bpulptZMkCSdvp/dH5Grmrxponhf7FpK+RvZYIFDE7S7c8nnuan8Naa+maFDHcZNzKTNcMepkbk5+nT8Ki1j/Sdf0ez6qsrXLj02D5T+Zp4ib5eSL02/4JOV0Iqoq9SPvu8n3tvy+iSSNSys4tPsorW2XbHEu0e/v9anooqVpobybk7vcKKKKBBUNzd29nEZLueOBB1aRgo/WqWr63FpJii8ia6up8+TbwLuZsdT7DnrWfpnhuO4ZtR8RW8dzqE7byjjcsK9kA6cVy1K0nP2dJXfXsvXz8hX6IuW3ivQroMY9Ut12nB81/L/AC3YzSan4k0+y0qe6gvLe4kVMxxxyqxdjwBgH1Iq9caXYXe37XY20+wYXzYVbaPQZFQDw/o6ypIml2aOhDKyQKuCO/AqJLF8rScfXX8tRe8R+HdLOl6Siz/NdzHzbmQ8lpDycn26fhWrRRXTTpxpwUI7IpaBRRRWgBRRRQAUUUUAHXrWHdeENKmkM9pE1hd53JcWrFCp+nT9K3KKyqUadVWnG4mkzmpF17Q5kuGnl1u2f5ZoliVZE9GUDr7j/Iv2HiWwvZ3t5fMsblOtveKI3I9Rzgj6Gtaql9pVhqaqNQtIbjb90ugJX6HqKw9jVp/wpadnr+O/5is1sW856UVzyeFprfMNjrd9bWf8ECEHy/YMcnHtTJLTXdEAns72TV7deZLW4A83Hcq46n2NH1ipFXnTdvKz/wCDb5X8gu+x0lFUtL1e01i186ykztO142GGjb0YdjV2uqE4zipRd0ygoooqgCiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigDxnxd490LXLe98PeMrYRCLxMmlm2h1JrZ/IZfluXwQTHgnI6dOa1V8NfDcY2+MrjjoP+Erl/8AjtWvi5pegPZ6Nd6j4e03U9QudXtLO3lu4iQhd8fOVILIBn5CdpOMio/iD4f8JeDfB82u2ngjwzN9lmh86OTSovmjaRUbbheG+bI69KcNl3vb8F/w/wAypavTt+r/AOGPShjaMcjHFLSKQVBXgY4paRC2CiiigYVzfxH/AOSWeK/+wLef+iHrpK5v4j/8ks8V/wDYFvP/AEQ9AHSUUUUAFFFFABRRRQAUUUUAFFFFABXP+ElCrrJH8WrXBP5gf0roK53waS1nqZJyTqc5J/EVrH4JfI4q3+80v+3v0OiooorI7QqvqP8AyC7r/ri//oJqxUV0oezmVhkGNgfyprciavBozfCb7/COmHGP9HUflxWvWJ4OYN4P00g5/c4/U1t1dT436mGDd8NTf91fkFFFFZnUY2t/vdU0a367roy4/wBxSf61s1jXX77xjYJ18i2ll+m4ha2a3q6QgvL9WaS2SCiiisDMKKKKAEZgilmOFUZJPaub8NwHV7qXxJeoC85KWaMM+TCCR+BPJP8A9erni67a08L3hi5lmUQRgdSznbx+ZP4Vo6faLYabbWkf3YIljB9cDGa1Xu079zhmva4pQe0Ff5vRfdZ/gyxWHp3+n+KNQvusdqos4j6kfM/68VsXEvkWssp/5ZoW/IZrM8Kw+V4btWPLygyux6sWOc1zS1kl8z16fu0pz72X36v8rfM16KKK0OYKKq6lqVtpVhJd3smyNB+LHsAO5NYkja94gj8lbc6NZOPnldw07r6KB938ea56teMHypXl2X69F8xNj9II1XxVqOqAh4LYCzt2HIOPmcj8SBmuiqCxsrfTrKO1s4xHDGMKo/n9anp0KbpwtLd6v1f9WBKwUUUVuMKKKKACiiigAooooAKKKKACiiigAooooAKKKKAOf1WwudP1VNb0eAzORsvLZODOnZh/tD9akt/GOiTyJG159nlYfcuI2j2n0JIx+tblRXFrBeQmG6hjmjbqkihgfwNcjo1IScqMkr62aur+WqtfruTZ9CRWDqGQhlIyCDkGlrnpPCa2gMnh++uNOlBysYkLwk+6GltfExtpxZeJIhp91nCy5/cze6t2+hoWIcHauuXz3X39PnYL9zoKKZFNFPGHgkSRD0ZGBH6U+upNPVFBRRRTAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigDK8SeHNP8V6HLpWrJIYJCrq8TlJInU5V0YdGBGQa5j/hWU1/cWq+KfFusa9YWcqzRWFwIY43ZTlTKUQGTBweT1Fd5RQtHdA9VZmZ4ku5rDwtql3anE0FpLJGR2YISDXNrptnoV34Wu9MhSKe7mFvdTIAGuleFmJkPVzuUNk5Oc+prtJoY7iB4ZkDxyKVdWGQwIwRWPY+F4LK7tJpL69u0sVK2cFw6FLcEbeNqgsQvyguWOCfXNC0f3fr+YPb7/wCvkReNZpE0GOGN2jW6vbe2ldTgiN5VVue2QSPxqpbWNro3joafpFtFZ2t5pjyyQQIEj8xJFUNtAwDhyCe+B6VrSeHre4sb60vLm7uorycz/vpsmA8ECM/whSoIHY0lp4fS3nuLm41C9vbueHyPtMzIrxp1wgRVUcnOcZzjngUv+D+VrfJj/r8f8jltC0iDwpLotprXhzSBNI32eLVLTEkhmCkgtuiVl3ANyC3I565qz8TdetIPAHirT3h1AzHSLpNyabcPFloGx+9CFMc8ndgc5xg1u2/hwJe21zqGqX2ptaEtbrdGILGxGN2I0Xc2CRls4yaqfEf/AJJZ4r/7At5/6IeqEdJRRRSAKKKKACiiigAooooAKKKKACsDwgANNvSP4tQnJ9/nrfrnfBH/ACA5/wDr8m/9CrWP8OXy/U4qv+9U15S/Q6KoLi8gtHhW4k2GeQRxjBO5j24qeud8WSiJ9JYnBW+RunpTo01UqKDPQhHmlY6KkYBlKsMgjBFLRWJBz/gckeD7SNvvRNJGeMdJGroK57wd8llqMH/PDUp4+/qD/WuhrWt/EZxYD/daa7JL7tApskiwxPJIdqIpZj6AU6szxHN5Hhu+fpmIp/318v8AWppx55qPc74q7SKulO+pa9NqqwSR2xt1hheQYLjduJA64rdqGyh+zWFvB08uJU/IYqaqqzUpabLQc3d6BRRRWRAUUUUAc5qmdW8XWGmrzBY/6bcf73SNfzyfpXR1zvhL/Sv7U1Qjm9vX2N6xp8q/yNdFWtTR8vY4cH78HWf23f5bL8NfVmd4hm8jw5fvnH7hlB9yMf1qxpsXkaTaQ9PLgRfyUVm+LSW8PSQDrPLHEPxcf4VtgYGBwK5l8b9D15aYePm3+CQVHcXEVpbSXFw4jijUs7N0AqSub1b/AIn2vw6KnNpa7bi+I6Mf4Iz9ep9qivVdOHu6t6L1/rV+Ryt2E022m8Q3yazqSMlpGc2Fq47f89WHqe3p+tdLQBgYHAop0aSpR3u3u+7/AK27AlYKKKK2GFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFRz28N1CYrmGOaNuqSKGB/A1JRSaTVmBBaWVtYQeTZQpBFknYgwMmp6KKElFWQBRRRTAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigArm/iP/ySzxX/ANgW8/8ARD10lc38R/8Aklniv/sC3n/oh6AOkooooAKKKKACiiigAooooAKKKKACsDwWwbw7uU5BuJiD/wBtGrfrnfAoI8KQkjrLKR/38atV/Dfqv1OKp/vdNf3ZfnE6KuS8fSeXa2DD7yzFgM+grra5PxxEJhZIeyTv0z92PP8ASujA2+sRv5/kz06H8RHWA5GRyKKgsX83T7eT+9ErfmKnrjas7GT0Zz3hv93rPiCDP3b0SYz/AH0B/pXQ1z2l/uvHGvIf+WsdvKBn0UqeK6GtKvxfJfkcOB/gtdpSX/kzCue12+ttR09La1mWXzL6O3kC54O7JH6Vqazff2do9xcj76piMerHgfqaw7Two9rfabPG6hIlD3SsfvSAHDD15Yj8K3w8YRXtJu1tvVa/5Hp01Fe836HU0UUVxmIUUUUAFZXiLV4tK0mU7/8ASpkKW0S8tJIRgYHXqRWqTjrXNaAn9tavc+IZxuiybewBH3Y1OGce7HPvWkEvieyOPE1J6UqfxS/BdX8vzaNTQLA6Z4fsrNhtaKFd4/2jy36k1o0Viah4kTTtWa2ktpJYI4g800Qz5RJOMj0rKc1fmkduHw8nFUqS2X4ITxAfPvtIsU+Z3u1mZf8AYQEnP6VuViaRDPfapPrN5C0IkjEVrFJ95Y+pJHYk1L4k1GfTdJDWO37XPNHBAGGQWZv8M1jKoqcJVZbHRiGoKNP+Va+r/wAthdZ1iWwmtrSxtTd311u8qPeFUADlmPYUugaVJpdi/wBrdZby5kaa4kXoWPYewHFVdG07UW1m61TXUhW42LbwCFsqEHJIzzyfXmt6sqMZVJ+2ndb2T6Lv6v8AI5FrqFFFFdhQUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAVzfxH/5JZ4r/AOwLef8Aoh66Sub+I/8AySzxX/2Bbz/0Q9AHSUUUUAFFFFABRRRQAUUUUAFFFFABXP8AgZxJ4NsnHRmlP/kV66Cue8CKU8E6eD6OfzkY1qv4b9V+pxTv9cp/4Z/nA6Guc8Sw+dqdgnrb3Y+uYsf1ro6xNRj83xXpqnp5E2fxAH9a0wz5al/J/kz0qTtK/r+Rc0F/M8P2Df8ATug/IYq/WP4Vkz4XtC5wUVlbJ6YYitOC7t7rf9mmjm2HaxRgcH04qK0Wqs/Jv8yZr3mYafuviPKO02mBvqVkx/I10Nc9fDyvH+lP/wA9rSaPp6YbrW/JIsUbSSHCoCzH0ApVNeX0PPwmjqR7Sf4pP9TG1T/T/EGn6cOY4SbuYfThR+dbdY3h2N5459WuBiW+fco/uxjhR+XNbNVW0apr7P59T0Z6Pl7BRRRWBmFFFFAGJ4tu5bfQWgtji4vpFtIT6M5x/LNaljZx6fYQWkAxHBGEX6AVjeI/32teH7bn5rwy4/3EJ/rXQVrLSCXzOKl72JqSfSy/C7/NfgZms6s+li1EFs11JPLsESnBIwSSPfpVfw2kkyXupXEbRyXs5IRxhlRflUH8jRdf6T40sYu1pbSTn6sdgrbrlScpNvoezOUadFQS1krt/N6fkyrqGp2WlW4n1C4SCMnALdz6ADk1zo1O38S+KNNSx8x7ayD3MpeMqC2Aq9R2JzVm0X+3PFk9643Wmlk29uD0aU/fb8OBXR1y2nidbpQT7auz732uu23qcOrCiiiu8oKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACub+I/wDySzxX/wBgW8/9EPXSVzfxH/5JZ4r/AOwLef8Aoh6AOkooooAKKKKACiiigAooooAKKKKAGyuY4XcdVUmsPwR/yJenf7jf+hmtq4/49Zf9w/yrH8GJs8G6cBz+6z+ZJrVfw36r9Tilf65D/DL84m5WNcgt4zssdFtZD+oFbNY0m4+N4AOi2LH6fOP/AK1VQ3l6P8j0Ybv0ZiyEr4T1GyX739otbr+MgP8AI1q2kEWkeKntoEWK3vYA6IowA6cED8Dmsmf/AJGGSw7tqsU5HqNhY102p6XHqSxMZZYJoGLRSxHDKT1/Cu6tNRtGW0rv70rfijebS0ez/wCAZmvfu/E/h2f+7PLH0/vR4/pVvxNO0Ph+4WP/AFkwEKD1LHH8ia5bX9YntorOC/O7UtPvVkDlcLKmDtPHrkfka6G/nXU9R0SGPmORvtjD0Crlf1NZqhKDhKWyv+Gv4nBh6MqdepKWzaf/AJKl+htW0K21rFAn3YkCD6AYqSiivPbbd2aBRRRSAKKKKAOev/33j7SI/wDn3tp5fpuwtdDXPQ5m+I1y/OLfTkj/ABZy39K2Zr6CKxmu1kSSOFWZijAjgcitKjskvL/gnHg4uUqkl1k/wtH9DM0r/SfEur3Y5WMpbIf90ZYfman1/Vl0rTz5fz3k+Y7WFRlpJD049B3qhp10uheC/wC0bwZkkU3DjoXdzkD8cgVNoeiyI66rrDGfVJVyS/SAH+BB2rzZTm/3VP4nq32T/Xt6Hp4qV6rjHpp92hd0HTf7J0S2tG5kVcytnO5zyxz9TWhRRXXCEacFCOyMNgoooqwCiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigArm/iP8A8ks8V/8AYFvP/RD10lc38R/+SWeK/wDsC3n/AKIegDpKKKKACiiigAooooAKKKKACiiigCC/Ypp1yy8ERMR+RrM8H/8AIn6b/wBcR/M1paj/AMgu6/64v/6Cao+FVC+E9MCjA+zofzGa1/5dv1/zOKX++R/wv80a1Y2SfHWP4Rpv6mX/AOtWzWNGxbxxMD0WwUD/AL7JqqP2vRnow6+hmyWpPxLifHHkeaf++Sua6uojawG8F0Yl88J5YkxztznH51LRWq+05fJJCnLmt5I5rxxBCPD8l40a74ZYSX6HaJBxn05qHwjBMb66a55Nin2JD6gMSf6VvazYDVNFu7I4zNEyrns3Y/niuek16WT4eQTxE/brlVs0HRvOJ2H8eCa2p1JOi6a6v7v61PNniFhq1SUtnFP/AMBbv+aOuorA8HTytoRtbpy9xYzyW0jMcklW4/QityaVIIHmlbbHGpZmPYAZJrklHlk4nRRrKrSjV2ur+n/DDyQASTgDqTXNr4va8DDRtE1G9+YrHKUCQyYOMhyelM1zVU1ays9N0W5V31UlTLGf9XCPvt7HHGD710VrbRWdpFbW6BIokCIo7AVpZQjeS1OZ1J4io40Z2it2rO7etl00WvzRgf8ACbafFp8z3qtbX8Aw9jJ/rC/YL6g9iO1SeGtXv7qa6sNdjSLUIdsoVBgGNgCPyOQa2pLO2mlWWa3ikkUgq7ICQR0INYPipX02a08Q2yF3sjsuFXq8LdfyOCPxqo8kvdS1ZjVWJoWrVJ3jHdJWuurfmtHp2fci06526l4o1cY2RsIUPvEnP6modI8LF7GzaS6kFnPGs1za54kfGR+HPP0H4QwxSW/w2hRv+PnUnUk/3mlfP/oNdBrOrRaJYIkSGW6lHl2tuoy0jYwOPQcZNceKlTU5SntFf8D9D18uqTw+AjNaObbfz1/OX4FCdRrnimK0QD7BpJEkoA4eb+Ff+Ajn9K6Sszw/pbaVo8cMx33MhMtw/XdI3LH+n4Vp1lhoSUXOfxS1fl2XyRK7sKKKK6RhRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFc38R/+SWeK/8AsC3n/oh66Sub+I//ACSzxX/2Bbz/ANEPQB0lFFFABRRRQAUUUUAFFFFABRRRQBS1k40K/wAf8+0n/oJqv4W/5FPTP+vZP5VY1ogaBqBJwBayZJ/3TUfh3jwvpeP+fOL/ANAFa/8ALv5nE/8AfF/h/VGjWNbtu8Z3o/u2sY/Un+tbNY1k27xfqfH3YYR+hNVS+Gfp+qPRhs/Q2aKKKwMwrDt/CWn2+qm+33Ejee1wkLyfuo3bqVUfWtyiqjKUdmY1KFOq05xvbY47UfDOqXuu30VvcfZNMvCtw8iEbvNVcAY6/ewx+lR6hrlxeeEbjTbnEermZLCVP7zOcbh7MuTXa1lXnhyxvdetdWkDLcW3TaRh/Tdx2reNZO3P0/T/ADPMrZfKKk8O/jund9H27Wu2vViaZ4Z0zSL57uwg8uV4hGcHgAdSB2JwM/Stais238Q6XdaibCC6DXIZl2bGHI68kY7GuaU7v3mexQwyhBqjDRauyNKkZVdSrgMpGCCMg0tRXVzDZ2slxcuI4oxlmPajYpJydkcv4q1Jodd0mytrf7TKr+akO7aC/ITJ7AcmtTSNGmiun1PWJEuNSlGAUB2QJ/cTP8+/8+f8N+Zr3jS51ecfJbR4jX+4zcAfULnPua7mvOoR9vN1papPTtppf1vc1rfEofy6fPr+IUUUV6JiFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRSEhVJYgADJJ7VmWHiTS9Su1trSeRpJFZ4i9vJGkyjGWjZlCuORypPBz0oA1KKhu7uCws5bq8lWGCJdzux4Aqtp2t2OqPNHavKskIBkiuLeSB1Bzhtsig4ODzjHB9KAL9c38R/+SWeK/8AsC3n/oh6v2PiXStRvEtrSeRnkDNEzwSIkwXqY3ZQrjv8pPHPSqHxH/5JZ4r/AOwLef8Aoh6AOkooooAKKKKACiiigAooooAKKKKAM7xF/wAivqn/AF5zf+gGjw8CPDOlgjBFnFkH/cFN8SME8LaoT/z6Sj81IqbRv+QDYf8AXtH/AOgitf8Al38zi/5jP+3f1LtY2mkN4q1rjlVtxn/gJNbNY2lbW8S62w674QT9EqqXwT9P1R6Mdpen6o2aKKKwMwooooAKKKKAGyOI42dvuqCT9K46O3kj8EWepIubi2mN4cfxAudw+m0/pXRa/P8AZ/Dt/J0PkMAfcjA/nUunWqR6JbWsiBlFuqMrDIPy4OaykuaVvI7qNT2VLn7yX4J3/MLLV7DUJGjsrqOZ1UMyqeQK5TxdcvfSXaRlmiswkMSA8SXDnA+u0Z/GukvPD9hdrDsRrR4T+7ktSI2Udxkdqxn063XxJpukWe947V21C6Z23Ev0Qk+uea5cW5ulyd9Pv0/Df5Ee0o04uVJvmeiv087+h0Gl6bBpOmxWlsoCoPmPd27sfc1coortjGMIqMVojkCiiiqAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAMrxRHNN4R1eK1DGZ7KZUC9SShwB71h3M0Fx/wAIT9gZG3TCSIKRnyhbPuI9hlQfqK7Gqdpo+mWF1LdWOnWltcTZ82aGBUeTJydxAyeeeaFo/u/AHqvv/EwvGV1b3Ogq8E8U0NpqdsLzy3DeWFmQsGx0xwSD0FVPEiTXPie9i0/5ph4euFYJyQzMPLB784bH411sdjaRJOsVrCi3DF5gsYAlYjBLepI7mmWGlafpUTR6XY21lG7bmS2hWMMfUhQOaVtLev4qw72d/wCtHc5SWWGfTfA4sCpZriJ41UjPli3ff+ABwaT4mwa43gDxVJb6jp6WH9kXRMD2DtKV8htw8wTAZPODs4yODjnqrbR9MsryW8s9OtLe5mz5s8UCq8mTk5YDJ555rH+I/wDySzxX/wBgW8/9EPVN3u++pKVrLsrHSUUUUhlTUv7Q+xn+yPs32ncMfat2zHf7vOayP+K1/wCoD/5GroqKuM+VWsjmq4f2kubna9Gc7/xWv/UB/wDI1SRf8Jdg+d/Yue2zza3qKtVbfZX3BSw/sp83O35N3Rif8VT/ANQf/wAi0f8AFU/9Qf8A8i1t0VXtv7q+47efyRif8VT/ANQf/wAi0f8AFU/9Qf8A8i1t0Ue2/ur7g5/JHOX9l4l1HT57Oc6Usc6FGKGQEA+mafbW/ia0tIbeM6SUhRUUt5uSAMc10FFP6w7W5V9xHu83Pyq+xif8VT/1B/8AyLVW30/xLa3l1cxHSS90ys4YyYGBgYrpaKiVVyi4pJX7Gsazi9kYf/FV/wDUG/8AItNf/hLdvyf2Ln382t6iuR07q12VKvzK3KvuOe/4rL/qBf8Akaj/AIrL/qBf+Rq6Gisfqv8Afl95y2Oe/wCKy/6gX/kaj/isv+oF/wCRq6Gij6r/AH5feFjlNS0/xdqlhJZ3DaKscmNxQyg8EHHI9qtf8Vl/1Av/ACNXQ0Uvqqvfnl95V3y8t9Dnv+Ky/wCoF/5Gq1oOl3Nk13d6o8Ul/eS75WhztVQMKozzgCteiqjhoxmpuTdu7FYKKKK6hhRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAVz/j62nvPht4ltbOGSe4m0m6jiiiQs8jGFgFUDkkk4AFdBRQBzf8Awnmkf8+fiD/wnNQ/+MUf8J5pH/Pn4g/8JzUP/jFdJRQBzf8Awnmkf8+fiD/wnNQ/+MUf8J5pH/Pn4g/8JzUP/jFdJRQBzf8Awnmkf8+fiD/wnNQ/+MVHN8Q9Dt4w9xBrsSF1QM/h6/UFmYKo5h6liAB3JArqKo6xpv8Aa1jHb+b5Wy6t7jdt3Z8qZJduMjrsxntnPPSgDJ/4TzSP+fPxB/4Tmof/ABij/hPNI/58/EH/AITmof8AxiukooA5v/hPNI/58/EH/hOah/8AGKP+E80j/nz8Qf8AhOah/wDGK6SigDm/+E80j/nz8Qf+E5qH/wAYo/4TzSP+fPxB/wCE5qH/AMYrpKKAOb/4TzSP+fPxB/4Tmof/ABij/hPNI/58/EH/AITmof8AxiukooA5v/hPNI/58/EH/hOah/8AGKP+E80j/nz8Qf8AhOah/wDGK6SigDm/+E80j/nz8Qf+E5qH/wAYqOP4h6HNJKkUGuu8L7JVXw9fkxttDYP7ng7WU4PYg966iqOn6b9gvtUuPN8z+0Lpbjbtx5eIYotuc8/6rOeOuO2SAZP/AAnmkf8APn4g/wDCc1D/AOMUf8J5pH/Pn4g/8JzUP/jFdJRQBzf/AAnmkf8APn4g/wDCc1D/AOMUf8J5pH/Pn4g/8JzUP/jFdJRQBzf/AAnmkf8APn4g/wDCc1D/AOMUf8J5pH/Pn4g/8JzUP/jFdJRQBzf/AAnmkf8APn4g/wDCc1D/AOMUf8J5pH/Pn4g/8JzUP/jFdJRQBzf/AAnmkf8APn4g/wDCc1D/AOMUf8J5pH/Pn4g/8JzUP/jFdJRQBzf/AAnmkf8APn4g/wDCc1D/AOMVGPiHobXD26wa6Zo0V3jHh6/3KrEhSR5OQCVbB77T6V1FUYdN8rxDd6p5uftNrBb+Vt+75TytuznnPnYxjjb3zwAZP/CeaR/z5+IP/Cc1D/4xR/wnmkf8+fiD/wAJzUP/AIxXSUUAc3/wnmkf8+fiD/wnNQ/+MUf8J5pH/Pn4g/8ACc1D/wCMV0lFAHN/8J5pH/Pn4g/8JzUP/jFH/CeaR/z5+IP/AAnNQ/8AjFdJRQBzf/CeaR/z5+IP/Cc1D/4xR/wnmkf8+fiD/wAJzUP/AIxXSUUAc3/wnmkf8+fiD/wnNQ/+MUf8J5pH/Pn4g/8ACc1D/wCMV0lFAHN/8J5pH/Pn4g/8JzUP/jFRx/EPQ5pJUig113hfZKq+Hr8mNtobB/c8HaynB7EHvXUVR0/TfsF9qlx5vmf2hdLcbduPLxDFFtznn/VZzx1x2yQDJ/4TzSP+fPxB/wCE5qH/AMYo/wCE80j/AJ8/EH/hOah/8YrpKKAOb/4TzSP+fPxB/wCE5qH/AMYo/wCE80j/AJ8/EH/hOah/8YrpKKAOb/4TzSP+fPxB/wCE5qH/AMYo/wCE80j/AJ8/EH/hOah/8YrpKKAOb/4TzSP+fPxB/wCE5qH/AMYo/wCE80j/AJ8/EH/hOah/8YrpKKAOb/4TzSP+fPxB/wCE5qH/AMYo/wCE80j/AJ8/EH/hOah/8YrpKKAOb/4TzSP+fPxB/wCE5qH/AMYqM/EPQ1uEt2g10TSIzpGfD1/uZVIDEDyckAsuT23D1rqKozab5viG01TzcfZrWe38rb97zXibdnPGPJxjHO7tjkAyf+E80j/nz8Qf+E5qH/xij/hPNI/58/EH/hOah/8AGK6SigDm/wDhPNI/58/EH/hOah/8Yo/4TzSP+fPxB/4Tmof/ABiukooA5v8A4TzSP+fPxB/4Tmof/GK6SiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAOO1zVrqLxqdPOp6pZ2i6ek4XTrAXDFzIykt+5kIGAPQVqR619jjgs4I9T1q7aITsDFFFKsbE4LhvKVecgLgNweOCanTSp18ZS6sWj+zvYJbBcnduEjMTjGMYYd6zNc8K/bfEP9rR6ZpWq+ZbrBJb6muAm0sQ6PsfH3iCNvPHPFJbL5/r/AMAb3fy/T/gle68SNP4k0K402O9uYLi1ug1nDhWaRGjBDBmCgqQwyxA6gHkZ6TStVi1a2kkiilgkhlaGaCYAPE69VOCR6HIJBBHNYP8AYWqWl7pN7pun6TCbKG4jeyimaGJfMZSNpEZzgKcnaMnnArW0DTLjT4rua/eN7u+uWuZhDnYhKqoVScEgKqjJAycnA6VS/wA/z/yEzVooopAFFFFABRRRQAVheMru/s/DbvpF0LS8knghjmMYfZvlVSdpGDwa3azPEGmzarpiW9uyK63MExMhIGElVz0B5wpxR1Q0clrfjHUpdDsn0lxaXkbo2ojYr+ViYQtHhhxuctg46Ia6fU/ELaXJcF9G1Ke2tkDy3UKx7FXGSQGcM2B12qar674XhvdM1BNLihgu9QngmnkckBzG6HnGf4V7Dr9SazNd8I3mqX2qO9ppeorex7ba41B2ZrH5Nu1E2EY3fNkMpyec4GU27ef/AAwJI2rnxNBDqcdhaWN5f3EtqLqMWyptaMtj7zsoHryR7c8VsqSyglSpIyQeorC0/Q7m08QW19I8Rii0pLJgpO4uGBz06fr7VvVTsv683+hKv1/rRfqFFFFIYUUUUAFFFFABRRTZfM8l/JCmTadgc4BPbPtQBx83iHUR4xEsc3/Eiiu10yWPy15mZSfM3YzgOUTGcZJrZu/Ei2+szaXbaZf311DCk7rbrGFCMWA+Z3UZyvTrzxnnGMfAEU3hWSzuLq6/tGWNpHlS+nEP2knfvEe7bjfz93tUcUmtx+OtQFna2E90dKtBOZrp40V90vK4jYsM567aPL+tv8w8/wCt/wDI128X2bf2ctnaXl5JqUMksEcMag/JgMrbmAUjd3IGQRnOM7kMjSwJI8TwsygmNyCyH0O0kZHsSK5zTPC9xpuo6NN9ojmSxt7lJ3IKtJJK6uSq9AMhu/HHWumpgFFFFIAooooAKKKKACub8aX2rWVnpw0CZYrqe9EeHQMJFEbvsORxuKgZHIrpKzNY02bULnSpIWRRZ3ouJN5IyojdcDjrlhS6r5DVtfR/kczf+Lbu/wBU0F9DmMNhLPALzdGpLmXOIueVKhSTjnla3r3xMNPllN3pGopZxSiJ73bH5QyQN2N+8rkjkLUV/wCF4dtmukRQ2yx6quoXAJI3nncR15ORxwKw9V8E3+of2ij2uk3c1zcGWHU7t2aeJNwIjC7DtAAKgq+O+3kiqX6/5f8ABJf6f5/8D/hjqf7fs1m1SOffAdLUPOZAAChTcHXnleCM8cqRVuwuxf6db3YhlgE8ayCKYAOoIzggEgH8a5fxRpo1PxdpdpbTbGuomGoRhciS1R1cA+mX+UeztXYUltcb3CiiigAooooAKKKKACiiigDnLm5vbLx1ZW/9oTy2d3bXErWzpHtjKbMbSFDfxHqxq2viWzbStKvxHP5WqSRRwrtG5TIMjdzx05xmoYtM1G98RxalqsVpbJa28sEMdtO0xfzCuWYlE24CDjBzk88c5cPhzXBYaJpsv9nra6RdRSeeszl50jyB8mwBDjHG5h7iiPRPy/N3/CwPr/XT/MvyeM7aNb2UabqD2thcNBdXQSPy4SpGW5cMyjOflBOO1dGDkZHSuWl8M3j+GPEWnCWDztUnuJIWLHaokAA3cZHTnANdQo2oAewxR0QPf7/+ALRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRXDa58U9J0zWbrSbPS9d1ia1Oy8n0nTnuIrNiM4kYdDjnAzik3YaVzuaK838GeMLjTPgnoWt6lZ634guZkKutjA13cuSz8kE5xgYyT6VraL8S9O1y11PydI1y01HTbb7TLpN5YGK7dOcFEJw2SMDB61UvdbXYS1Sfc7KivLfhV8ULvxLoNpFr+k64lwsEs1xrVzp6w2ThWPSRSBkDjoOVNa1l8XtDvNTggOma7bafcusdvrNzprx2UzMcKFkP948AkAUW1sHmd5RRRSAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigArx/wAN+NtK8CnWvC3iOG/i1qTVLu4t4YbGWZr9JZC6PGVUg8MByRjFewUVLV/ut+X+Q7/5/wBfeeB2fiC/0T4K+BiNTvND0S4Eq6pqtnbedLbKCxRcbW2hm43YOMD1q/8ADQ2Mvxju5tJ1vxBrdrNoAdLzXQ4d/wB/j93uRD5fHpjO6vbaKu95OXe/4qxNvcUfT8Hc8P8ADGp2mqfCbVPhjazTQ+LILG8gazmtpI+d74O8jZhgRj5u9c5Zp4MvYbXR9X8b/Ek6qRGJfD88ssjBwR8hXydhUEdc4wK+k6KE/e5n5fgOXvK3r+IDgUUUUgCiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooA//Z)

![A close up of a map

Description automatically generated](data:image/jpeg;base64,/9j/4AAQSkZJRgABAQEAYABgAAD/4RD0RXhpZgAATU0AKgAAAAgABAE7AAIAAAAOAAAISodpAAQAAAABAAAIWJydAAEAAAAcAAAQ0OocAAcAAAgMAAAAPgAAAAAc6gAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAEtldmluIFBvbGxhcmQAAAWQAwACAAAAFAAAEKaQBAACAAAAFAAAELqSkQACAAAAAzM4AACSkgACAAAAAzM4AADqHAAHAAAIDAAACJoAAAAAHOoAAAAIAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAyMDIwOjA2OjEwIDIxOjUzOjA4ADIwMjA6MDY6MTAgMjE6NTM6MDgAAABLAGUAdgBpAG4AIABQAG8AbABsAGEAcgBkAAAA/+ELIGh0dHA6Ly9ucy5hZG9iZS5jb20veGFwLzEuMC8APD94cGFja2V0IGJlZ2luPSfvu78nIGlkPSdXNU0wTXBDZWhpSHpyZVN6TlRjemtjOWQnPz4NCjx4OnhtcG1ldGEgeG1sbnM6eD0iYWRvYmU6bnM6bWV0YS8iPjxyZGY6UkRGIHhtbG5zOnJkZj0iaHR0cDovL3d3dy53My5vcmcvMTk5OS8wMi8yMi1yZGYtc3ludGF4LW5zIyI+PHJkZjpEZXNjcmlwdGlvbiByZGY6YWJvdXQ9InV1aWQ6ZmFmNWJkZDUtYmEzZC0xMWRhLWFkMzEtZDMzZDc1MTgyZjFiIiB4bWxuczpkYz0iaHR0cDovL3B1cmwub3JnL2RjL2VsZW1lbnRzLzEuMS8iLz48cmRmOkRlc2NyaXB0aW9uIHJkZjphYm91dD0idXVpZDpmYWY1YmRkNS1iYTNkLTExZGEtYWQzMS1kMzNkNzUxODJmMWIiIHhtbG5zOnhtcD0iaHR0cDovL25zLmFkb2JlLmNvbS94YXAvMS4wLyI+PHhtcDpDcmVhdGVEYXRlPjIwMjAtMDYtMTBUMjE6NTM6MDguMzgyPC94bXA6Q3JlYXRlRGF0ZT48L3JkZjpEZXNjcmlwdGlvbj48cmRmOkRlc2NyaXB0aW9uIHJkZjphYm91dD0idXVpZDpmYWY1YmRkNS1iYTNkLTExZGEtYWQzMS1kMzNkNzUxODJmMWIiIHhtbG5zOmRjPSJodHRwOi8vcHVybC5vcmcvZGMvZWxlbWVudHMvMS4xLyI+PGRjOmNyZWF0b3I+PHJkZjpTZXEgeG1sbnM6cmRmPSJodHRwOi8vd3d3LnczLm9yZy8xOTk5LzAyLzIyLXJkZi1zeW50YXgtbnMjIj48cmRmOmxpPktldmluIFBvbGxhcmQ8L3JkZjpsaT48L3JkZjpTZXE+DQoJCQk8L2RjOmNyZWF0b3I+PC9yZGY6RGVzY3JpcHRpb24+PC9yZGY6UkRGPjwveDp4bXBtZXRhPg0KICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICA8P3hwYWNrZXQgZW5kPSd3Jz8+/9sAQwAHBQUGBQQHBgUGCAcHCAoRCwoJCQoVDxAMERgVGhkYFRgXGx4nIRsdJR0XGCIuIiUoKSssKxogLzMvKjInKisq/9sAQwEHCAgKCQoUCwsUKhwYHCoqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioq/8AAEQgB9AKwAwEiAAIRAQMRAf/EAB8AAAEFAQEBAQEBAAAAAAAAAAABAgMEBQYHCAkKC//EALUQAAIBAwMCBAMFBQQEAAABfQECAwAEEQUSITFBBhNRYQcicRQygZGhCCNCscEVUtHwJDNicoIJChYXGBkaJSYnKCkqNDU2Nzg5OkNERUZHSElKU1RVVldYWVpjZGVmZ2hpanN0dXZ3eHl6g4SFhoeIiYqSk5SVlpeYmZqio6Slpqeoqaqys7S1tre4ubrCw8TFxsfIycrS09TV1tfY2drh4uPk5ebn6Onq8fLz9PX29/j5+v/EAB8BAAMBAQEBAQEBAQEAAAAAAAABAgMEBQYHCAkKC//EALURAAIBAgQEAwQHBQQEAAECdwABAgMRBAUhMQYSQVEHYXETIjKBCBRCkaGxwQkjM1LwFWJy0QoWJDThJfEXGBkaJicoKSo1Njc4OTpDREVGR0hJSlNUVVZXWFlaY2RlZmdoaWpzdHV2d3h5eoKDhIWGh4iJipKTlJWWl5iZmqKjpKWmp6ipqrKztLW2t7i5usLDxMXGx8jJytLT1NXW19jZ2uLj5OXm5+jp6vLz9PX29/j5+v/aAAwDAQACEQMRAD8A+kaKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigArlPFXxD0vwrfxacbHVNY1OWPzhp+kWZuZljzjewGABn1NdXXl2peJLD4c/FDXtV8XC4ttL1q3tTa6kls8scbRKytCxQEqcncOMHJpdbD6M6O5+JWiWnhW31y5h1GP7TMbaDT2s3F5JMCQYxD13cH296k8MfEHS/E7XcEdpqWl6hZxCafTtUtDb3Cxno+08EHHUGuV8SeJLW51Twl8QILS+n8P6fJdQ3EhtHEkSyIFW48sjdsypGcZw2afo+s2vxC+Jcev8AhdJ5tIsNGntG1CS3eFLiWV0IRN4BbbsJJxgZod7O3n8tNPv/AFtuJW0/q+tvw/4Ox1x8b6aPhz/wmnkXX9nfYft3lbF87ZjOMbtu7/gWPeqviH4kaN4a1DTrK/t9QluNStXuLWO0tjM0m3b+7CqSxc7uOMcHJFeT3Pj7Srf9n688GyQaifENhpDWd5YrYyf6OVBUu7kBAnGc56ds16Cf+SweEf8AsXrr/wBCgpvWdlt/wJP9ELVRV9/+Cv8AMkm+KVrqHgfxDqOl6ZrVtqekw/vdNuLHbdws4+RzESQV/izk8Kao+EfiybvwA2ra/oPiKFtN0lLy7vrjT1iivCFG4wncFbJyR90Y9KfdWsl58TvHNrbruluPDdsiqO7HzwK5lPGWi6z+zrqvh+ynmGraP4cEN9azW0kbQOkYQgllCk5HYmp5vclJdvy5k/y1+RdtYrz/AD5f+Cdmfi9o5t7m9i0bxBJpdtCs7ap/ZxS1ZS6odrsRuxu3HAPyqxGcV0+t+JbLQo9OM6y3D6ldx2ltHbgM0jv35IG0AFifQVIlhb6p4VXT7yMSW91ZiGVCOCrJgj8jXl/wztdY1jxctr4ijYxeBIX0y3lc5FzOxIEv1EAQf8DPrWjS5+Xt+XX9F8zNO8Ob+r9P1b9D2OiiipKCiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooArabqFrq+lWmpafL51peQpPBJtK70dQynBAIyCOCM1Zrm/hx/ySzwp/2BbP/wBEJXSUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAHO044PYkVh+EvDY8MaK9pJdfbbq4uJbq7uzHsM8sjFi23JwOgAycACtyigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigDm/hx/ySzwp/2BbP8A9EJXSVzfw4/5JZ4U/wCwLZ/+iErpKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKiu7qKys5bq4bbFChdz6ACjcTairslorM0HXIdf0z7ZAjRbXKPG/VCPX8CD+NZSahrniLzJdEe3stNJMcdxMpaSTkguo7e2a0VOV2npY5JYynyRlC8ubZLr/l8zpBcQm4NuJo/OC7jHuG7HrjrjmpK5K+8MJounxajoas2oWJMru7EtdL/GrfUU+bX73xAwtfCjxp+5Es93IMiIkZEYH9719Oav2SesXoY/XZU24Vo2l0S1v6barr0W+x1VFZehayNWtWWZPIvrc7Lq3PWNvX6HqDWpWMouLszupVI1YKcHowooopGgUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABWDF4ttJZ4SLS7WxuLj7NDqDBPJkkyQAPm3gFgVBKgE455Gd1wWRgDgkYz6V5xBdQv4H0nw1HKg1qG6t4ZLIN+9jMcwZnZRyF2qW3dMEc8ihb29A6Ho7MqIWdgqqMkk4AFZmjeItO1+S9GlTC4js5RE8yENG5KhvlIPI5xn1BqxFq1jNqF5Yxzj7RYqj3ClSBGGBKnJGDwD0Nc34U1nS7zxX4jjs9StJ3nuo3iWKdWMiiBAWUA8gEEZFC3+QPY07TxPJfyyix0LUp4Yrh7dpw1uqbkYqxw0obGR6Vu1wF5PosUkcngzVZG1eW/VzYw30kiuWk/feZAWIUYLknaMEDkV1uowa5LcKdJ1HT7WHZhkubB52LZPO5ZkAGMcY9eeeBfCmD3M34cf8ks8Kf9gWz/8ARCV0lc38OP8AklnhT/sC2f8A6ISukoAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigArlp4pfFOtXlnJcMmkWMiRyRRqP9IkHzFS3oOARWtr2rHSNPDwx+ddTuIbaH+/Ien4dzS6BpR0jSI7eV/NnZmlnk/vyMck/0/Cto+5Hn69P8zz66WIqqh9lay/RP13a7LXc5vX5Lnw3capJbQO1tq0QWJoxxFPjbg+mRzn1FdXpVium6Ta2SdIIlQn1IHJ/E81ZdEkXbIquuQcMMjIORTqUqnNFIujhFRqymno9l2u2397CqmnaXaaVDJFYwiJZJGlYDuxP+R+FW6Kzu7WOtwi5KTWqMXWfD4vpvt+nzvZamiYjnjOA/orjowqbQNXOraeTOnk3kDGK5hP8Djr+B6itSsPUvDstzqT3+majLp08sXlzGNAwlx90nPcevpWkZKS5ZP0OKpSnRqe2oxvf4l389dLr8V8jcorF0HWJrppdO1VRFqlqP3qjpKvaRfUH9DW1USi4uzOqjVjWgpx/4byfmgoooqTUKKKKACiiigAooooAKKKKACiiigAooqO4t4ru2kguF3xSLtZckZH4UDVr6klFYn/CH6F/z4/+Rn/+Ko/4Q/Qv+fH/AMjP/wDFVF59l9//AADo5cP/ADP/AMBX/wAkbdFYn/CH6F/z4/8AkZ//AIqren6Hp2lTNLYW/lO67WO9myPxJoTnfVfj/wAAmUaKXuyd/Rf/ACTMfWfGi6L440vQriwZra/iLPf+cAtu5cJGrKRyGYhQQepAxzTLPxuL34l3PhWDTyYILR5jqHncNKjRh4gmOwlT5s9cjHBqDX/Dv/CQeMp7a6imWzuNDeEXKpxFL56shDdN6kBh9M1l/wDCK3+n+KbG1sXnlceHL6GTUzEVV7uWaJi7EcKzNubGfXHSqg/dTf8Ae/Dmt+XzuvMw3k16fpf8/wAGdnbeJdCvdUn02z1rTri/tgxntYrtGli2nDbkByMHg5HFSrrmkvpkOopqlk1jcMqQ3QuEMUjM21Qr5wSTwMHk8V5v4X02MN4dsdTuvFy3WkuCmnz6PEltbyJGyN/pEdsqsmCwBWU7gR1JxTtI068s/GsfiSbRrweH768k+x2HkyGTTpnAU3bQYyvmkPkYygfccb5MVbW39f0+m/8Akr9f6/pdf6v6HFrEC22oXN/c2EFvYyussqXYZYlUAkykhRGwByVOcDHPNR23inw/e6VPqlnrum3Gn2xInu4ryNoojgH5nBwvUdT3rg7rSNRQ3t7Lpt1PZ2vi46hPbLAzPPAIgBIiYzIFcq4AyTs4yQBWl4huU1waTrGn6RqUtjpmrxz38UumSxTToImCusLqJJPLd0bhSflO3JGKmLvFN+Xyulr+PltuPq/n+Den4fidrpuqafrNit5pF9bX9q5IWe1mWVGIODhlJHBq1XIeFoprzxnr2vW9pc2emX0NtFEt1bvbvPLHvDymJwGXhkTLAE7PQAnr6pgFFFFIAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigDm/hx/ySzwp/2BbP/wBEJXSVzfw4/wCSWeFP+wLZ/wDohK6SgAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAhvLlbOxnuXGVhjaQjOOAM/wBK5e1+INjJZRSXVrcR3B5mhjQv5SYBEhP93BBrS8ZXBt/B+oFfvPGIgPXcQv8AWtO0sILSBFSJN4iWJnCjLBRgAnvW0eRQvJXPNrPETxHs6M1FJJu6vu3bt2MGzmTxJ4pTULfL6bpqMkMhGBLM33iPUAcZ9a6emQQRW0Kw28aRRIMKiLgD8KfUTkpPTZHVh6LpRbk7ybu35/5JaIKKKKg6AooooAKKKKAM3VdA07WtjX0JMsYIjlRijp9CP61n6ZqNzpV/HomtyeYzDFneHpOo/hb0cfr/AD6KqOsaTBrOnPaXGVyQ0ci/ejcdGB9RWsZ6cstvyOKth2pe2oq0/wD0ryf6Pp6XReornR4H0hl33X2q5uupupLl/Mz68HH6VHcaXrGiQ/atI1K6v44jueyuyJDIvcK2Mg+lHJB6KQniMRBc1Slp5O7+6y/C501FcveeLob2wjg8PuZtSu/kjiKnMHZmcdtv+eKlGneJdNVXs9Vj1MAfPBeRhM+u1x0/Gj2TS9529RfXoSf7qLmlu1Zpfjr8rnR0Viab4otLucWd8p07UQdrWtwcEnttPRge2K26iUXF2Z1Uq1OtHmpu/wDXXsFFFYl34w0O0l8pr5ZpB1S3UyEeuduaIxlL4VcKtalRV6kkvV2Nuiud/wCE20tzutY726h7zQ2rlB9Sea1NM1rTtYjZtOukm2/eUcMv1B5FVKnOKu0ZU8Xh6suWE036/l3L1FFFZnUFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQBzfw4/5JZ4U/wCwLZ/+iErpK5v4cf8AJLPCn/YFs/8A0QldJQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAc94s/fnSbFeTc6hGWX1Rcs38hXQ1z0n+nfEKFesem2bP9JJDj/0EV0Naz0jFf1qcWG96rVqedvuX+dwooorI7QooooAKKKKACiiigAooooAKKKKAIYrS3huJZ4YI0lmx5kiqAz49T3qaiijcSSSsihrGkW2s6e9tcrhiMxygfNG3Zge3NZUXhrULyNf7e1y5nwuPKtT5KdO5HLfpXSUVpGpKKsjmqYSjVnzzWvq9fVdfmc5L4MtwpFlqWoWquNsqrcFhIvcc9/f3rbstPtNOt1hsbeOCNQBhFxn6nvViilKpKSs2OlhaFGXNTik/6+75BUaQRRzPLHEiySY3uFALY6ZPfrUlFQdFkwooooGFFFQ3dtHe2klvNu8uQbW2nBx9e1J3toBNRWB/whekel1/4FP/AI0f8IXpHpdf+BT/AONcvPif5F/4F/8Aai1N+isD/hC9I9Lr/wACn/xq5pmgWOk3DTWYm3su0+ZKz8Zz3PtVRnXcvegkvX/gBqUdW8R30XiJdC0DTIb7UBbC7mN3dm2hiiLFB8ypIxYspwAuMAkkcA2rPWLiCwMviqGy0aXz/KTF8JYpeMgq7KhOeeCoPynqME4HjCDQ5/ENlJ4h0LWne3hZrPVtIjumdCeHiLWp81eCD8w2nJ5yK5O+S8sfDenSa/8A2peaUviq3l05NShe4vPs23IDoFMjHfvKhgZANueRiuhf185Jffr+Y5aXa7fkr/n+iPTrXxNoN9pM+qWWt6dcafbkia7hu43iiwATucHAwCDye9JZeJdH1iwurnQNX0zUltlJdoL1HjQ4JAd13bRx1x07VxWoXGrXt14l8QeH9GnNvLZ2lrHHf2EitcOkjmWUWzhXbYkgwCAXKYGeM17C0vbrxJq95FL4g1OGfw9JAt3qumLas8gc4jVVhiYkZJG5f4vlJ5qZtpO3Z/gm/wAbfiPTT1X6flf8DvJfEukWOn291rGraZYia3E+571PLK/KCyu23cmXUbsD7y9M1oWd7a6jZxXen3MN1bTLujmgkDo49Qw4IrgND0e6XxN4FnutOmC2PhuaKSSSA4t5SLcbSSPlYgOMcHAb3ro/BtpNZWurJNbvbh9XupI1dCu5WfIYD0OSc961kkpW9fwdv+CZxbcU35firnRUUUVJYUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAc38OP+SWeFP+wLZ/8AohK6Sub+HH/JLPCn/YFs/wD0QldJQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAc94f/AHviTxFc9d1xHD/3wmP610Nc94O+ex1C5P8Ay86jPIPpnH9K6Gtavx2OLA/7upd7v723+oUUUVkdoUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFVL/S7PVPs326HzfstwtzD8xXbIudrcEZxk8HirdFAeQUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAc38OP+SWeFP8AsC2f/ohK6Sub+HH/ACSzwp/2BbP/ANEJXSUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFMmkEMEkrdEUsfwFPrz5tM1bW9Z1Wewu2jtLm9axulz0iQD5h+RX8a1pwU73drHDi8TKgoqEHJvovTf77HR+CYjF4OsN33nVpCfXcxP9a3qjt4IrW2jgt0CRRKERR2A4FSVE5c0nLub4el7GjCn/ACpL7kFFFFSbhRRRQAUUUUAFFFFABRRRQAUUUjOqDLsFHqTigBaKqy6pp8AzNfW0YHUvMo/rWLd61Pq+pQ6f4YvIchTLcXQUSLGvQL6ZJrnqYinTW932W7E2kdJRXP6RdanD4iudL1a8S7ItlnjdYRHxuIPT8K6Cro1VVjzJW6a+QJ3CiiitRhRTfMTzPL3Lvxu255x64p1ABRRRQAUUVDd2y3lpJbyM6LIuC0bbWH0PY0CldLQmornv+EOs/wDoIap/4GNR/wAIdZ/9BDVP/Axq15af834HH7XFf8+1/wCBf8A6Giue/wCEOs/+ghqn/gY1XdM0GDSrh5Ybq8mLLtK3E5cDnORnvxScYW0f4FwqYhySlTSX+L/gDf7d/wCKz/sD7P8A8uH23z9//TTZt24/HOfwqC08U20us63ZXvlWUWk3EFv9omnAWVpY0ZeoGDlwoGTk/XFZusNJonxBt9fuLS8uNOm01rJ3srWS5eKQSB13Rxhn2kZ+YAgEc4yK5fWtFvNb0XxJqH9lXMltrOs6c8NrPasJJIIngR3eIjcqna5wwB2jJArKOqV/P/0qy/A618TT8rfcr/jc9CtPEmkatY3dxoWrabqItVPmNDeIyRtgkB2XdsHHXHTtWZJ8Q/DtlrVnpOq6rp9peXVkt2rG9j8khiAFV2Klic5X5RkDPtVXXdPuZPF2oT29pK0cvhyaDzEiJDv5mVTIHJ5OB15NZGkLcaBq3hW+1HT9QFv/AMI2ti5gsZZ2hn3RHY6xqzJwDyQBwckUoPma7f8A7f6pff8AfEm1dL+vh/JN/cdvdeI9EsdWh0q91nT7fUZ8eVZy3SJNJk4G1CcnJBAwO1QeIvFWi+GrVjq2sadp87xO8Ed7dJEZSo7BiCecdPWvObnRZoD4i0nxBf8AiyFdSvp5/K0rSY7qG7ikb92RMLZyjBQqYd1K7BjC7TW7ul0DXPFa3um6neXOrBGsJrayecSxrbqgiMigrGQ4c4cqPnyDyaJX5Lrt/S9f6saKynbz/Xf0Ox8PahLq3hnTNRuVRZru0inkWMEKGZAxAyScZPrWCPH8R0OS6WwaTUJNRuNNsdPik3PdyxSMmc4+VcLuZiMIuSScc6/hCCW28E6JBcxPDNFp8CSRyKVZGEaggg8gg9q860XwtrGhTXvjDTbe8uNRh1S/WbTLhf8Aj4snuGbEAYDY3AkUg4cnBJBXbpOyqtdNfzX9fiZ07ukn10/J/wBfgek2+rSrcyw6qdNtWgtEuJ1jvzI0RO7cSrIuIxt4c4zhuBinWPiTQ9T1B7DTdZ0+7vI08x7a3ukkkVePmKg5A+Yc+49a46/hutT1/wAS6ha2F79nvfC8UcBltZI2eTdcHy9rAHeNy5XGRkcc1KmkXMFr8OY7Wxkg+xNtnEcBAt1NjKp3AD5Rv2jnHOO9TG7vfp/nL/IqWjsv60T/AFOol8V+HYdWOlTa9pkeoqyobN7yMTBmICjZnOTuXAxzketZ/jTxtpPhLRL97jV9MttUjspLi0tLy5RGmYK20BCwZgWGOOvSuE85tO8B+H/Dd1pWoxara6zYfbHeydYmkF4jPMJyNkm8/N8rMx3ZIGG26GrfaNN0bx5pU2kapdanq5upbWS20+SZLmN4NsY81QUUrgrtZlPHAO4ZUvgbXn+Ftvv/AANI2VRJ91+b3+78TurrxRo+kWFlP4g1fT9La7jDR/a7lIQ5wCQu4jOMj8xVnUtc0nRow+r6pZWCFC4a6uEiG0EAtliOMsoz6sPWuLRz4c8XXeo61pWoXdvfaRa29tLaWEt2U8vf5kDLGrFMl1bJAVuecrUPhbw5qGnat4PXU9PZTYaXfA5UOtmXliMcW8ZUMEyvB6KcZFW1eVl5/r/l/XXKOyv2/wAv6/rTsV8WeHHlso01/S2fUBmzUXsZNyM4zGM/PyMcZ5rXrza60CVfh34zt7fSpEubzUbqeOOO3Iec7wUcADLdBg+1ekD7oz6VK1SffX7w+00LRRRQMKKKKACiiigAooooAKKKKACiiigDm/hx/wAks8Kf9gWz/wDRCV0lc38OP+SWeFP+wLZ/+iErpKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAEd1jjZ3OFUEk+grB8Eo3/AAi0E8gw91JJO31Zz/TFWvFF39h8LajPnaRAyqfQt8o/UirOkWn2DRbO0IwYYEQ/UDn9a12p+r/L/hzifvYxf3Yv/wAmat/6Sy5RRRWR2hRRRQAUUUUAFFFZOr69Fpzra20ZvNRl4itYzz9W/uj3NZ1KkKUeabshXsa1ZV94m0jT22T3sbS5wIov3jk+mFzWPNp3iTUmi07WJIGsZmEs81v8pVRyYvzxz6V0VjpNhpq4sLOGD1KIAT9T1Nc0atas/wB3HlXeW/3frcV2zKGta1f/APIK0Rooz0m1B/L/APHBk1HLaeKo1F2NQtZpYzuNjHDtjkHcbzzn0rpKKbwzkvfnJv1t+X63Cxz32bxLqf8Ax9XcGkwn/lnbDzJcehY8D6inp4L0bO+7hlvZu8tzMzs364/St6in9UpPWouZ+ev4bL5IOVdTKi8MaHCQU0q1yDn5ow386v29nbWm77LbxQbvveWgXP5VNRW0KNOHwxS+Q7IwL7/R/Hmly/8AP1bTQH/gOHrfrB8T2l/K+m3OkwrNdW1zkK5woUqQST6dKi8/xPphEt3Db6rC330tRskj9xn7w/WuSNX2NSonF2ve6V1sv1vsK9mdHVbUb1NO06e7l+7ChbGep7D8TxWR/wAJlp0X/IQgvrD1+02rD+Waz9b8TaNqkNlbW9/E0b3cZm3ZUCMcnOQPatHjMPbSav6m1HklUSk9OpYh8NyXOlC9lmlTWpB5wn3kbG6hMdNvbFaujaudTgKy280FxEAJleMqA3fB79KsQ6rp9x/x739tL/uTK38jVsHIyORW0FHeDNqmJdWLU1ft5eXp5dAooorY5QooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAMK28GaLa6mt+kV3LLHM08aXGoXE0MUjE5dIncoh+ZsFVGMnGM1u0UUAFFFFABRRRQAUUUUAFFFFABRXKfE/UdU0n4Y67faAXW/htS0boPmjGQGce4XcfwryW3fR9EtvipHbapM1h/ZVi9pcS3ryvJK9u7KyuzElmcg8Hr0pdJPsi4xvb+uqX6n0LRWT4VN83g/Rzq4IvzYwm5z18zYN365rWq5R5ZNGUXzJMKKKKkoKKKKAOb+HH/JLPCn/YFs//AEQldJXN/Dj/AJJZ4U/7Atn/AOiErpKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAOe8W/wCkLpenDn7XfRhx6ovzN/IV0Nc9df6X8QLCLqtjZyXB+rnZ/IGuhrWekYo4sP71WrPzS+SX+bYUUUVkdoUUUUAFFFUdbufseg31xnBjgcj64OP1qZyUIuT6AYdg+t+J7Nbo38enWErNsW1Q+c6hiOWP3enUVt6Xoljo6N9jixI/MkzndI/1Y80zw9bfY/DenwdCtuhb6kZP6mtKuTDUEoxqT1nbd6+tu3yJS6hRRRXaUFFFFABRRWNf+JrOx1BLYhpQGAuJE5WAHgFj9ccVMpKKuzSnSnVdoK5s1zniDWb62vhDpQDC0j+0XYwDlMj5eehxk/StjVL5dN0m4vGwREhZR6nsPzxVbQ9M+yacWu/3l1d/vLlnH3mI+79BnGKmd5PlRvQ5Ka9rNXWyXfv9y/GxowTJcW8c0Lbo5FDKfUEZrGm1i7v7trXw/HFIIjia7mz5an+6MfeNQS6JqttDJY6XeRrp8zYxJnzLdSeQh7jr1rds7OGws47a1QJFGMKB/P60velpsNqjS95Pmvt5evn5GE1xqWi6lbz6xfLcWl0TE5VAiQN1U/Q8jJp17Fbal4wsoWjimS3tpJnBAYHdhRn1remhiuIWinjWSNhhlcZBqtYaRYaWG+wWqQlurDlj7ZPOKTp30eqH7em1zNe9ZrRK2vX7itN4Y0Of/WaVaj/cjC/yxUumaFp2jNK2m2/kebjeA7EHHsScVoUULD0VLnUFfvZHDZBRRRWwwooqO4h+0W7xeZJFvXG+NsMvuD601uBJRWN/wjv/AFGNW/8AAr/61H/CO/8AUY1b/wACv/rVtyUv5/wNOWPc2aKxv+Ed/wCoxq3/AIFf/Wq1YaX9glZ/t17c7lxtuJt4HuBjrSlCmlpL8BNRtoxv9tW//CTf2Hsl+1fY/tm/A2bN+zGc5zn2/Gs+78baVYPrC3YuE/siaC3lxFvMskyqY1jCkliS6ryByfTmsvWNQs/D3xOt9W126h0/TbjSWtUvLlxHCJhKH2M5+VWK5IBPODjpWDKNF1az8Z6rq8lz/wAI7qN9bGDU7Ld8nlQxj7SjrnCpIufMGVG0k/KDWC1ivn/6Vb8iV8TT8rfcv1udxpviG8vdRS0vfDGsaWJEZlnujbvHkY+UmGZypOeMgA4POa26868Pa1O/jKwsPD/jM+MNKeBxfblt5jZbQdjmeBVUFj8u1wWOCR0Nei1RKCiiikMKKKKACiiigAooooAKKKKACiiigAooooAKKKKAKOsnUxpE50GO0lvwB5Ud6zLE/I3AlQSMrnBwecV5lap4L0HWFv8Axj8OofDF8r7xf/ZEubMNnhhNGCqHvl1Q13nijX4dN066gtdc0bTNTSFZ1OqTAJFGZAhkZdwO3JwDkDdgZrgUi8B6oyzeOvidp/iaTgm2l1eC3swQe1vG4B/4GWoV73Q2vd1PWLa5gvbWK6s5o57eZBJFLEwZXUjIYEcEEd6gXWNMfU205NRtGvl+9aidTKOM/cznpz0p1gbP+yLc6N9nNl5C/Zfs23yvL2/Ls28bcYxjjFcLAAPhXol1GN10L22mD/xec1wA5+p3OD9TT+1b+tRdD0So47iGWWWKKVHkhIWRFYEoSMgEduCD+NPZQ6FTnDDBwSD+Y6VyvhGzisNc8T29t5hRb2MgyyvKxJgQnLOSTye5qQN8axph1M6aNRtDfDra+evm9M/cznpz06VcrzqIf8WlsbtRm8F7FPu/i883QDH1zksPpXZ6jql3ZXCx22hahqKFNxltpLdVByfl/eSoc8Z6Y5HPWn0Azfhx/wAks8Kf9gWz/wDRCV0lc38OP+SWeFP+wLZ/+iErpKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAOe0T/AEnxXr12eQjxWye21ct+proa57wb+8029uzz9rv5pgfbdj/2WuhrWr8duxxYHXDqX81397b/AFCoby8gsLOS6u5BHDEu5mPas3Wdbeznj0/TYhdanOMxxE/LGv8Afc9gP1qC18KwvKl3rk8mpXgbfmRiI0PoqDjH1rz51pSk4UVdrd9F/m/JfNo7L9iCCHWPEkbXM95NpWnyn91bRIBK6f3i/VSfQU9PCtxb5hsNdvbazB3JAuGKt/vHnHtXR0Ulg6b1ndvvd/o9F5LQXKjlEvvEsN22jJDHPcK28ajMuIzD2JA/i7YH/wBen6ppHiXVNPks577T/JmwH2xMpxkHg8+ldRRU/U1KLjOcmn59O3n6vUOUQAKoCjAAwBS0UV3FBRRRQAUVlahq80OopYabZ/bLkx+Y4MgRY1zgEn+lRQ6+YLp7bW4UsZdnmRnzNySL3weOR6VHPFOx0LDVXHmS891f7tyG7R9b8QvZrNNFZ2UeZjC5QvI3Rcj0HNaUOjWEGmyWMVuqwSKQ47t7k9SfeqnhaNv7IN5KMS30r3Df8CPH6AVs1MIprmfU0r1JQl7KL0j+a6/ecrBp2q3TQ6TeqPsNlMGa4Y83CDlFx/P6fn1VFFVGCiZVqzqtXVvTv1fzCiiirMAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooA8y+MHi/w54S0W8uZoNFufEr2sa29vqFsJGnhMwyp6EqCGbGcZXNZmt+N/hJaeHL640lfC91qMds7Wtutij+ZNtOxdoXJy2BXcfEfUdL0j4e6tqOuabFqlpbwhmtJgNszbhsU5Bx8xXntXns/j/4l21x4giOi+FUfw/aRXlzH5s5Zo3QuNmByVCnOccjjNL7LTNOzS/rQ9W8MyzzeE9JlvLKOwuHs4mltI02rAxQZQDsAeMe1NTwxpKagt4ts+9JTOkZnkMKSHOXWItsDck5C5ySepqxod5LqHh/T725a3ea4to5Xa1LGIllBJTdzt54zzir1aT+NmMfhRTi0myh1C8vooilzeqiTyCRsuFBC9+MAnpiquneGtN0q+ku7L7YJpjmQy388okOAMsHcgnAAyRkYrWoqCjJHhnSRqAvPsz+Ys3niMzyeSJP74i3bA3JOduc89ea1qKKPIDm/hx/ySzwp/wBgWz/9EJXSVzfw4/5JZ4U/7Atn/wCiErpKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKgvrj7Jp1zcdPJiZ/wAgTU9YHjKZxoP2KA4m1CZLVP8AgR5/QGrhHmkkc+Jqeyoyn2X/AAxN4Qt/s3hDTU/vQiT/AL6O7+tP1zWjp4jtLKP7RqVzxbwDt/tt6KKZrV+dF0qC105A13Ni3s4vfGM/QDmpNH0OPTC9xNK93fzAedcynLN7D0X2rjr1Z1ajp0tH1fa/6/lv6ujT9lSjSXRJfchdE0YaXFJLcSm5v7g77i4YcsfQeijsK1KKK2p0404qMdjfYKKKKsAoorK1rXbTSbeVXmT7UIy0cPdj0H60pSUVdl06cqklGCuzVBz0rL1+/nsrGNbIqLu4mSGHcMjJPJI9MA1jRWs/haGyvpJpngk+W/RmLBWbneB7Hg+tXbR31/XIdQRGXT7IMIS4IMrsMFsHsBWTm2uXZnbHDxpy9pfmgr692unzf4alzStVluLmax1GJIL6DBKocrIp6Oue38qs6pfrpelz3jruES5C56nOAPzIqrrenS3KR3unkLf2h3RH++O6H2NZl/qcOu2ml28HDXN2vnwn7yBOXU/lQ5OKae/QUKMKso1Evd+0u3V/Jrb7jU0TTpbOGW5vW8y+u2Ek7enoo9gKu3lja6hD5V7AkyejjOPp6VPRWiikrHLKrOU+e+oyGJIIUhiXbHGoVV9ABgU+iiqMm76sKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooA4jx7rngXUdL1Dwn4s8SWVgZ0VZ4jcpHLH0dSNwOD0IyK43xbe+CPEGrXF9pHxQ0/RTqNkNP1NI2imF1AM4A3H5GwzDcM8HpXquuW8/8AZs8+k6ZY32o4Hlx3beWj8gHLhWI4z2PSuD13W/G3h/Q7rVr3wP4de2tI/Nm8nUWZlQfebHk8gDJPfAOM1OnUrXp/Wx6BoEOn2/hzTodFkWTTo7WNLV0bcGiCgIQe/GOa0Kp6PcNd6HY3MiQI01ukhW3ffGCVBwrYGV9DgcVcrSd+Z3M425VYKKKKkoKKKKAOb+HH/JLPCn/YFs//AEQldJXN/Dj/AJJZ4U/7Atn/AOiErpKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAK5vWpom8Y6Qk7qkNpDNdylzgDgKp/Amukrim06DxD8SLx7pFmtNNgSMqTlWc5IB+mW/EU+aUYylBXdvz0/U4cZqoU/wCaS/D3v0NLQo31jUpfEN0pEbAxWEbD7kfd/q38q6OkRFjRUjUKqjCqowAPSlrCjS9lCz1e7fd/1+B2pWCiiithhRRTZHEUbSP91QWP0FADiQASeAOprn/DNrHd2DardostzeyFyzqDtUEhQPQYFNsfEDap4Svr2RVjmhjkDKmcZ25H8xWnoUPkeH7CM8EQIT9SMmsk1OSaO6UJ0Kc4y0d0vuu/8i+QD1GaKKK1OEKpDR7BdTGoJbql0M/vF4zkYOR0J96u0Umk9yozlG/K7XCiiimSFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFAFHWU1GTSJ10W6t7S9wDHPcxGSNORnKgjPGe4rzHVp9a1zS7rSNT+KXhG3tbuNoLhra2US7GGGA3TkAkZGccV1nxZtGvvhL4igS7SzJs2Yyu+1cKQxUnsGA2/jXIaN4s+EjaLZvF4bigV4Vbyx4alfaSORuWEhvqCc+tTo7oq7STXn+h6hoVnZad4d06y0qQS2NvaxxW8iuGDxqoCnI4OQBzV+quly2k+kWk2mRiOzkhRoEERi2oQNo2EArxjggYq1WkruTuRHRaBRRRUjCiiigDm/hx/wAks8Kf9gWz/wDRCV0lc38OP+SWeFP+wLZ/+iErpKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAwfG18bDwleOjlJJQIkIOOWOD+maZ4J0ldL8NxMYwk11++k9efuj8Bj9aoeNE/tXWdE0NeVmmM0wB6Io6/lu/KuwACqAowBwAO1KXNzJX0S/F/5W/E8+lFVMVOq18Nor7rv80vkFFFFM9AKKKKACs3xFcfZvDd/J0/cso+rfL/WtKsLxK32v7JpEfzSXkylx/djU7mJ/KoqO0WdGGjzVo32vd+i1Zg6zaT6HYtFDGTDqNtFA2ONkq4HP1XP413MaCKJY1+6qhR+FKyq4wyhhnPIpaUKfK20VXxLrQimtVe777fkkFFFFaHKFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAcn8UNFufEXwz1rStPs5L26uYQsMEcioWcMCpyxAwCATk8gEVm2Xi/wAax2Ful38NL8TrGok8rUrTZuA525k6ZrsdX0+TVdKms4b+6055QMXVmyrLHgg/KWBHOMcg8GvPPF/g/W9D8Harq2n/ABB8StcWNs9yq3E8Gxwg3FTiIEZAIB7Ejg9KV+VNvYpLmsl/V7HpNlNNc6fbz3Vs1pNJGryW7sGaJiMlSVyCR0yOKnrK8L3a3/hHSLyOW4nS4soZRLdEGVwyA5fAA3c84A5rVq5K0miIu6TCiiipGFFFFAHN/Dj/AJJZ4U/7Atn/AOiErpK5v4cf8ks8Kf8AYFs//RCV0lABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUVheLLt7TT7RopGjLXcYJU445J+o4rdrSVNxgp97/gU42in3CiiisyQooooAKRmCqWYgKBkknpS1zfj3UDYeE51Q4kuSIV+h5b9AfzqJycYtrf9ehFSpGnBzlslcq+G7hde8XaprK5MECLaWxI/h6k/mM/8CrrqyfDGlJo3h62tk++y+ZK2PvO3J/w+grWrR9r3sc+DhKNFOe8rt/N3t8tvkFFFFI6wooooAzdd1CWwsF+yqGubiVYIM9A7dCfpik0vRItOc3EkstzeOgWSeVyxPcgegz2qvrX73XdEt+xmeU/8AXP9a26zSvJt9DrlJ06MYx+1dv77foFFFFaHIFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFAHMfEe+0zTPh5q97rkE1xZQxBnhguGgeQ7htUOpBXLYGQa5TXfhv4S0vwXda1eaBeX7Wdr9rl0+fW7p0bau5l+ZyGxg9VwcV3/iTQLPxT4bvtE1MMbW9iMblDhl9GHuDgj6Vxd54K8d6ror+HdV8YWLaRJH5E9zBppW8nh6FSxcopI4LBffFLXW2/8AX9fkUmtL7dfwO50S+tdT0CwvtPUJaXNtHLCoGAqMoKjA9jV6szTJbKxuv+EdsYnjGnWUDKMDaI2LogBznI8o549K06uVuZ22Iimkk9woooqRhRRRQBzfw4/5JZ4U/wCwLZ/+iErpK5v4cf8AJLPCn/YFs/8A0QldJQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAcj8QJCtlZIucmVn4HoP/AK9darBlDDoRkVynjJfOurdM/wCqtbiT/wAdArpbJ/M0+3fOd0SnP4V3Vl/s1P5/mbz/AIUfmT0UUVwmAUUUUAFef+P2Oo6rb2CE7YFTdju8jhQP++QTXoFefL/xMb60vTydR1sSRn+9DCCF/rWMverU6fnd+kdfzscOYa4aUe+n36fqegjjpRRRWx3BRRRQAUUVm6zrUGixQSXCsyyybCF6qMElsd8Um1FXZcISqSUYK7K0v77xzbr1FvZNJ9Czbf5CtusLQm/tHUr/AFgZ8qVhBb+6J1b8TW7UU9U33NsT7slD+VJfPd/iwooorQ5gooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKK5X4mi8Pw11n+zLy8srwQgwTWKyGUPuGAPLBbBPBI6Amsyz+LmjSWMD3Wl+IYJmjUyRHRLlijY5GQmDg9xQNrRM3bT/kpGrf8AYKsv/RtzXQVUsHtb2GLVLeAo93BGd8kJjlKcsqsCNwxubg9CTVuqluDfM7+n5JBRRRUiCiiigDm/hx/ySzwp/wBgWz/9EJXSVzfw4/5JZ4U/7Atn/wCiErpKACiiigAooooAKKKKACiiigAooooAKKKKACiimTEiCQjghTg/hQByMupW+u6hdzWhfy00mZTuGCrFsdPpzXR6K/maDYNnJNvHk++0VkeG9Mto/Di3aRbLi4t2V3yeRk44/KtHwy+/w1YnOf3WPyOP6V6OK5OVxhtF2/M6atrNR6M1KKKK845goorN8Q6o2i6Dc6giLI0IXCseCSwH9acU5NJGdSpGlB1J7JXfyG+Jr06f4Zv7hDhliKqc9C3yg/maxo7JbXxF4a05BxY2ksj+5Khc/nk1Z8TTC+0fS4V+7qF5ApAP8J+Y/wAqfb/6R8RLyTPFrYRw49Czlv6VhR1xVSX8sbffe/6HHimpqEV1lH8HzfodDRRRWx6AUUUUAFYcgGoeM1RgGi0+2JII/jk4x/3yK3OnWsPwx/pEF5qTdb25Z1P+wvyqP0NZy1aR00fdhOp5W+//AIFzYggitYFht41jjQYVVHAqSiitDnbbd2FFFVNQ1Wz0yNWvZhHuOFUAlj9AOaqMXJ2irsEm3ZFuisceK9HaQAXXyn/loY2Cg+mcdajl1qbU5Ps3h4CTn95dup8uMe394+1a/V6v2lb10L9nPqrG5RWN/Z2uJyuuq57q9ouP0NNa08QyjypNQtIoz96WKI+Z+APFHso/zr8f8g5F/MvxNS4v7S0YLdXUEBbkCSQLn86bHqdjMCYb23kC9SsqnH61UtfDum26sZIBdSv9+W5HmM35/wBKkk0DSZWUtp1tlem2MD+VO1Da7/APc8xs/iLSLZtsuoQ59Ebd/LNRr4p0VmAF/Hk+oI/pV+2sbWzGLS2hh9fLQLn8qkeGKTPmRo+eu5Qc0r0Oz+9f5B7nZlY6vpwjVzf2oRujGZQD+tSJqFlKwEd3A5PQLKp/rUA0PSg7P/Z1qS3XMKn9MUkmgaTKMNp1sP8AdjC/yo/cef4B7nmaFFYp8NQwHOl3d1Yn+7HKWQ/VWzS/2PqX3v7euPM9fKTb/wB80ezpvaf3p/pcOWPRmzRWL9n8RW/+rvbO8HpNCYz/AOO0v2/XYv8AWaNHMO5hugP0Io9i38Mk/nb87Bydmv69TZorF/4SMQf8hLTry09XMe9B/wACWtCx1Sy1JSbG5Sbb1CnkfUdamVGpFXa0/D7xOEkrtFqiiisiAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooA8a+J1l490O58T6n4Ttb3UbfWYLQRSWNyRNp7xNhtsXVg4PO3nk54FdKvjL4gBBn4WXJ46nXbT/ABrT+Kdpql98Ldet9BWR757U7EizvZcguq45yV3AVynij4h+CNR+E95pul6rb3FzdWBtbPTIGzdCYrtjTyh8ysGx1AxildqLW9v6/r+r38TV9P6R6hYTXFzpttPe2hs7mSJXltjIHMLEZKbhwcHjI4NWKy/Dkd7a+EdLj1c7r6KyiW5PrIEG79c1ykFzdp4U0zxU99dte3NzC80ZuHMLRyyhPLEWdgwrDBAzlc5OTm5Jc7S7mUXeKZ39FZXijUpdH8KalqFtjzre2d48jIDY4OPrWSsMugeINDjivry4XUfMguVubl5g7CMuHAYkIcqRhQBhunAxPUp7HV0VwcCNpl5JJ4pg12JZdRkWK+XVH+zKrSHygUSbKr91eUwMjNdVqPiPQ9HuFt9W1nT7GZk3rHc3SRsVyRnDEHGQefY0dLh1sZvw4/5JZ4U/7Atn/wCiErpK5v4cf8ks8Kf9gWz/APRCV0lABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABUV2xSynYdRGxH5VLVfUG2abdMeiwuf0NVHWSGtyn4a/5Fqx/wCuQ/nUHhZjFpbadLxPYyNHIPUEkqfoQf0q14f/AORdsP8Argv8qrECDxuCOBc2R3AfxMrdfyrrl70qkfO/3X/Rs1erkvmbVFFFcRiFcx8QSx8IywxjLTTRxgZxk7s/0rp657xYPMfRIMZ8zVISwxnIUEmtaOlRM4cwXNhJx7q336GXpc/9p3HhSBTv+z2jXE3thdik/wDAga1PDx8/X/EN2OQ10kGf+uaAY/Wuc8H3VloehahqV5cx+arPDAkjgFlQZCqOpyW7V1fhSyay8N2omz58wM8xPUu/zHP5gfhWVOm4e1nL7Unb0Wi/BJ/M48LP23sF1jHmfk7cq/N/cbNFFFB7QUUVBfXH2TT7i54/cxM/PsM0PQcU5OyM3xDPNILXS7V/Llv3KM46pGBlyPfHFalrbRWdrHb267IolCqPasHRTfaprC6nqVp9mWO0VYQDlWLnJYfhgY7V0dZw95uR1YhezSo9tX6/8NYKr31/b6datcXb7EXp6sfQDuaZqmox6ZYtcSAu2dscY6yMeiiqNhpE01xHqGty+fdAZjhAxHB9B3Pv/wDrrqhTVuee34v+u5zxirc0tiM65qFxGI7PRblJ3GVafCxqD3J/p1q1puix2churpzd3z/fuJByPZR2FadFOVbTlgrJg56WirDTGjIUZFKnqpHBpVUIuFAUegFLRWFyAooooAKKKKACiiigAooooAKKKKACiiigApiQxRyM6Rorv95goBb60+igAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACq62Fml210lpAtw3WYRgOf+BdasUUAFcxF4Rmjgt9NbUUbRba4E8dr9m/ena29UaTdgoGx/ADgAZ6k9PRR1uBl3WkS6h/advqV6Z9OvoRClqIghgG0hyHHLZyDz0xVWz0C9Go2Vzq+pR3o09GW1SO18o7iu0u5LNubbkcbRyeOmN6igHqc/caBqmpLHa6xq0FxYLMsrRw2Rikl2tuVWfzCMZAzhRnHaugooo6WA5v4cf8ks8Kf9gWz/8ARCV0lc38OP8AklnhT/sC2f8A6ISukoAKKKKACiiigAooooAKKKKACiiigAooooAKq6qQNHvSeAIHz/3yatVS1r/kAah/17Sf+gmrp/GvUqPxITQ1C+H9PA/59oz/AOOiqeqfuvE2jS9mMsTfiuR+oq9pC7NEsV67beMf+OiqXiL5Dpk4/wCWd9Hk+gOQa6Ia12u9/wAUzSP8R/M2aKKK5DEK5DxxeiyvNJkJAaMzyrk4+ZY+P1IrrJ54raFpbiRIo1xl3bAHOOprzfQ7RfEXilHuZZbu2t5JriSJ2zHGWc7V/HGce31rOU5qXJBatS17WX+bRwY6ScI01u5R/wDSk/yTOk8NeENN06wguJ7cT3UsSs5nAcIxGSFB6c/jXT0UVauoqLd7aHVSo06KtTil6BRRRTNQrH8VyFPDV0qffl2xKPXcwH8s1sVh6832rUtL02PLObhbiQDtGmeT9TWdT4WdOFX76LfTX7tTZijWGFI0+6ihR9BUN9qFtptt595JsTOBxksfQDuaTUdRg0yza4uWOM4VRyzt2AHrVDTbC4uroaprCgT4/cW/UW6//FHua6qdNcvPPb8/QySv70hllb3OralHqeoQmCCEH7Lbv94E/wAbD19B2rcooqalRzfZLYmUuZhRRRWZIUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFAHN/Dj/klnhT/sC2f/AKISukrm/hx/ySzwp/2BbP8A9EJXSUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFZ+vnHh2/x/zwf+VaFZniNivhu+I/54kfnWtH+LH1RcPiRa00FdJtAeCIEB/75FN1PT49T0+S1kJXdyrjqjDkH86ksv8Ajwt/+uS/yqek5OM+Zb3FdqV0Zuh30t7Yst1gXVu5hnA7svf8etaVYsY+yeM5ETiO9thIw9XQ4z+RraqqySldbPX+vmOaV7rqcz44lD6baWJ5F3cr5i+safO38hUngbTYdP8AC1u8S4kuh50jHuT0/DFYnjfU4rbxHFFOcbNNnMRz0kkDL/JcfjXYaND9n0Oxh6eXbxr+SiuSVVusqUXoo3a829PwR5VCEKmJqVmtU+VPskl+rZdooorU9IKKKKACsPQ1Fzq+r35+ZjcfZ0Y9lQDOPbNbM0qwwvK/3UUsfoBXK6bezW3h20tbMb9T1AvKPRAzE7z7AUowdSoor1Oukn7KbXWy/X9DQtF/tjxBNey/NbWDmG3XsX/if+grdqrplgmmadDaRHcIxyx/iJ5J/OrVb1pqUrR2Wi/rz3OebTemwUUUViQFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAc38OP+SWeFP8AsC2f/ohK6Sub+HH/ACSzwp/2BbP/ANEJXSUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFZXihtvhm+P/AEzx+ZArVrH8WEDwte54+VR/48K3w/8AGh6r8y6fxr1NWD/j3j/3B/Kn0iqFUKOgGBS1i9yDF1b9z4g0a46AySQt77l4/UVtVi+J/wB3YW11zi1u45iR2AOD/OtdZY3hEqOrRldwcHII9c1tU1pQl6r8b/qaS+FM4g6Xa+J/HGtw3u8xQRRRqUbBUjBOPxBH4mu5VQqhVGABgCuK+Hshu77Xb9s/6TcKwz6fM3/s1dtXJBRaVRLWSX5X/U83A605S7yl/wClNBRRRWh3BRRTXdYo2kkYKigszHsBQBgeL7uZbCLT7RHknvn2bU6lBy39B9CauaFo/wDZsJluCr3coAkYdEUcBF9gKxdInl1jxob6UFYY7ctAp7KTtH4n5j+NdfW0mqd4R36/5ei/P5HXWcqcI0vm/n/SCiiisTkCiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigDm/hx/ySzwp/2BbP/wBEJXSVzfw4/wCSWeFP+wLZ/wDohK6SgAooooAKKKKACiiigAooooAKKKKACiiigArG8Wrv8L3S9NxjH/kRa2axfFgLaA6r1aWMY9fnFb4b+ND1X5mlP416m1RRVPWLv7Bot5dZwYYWZfrjj9a5pSUIuT6GZRtPEthqmtPpdsrSp5TEzMPkcggFR69axtQvf+Ee03WdM3bQ0ZeyGeSJPl2r9Cf51DMi+G9K8M6jKpCWwaOcAc/vULH9c1NDpOp31ja6/czNdagCs8dq4URqhOdoGODgg59q4cLiKtZTwtTSUlv0jdJp/e2vvYU5XvB9Sz4Ctfsun6ih6x3zQ5yOQiquf0NdVXM+DLu2Hh+WaSeJWe5kkk3OBt3MSM/UYrpQynGCORkc9RXq1Kbpy5LbafccuDpyp4eKkrP/AD1FooorI6grB8S3iP8AZtIWZY5L6RUc5wVjzz+fT35q7qOsxafMkCwzXVzIpZYYF3Nj1PoPeuZt9EOreIpl1gs0ptfNl2HHlOx+VR9FFbxvSj7V79PN/wCSOvD0k5c03ZLU2fDqLPe6jfxKFgkkWCADpsjG3I9if5VvVXsbKLTrGG0twRHEu0Z6n3PuetWK5oppa7mVeaqVG1tsvRaIKKKKoxCiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigDm/hx/wAks8Kf9gWz/wDRCV0lc38OP+SWeFP+wLZ/+iErpKACiiigAooooAKKKKACiiigAooooAKKKKACsXxSxGkRgfxXMQP/AH0K2qwvEtzCIrW3EiGY3cJMYYbgN3XHpXRhk/bRNKfxo3awPGWZdHgsgT/p13FbnHoWyf8A0Gt+uf1v/SPFGg2vVRJLcNx02Lx+przMZrRce9l97S/UxlsaOr2Wn3ummDVdi2u5Sd77ACDxzWfe+JYbXXJtHSLDR2TT+aGxtIBO3GPQZzmoPHMLXul2WnI203t7HETjoOST+GK43U7LUtM0oazq6Mt9NLJauT1KNBsU/oTXq0MPTb53u9P8v1PBzHMK1CbjTjpGzb8uq/FW+Z13hnw7ps/hvTZ7q0SSU24JJzg5yeR0PWprzQotH8rUtHhkMtu+ZIw7MXiPBUA+nUVsaRF9n0Oxh6eXbxr+SgVco+s1FN63XbpY9zCOVKhCF9EkvwKtjqNpqNv51nMsij73YqfQjtVDUNZaSYWGilLi9fqw5SFf7zH+lTXnh3Tb66+0XFvlz98KxUP/ALwHWrlrZW1jF5dnBHCmckIuM/X1pXoxfMrvye3/AAfwOi8FqjNS3g8PWd1qV9cPczsoMszgAt6Ko7DPanaFaXIa61G/QRXF8ysYh/yzVRhQffFQ34/tXxNbWB+a3s1+0zjsz9EU/wA63a5JSlVnzS6f1/wDecvZ07dZb+nRfr9wUUUVRyBRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAc38OP8AklnhT/sC2f8A6ISukrm/hx/ySzwp/wBgWz/9EJXSUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFc0ml22qeLtQmulZjaNB5RV++3JH54rpaxtD+fUtZm9bvZnH91R/jXTRk4RnKL6fqjWDaUmv61NgsFUsxAAGST2rnNKlfXPEp1qKFksIbY29u8nBlbdkuB6cYqfxXcyjTY9OtGxdalILdMfwqfvt9AP51B4h1C48PaVYWOhQq9y52RRkZzGi5b8cAfnXmcrxGIVNbR1fr0+5av5HFXrRowdSey/pD9ZHn+MNAt+oQzzt+CAD9TVnxToj+INDeyikSOTerozjgEHn9M1m6HejxJ4mbWbdGWztrUW6Fxy0jEM2PoOK6qvRk3TlHuv+HOOhCni6dVvWM396so/oIqhVCqMADAFLRTXdY42dztVQSSewrA9RLoh1Y95q91JqD2Gi28dxNEuZpZHxHET0Bx1PtWNba7rlynkrb7pNRJezlwAsKZIO76DB/Hv0rpdL02HSrJbeHLHO6SRvvSMerGsVL2nw6HdKisNrVs30XT1fpt6+hFpGltp8cslxMbi7uWDzykYyewA9BWjRRWqSSsjknOVSXNLcKKKKZAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRQTgEnoK5CDxNqj6bZa9ItoNJvLlIhbiJhNHG77EkMhfaeSpK7RwepxydbB0OvoqlrOpx6Nol5qUyl0tYWlKA4LYGcfj0rKttV1ez1bTrbWzZyR6kj+X9miaMwSKu/YdztvBUNyAvK9OeADoqK5CPxPqbaZBr7C0GjzXSxfZ/LbzliaTy1k8zftPJDbdvQ4z3rr6OgHN/Dj/klnhT/sC2f/ohK6Sub+HH/JLPCn/YFs//AEQldJQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAVi+Gfmtr6Uf8tb+Z/wBcf0rarG8J5PhuCQ5zI8jnPu5reOlGT81+v+RovgfyILYC/wDHl5K/zLptukUY7B5PmY/XHFNgH274g3Ujcrptokaj0eT5ifrgYp/hM+fDqN+eftd9Iyn/AGF+Vf5Gm+FB58mr6gf+Xm/cIfVE+Vf61xYH+DKp/Nd/e9PwPMr+9UpQ7tv5Jf52N6GCK3j8u3iSJM52ooUfkKfRRWx3JJKyCsjxTO8Ph24WI4kn2wp77iAf0zWvWJrv+karo1l133JnI9o1zz+dRU+FnThUvbRb6a/dr+hr28CW1tFBEMJEgRfoBipKKKswbbd2FFFFAgooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigBGAZSp6EYNcRFourHw/YeGJNPcQ2tzEX1HzY/KaGOQOCq7t+8hVXBUAHPOBz3FFHW4dDD1mzvde0/WtGltBbW8tsI7a8MoYSsynOUHK7Tjr1zVSO31TWNa0ee/0yTT49M3ySPJNG3nSNGUwgRidvzMcttPTjrjp6KAepw6aJqv/AAjlv4VbT38iK5TOo+bH5XkrKJAQu7fvwAuCuM98c102o+HND1i4W41bRtPvplTYslzapIwXJOMsCcZJ49zWlRR0A5v4cf8AJLPCn/YFs/8A0QldJXN/Dj/klnhT/sC2f/ohK6SgCjqOi2GqtG1/B5pjBC/Oy4z9CKpf8IfoX/Pj/wCRn/8Aiq26K2jXqxVoyaXqy1UmlZMxP+EP0L/nx/8AIz//ABVH/CH6F/z4/wDkZ/8A4qtuiq+s1/5397H7Wp/MzE/4Q/Qv+fH/AMjP/wDFUf8ACH6F/wA+P/kZ/wD4qtuij6zX/nf3sPa1P5mYn/CH6F/z4/8AkZ//AIqj/hD9C/58f/Iz/wDxVbdFH1mv/O/vYe1qfzMxP+EP0L/nx/8AIz//ABVH/CH6F/z4/wDkZ/8A4qtuij6zX/nf3sPa1P5mYn/CH6F/z4/+Rn/+Ko/4Q/Qv+fH/AMjP/wDFVt0UfWa/87+9h7Wp/MzE/wCEP0L/AJ8f/Iz/APxVQv4E8OSOWfTssep8+T/4quhopPEVnvN/ezCtThiI8tZcy89fzOd/4QHw1/0Df/I8n/xVSJ4J8PxqFTT8Af8ATaT/AOKreorKcpVFabuvMnD0aWFnz4eKg9rxVnbtoYf/AAhmg/8APh/5Gk/+Ko/4QzQf+fD/AMjSf/FVuUVj7OH8qO765if+fkvvZh/8IZoP/Ph/5Gk/+KqzYeHNK0y6FxZWvlSgEBvMZuD9TWnRTVOCd0kKWKryXLKba9WFFFFWc4UUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAcbonhzxjoOgafpFn4k0N7fT7WO1iaXQpi7KihQWIuwM4HOAKvfYfG/wD0MPh//wAEM/8A8mV0lFAHN/YfG/8A0MPh/wD8EM//AMmUfYfG/wD0MPh//wAEM/8A8mV0lFAHN/YfG/8A0MPh/wD8EM//AMmVQ1yTxvovh7UdU/tzw/N9htZbjyv7DnXfsQttz9rOM4xnBrpdQ1vStJdF1TU7OyaQEoLm4SMsB6biM0+G403XNOk+zzWuo2UwaJ9jLLG4IwynGQeDgigDG+w+N/8AoYfD/wD4IZ//AJMo+w+N/wDoYfD/AP4IZ/8A5Mrbn1C2try1tJ5ds92WWFdpO8qNx5AwOPWrNAHN/YfG/wD0MPh//wAEM/8A8mUfYfG//Qw+H/8AwQz/APyZXSUUAc39h8b/APQw+H//AAQz/wDyZR9h8b/9DD4f/wDBDP8A/JldJRQBzf2Hxv8A9DD4f/8ABDP/APJlH2Hxv/0MPh//AMEM/wD8mV0lFAHN/YfG/wD0MPh//wAEM/8A8mVQ1iTxvpNjHcf254fl33Vvb7f7DnXHmzJFuz9rPTfnHfGOOtdbFe2s91NbQ3MMk9vjzokkBaPIyNw6jI6ZqJZ9P1MywLLbXhtpV82MMsnlSKQy5HZgQCM8ggGgDG+w+N/+hh8P/wDghn/+TKPsPjf/AKGHw/8A+CGf/wCTK6SigDm/sPjf/oYfD/8A4IZ//kyj7D43/wChh8P/APghn/8AkyukooA5v7D43/6GHw//AOCGf/5Mo+w+N/8AoYfD/wD4IZ//AJMrpKKAOb+w+N/+hh8P/wDghn/+TKPsPjf/AKGHw/8A+CGf/wCTK6SigDm/sPjf/oYfD/8A4IZ//kyj7D43/wChh8P/APghn/8AkyukqCK+tJ7ua1huoZLi3x50KSAvHnkbh1GfegDk9Pk8b399qlv/AG54fj/s+6W33f2HOfMzDFLux9r4/wBbjHPTPfAv/YfG/wD0MPh//wAEM/8A8mV0McEUMkrxRIjzPvlZVAMjbQuT6naqjJ7ADtT6AOb+w+N/+hh8P/8Aghn/APkyj7D43/6GHw//AOCGf/5MrpKKAOb+w+N/+hh8P/8Aghn/APkyj7D43/6GHw//AOCGf/5MrpKKAOb+w+N/+hh8P/8Aghn/APkyj7D43/6GHw//AOCGf/5MrpKKAOb+w+N/+hh8P/8Aghn/APkyj7D43/6GHw//AOCGf/5MroZpo7eF5p5FiijUs7uwCqBySSegqJtRsk0/7e95brZ7A/2kyqI9p6NuzjHvQByk0njeLxDaaX/bnh8/abWe483+w5/l8p4l24+18587Oc8be+eL/wBh8b/9DD4f/wDBDP8A/JlbyJbXEkV5GsUr+WRFOoBOxtpIVvQ7VPHBwPSpqAOb+w+N/wDoYfD/AP4IZ/8A5Mo+w+N/+hh8P/8Aghn/APkyukooA5v7D43/AOhh8P8A/ghn/wDkyj7D43/6GHw//wCCGf8A+TK6SigDm/sPjf8A6GHw/wD+CGf/AOTKPsPjf/oYfD//AIIZ/wD5MrpKKAOb+w+N/wDoYfD/AP4IZ/8A5Mo+w+N/+hh8P/8Aghn/APkyukqG7vLawtXub64itoI8b5ZnCKuTjkngcmgDB+w+N/8AoYfD/wD4IZ//AJMqho8njfVrGS4/tzw/Fsuri32/2HO2fKmeLdn7WOuzOO2cc9a620u7a/tUubG4iuYJPuSwuHVu3BHBp8MEVvGUt4kiQuzlUUKCzMWY8dyxJJ7kk0Ac99h8b/8AQw+H/wDwQz//ACZR9h8b/wDQw+H/APwQz/8AyZXSUUAc39h8b/8AQw+H/wDwQz//ACZUltZ+MVuomvNd0OW3DgyxxaLMjsueQrG6YKSOhIOPQ9K6CigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooA5LV476T4kaeNMube3l/sufLXFu0ykebHxgOnPTnP4UzUtT1w66mjxG6keG0W4mn0qK3jZ2Z2UDbcOQFAXtuJJHI6Hp20+1fVI9RaLN1HC0KybjwjEEjGcdVHPtUGo6JYarLFNdxyCaEERzQTvBIoPVd6MGwfTOOBR0S9f1/zH1b/rocpdDWLzUPCqam76ffme6VpAsbSbRGwVsAsgYrg9wCenaug0C7umvtV069umvTYTqqXDqquyuivhgoC5BJHAHGKll8MaXNb20Lx3AFqzvDIl5Msis+dx8wMGJOTkknrVvTtMtNJtjBYxFEZzI5Z2dnY9WZmJZj7kk0dWIt0UUUAFFFFABRRRQB57NFfWPinW9b0W1a6uo71baaBBzLG9vFtP8AwF9pz2UvTdLjTw0vimI38tvIk1qrXUUAkkkmeNMlUIILOzHqOrV3lvY29rcXM0EeyS6kEkzbidzBQoPPThQOPSq0+gaZc/bvPtt329ka4O9gWZAArAg/KRgYK4IIz1o6W8rfl/wR9bnMwazrVpNrNtdNfr5OmG7t21BLbzUcbhx5HyleBwRnIPUVv+G0vpNJtr7UdSlu5Lq2jkaMxRpHGSoJ2hV3c57sfbFOg8MaVA1wywSySXMBt5pJrmWV3jOflLMxPc9+K0re3itLWK3t12RQoERck4UDAGT7ULb+vP8A4BL3/ry/4JJRRRQMKKKKACiiigCK6uYrKzmurhtkUEbSOx7KBkn9K830G+Sx1jSNWuIb2K71V5IdRMthNEgMrbohvZAp2kBBgnO7ivRdQsLbVLCWyvo/Nt5htkTcV3DOcZBBpL/T7XU7M2t7F5kJZW2hiuCpDAgjBGCAaFo7/wBeYPVW/ryORkv9ZmsvEmoJrT240m6mFvB5EXlsqIrbZCVLEdsgqeTyeMXILzU9Y8VT2sWpzWFmmn21yI4YYy+9y+RudW4wvIxngYI5zJZ+DLZr3U59X33C3V81ykK3cvklcLgPFkIxyvcHtW9Hp9rFqU1/HFi5mjSKSTceVUkqMZxxuP50R0Sv2X5f5g+tvP8AP/Is0UUUAFFFFABRRRQBkeLP+RL1r/rwn/8ARZri7zSLxPD93obQMNEitX1COTb8hHlkiD8Jcvj0AFei3drDfWU9pdJ5kE8bRyJkjcrDBGRyODRLZwTWD2Uibrd4jEyZPKkYxnr0qWnZ+f8AwSk9V5f8A45b26Njo9jpl5qnnrpcUslrpsNsSFIADu842gcEBQcnk9ql07xFqI03QNV1OZVtLkvaXqlFULKWKxyEjOOU2kA4y49K3ZvDWlzm3Jimia3hECNBdSxExjojFGBcD0bPU+pqpq3heG48Kr4f0yCCCxeRFkSQkhIt+59vX5vTpjOe1aSd3dd/1M4qySfYseGbq81HS21G9kJS8laW2iKgeVAeEHQEkgbuf72O1bFNjRYo1jjUKigKqgYAA7U6l6FBRRRSAKKKKACiiqt/p0GpQrFcPcoqtuBt7qSA/iY2Ukex4oA4/wAK3tzbWPhy1ifZDdT3vmptHzhWcryeRz6U/U9f1aKz8QG1u1jltdWtra2ZolYRo/k5BHf77d889RxXSzeHtMnsLazNsYobQ5t/s8rwtFwR8roQwyCQeec81FF4W0eGzntUtD5VxOlzKDM5LyrtIcsTknKKTzyeTnJoW/8AXdB/X5lewlvrLxW+l3OozahBJZfaVe4jjV42D7SAY1UEEEdQTx1rfqv9ht/7SF/5f+kiLyfM3H7mc4x061Yo6f13/wAg6sKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAoorlPHviDU9Is9N0/w6IBq+s3q2VrLcqWjh+VneRgOuFU4Hc4pAblzrWn2mtWWk3Fxsvr9JHtotjHzBGAXOQMDAI6ke1X68jt9J8TaT8ZPCMfinxNF4gL21+YZF05LVo/kTIIViGHTHANY/j3x/feF9Y1K4tPi3phlt2dodA/sVZQSCcQvKhLA9iTj3xTukl53/ADBJu57bLfWkF5b2k91DHc3O7yIXkAeXaMttU8tgEE46VPXi/i+HxF4o8efD7UND8Sf2G+o6dcSwD7BHcfZnMSvIfmPzbgyrg9NuR1o8c/ES88KajY+EtS8dW+jXsVitxd67LozXD3DsxAVIEBROFJOfUY70P3dHvr+F/wDIFZq62tc9oorzz4S+O28Y2eqWsmrwa42lzIi6pBaPai5R1yC0TAbWBDA446Yr0Om1YSdwooopDCiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAK5jxz4XvfEen2M2i3sdhq+l3a3ljPMm+PeAVKOBztZWIOOa6eigDzrTPC3ji98d6J4i8X6hobrp8FzE1rpkUqKnmKoBVnyXJxznbjAxnNY8Pw68d6b4d1bwrpGqeHI9EuorhYruW0lN7J5u47ZCDsz82C+GOOcGvXaKTSasOLcXdep59rHgrxENF8J3Hhq80yLXfDkPlAXyO1tMGhEbglQGA4BBA7Uaj4W8Zi/tfE2hajo1v4keyW01K2nhlaxugrFlKkHzFKlmwecg16DRVNtu/zJSSSS7WMDwrB4rjtp5PGt5pc1zIV8uHS4XSKEAHPzOSzE++Olb9FFD1BaBRRRSGFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQB//9k=)

### Twitter Sentiment

## [1] "Using direct authentication"

For further understanding of more subjective methods, a Twitter sentiment analysis is conducted for specific parts of the United States heavily impacted by COVID-19 cases. As noted by the top-ranking states of COVID-19 cases, New York state remains the state with the most cases. Twitter sentiment data is sampled from the New York area compared to various other parts of the United States such as Los Angeles area and Texas. The Twitter sentiment analysis is conducted given understanding of basic developer account restrictions.

Get Covid related tweets starting 1-Jan-2020 originating in various parts of the US

TX geocode = ‘40.730610,-73.935242, 25mi’, n=100) LA geocode=‘34.04993,-118.24084,50mi’ NY geocode=‘43.17305,-77.62479,50mi’

* <https://rpubs.com/sumitkumar-00/twitter_sentiment_analysis>
* <https://juliasilge.com/blog/joy-to-the-world/>
* <https://www.rdocumentation.org/packages/rtweet/versions/0.7.0/topics/search_tweets>

#### Twitter sample NY/NY

## [1] "2020-06-01 17:19:31 EDT"

## [1] "2020-06-05 17:23:10 EDT"

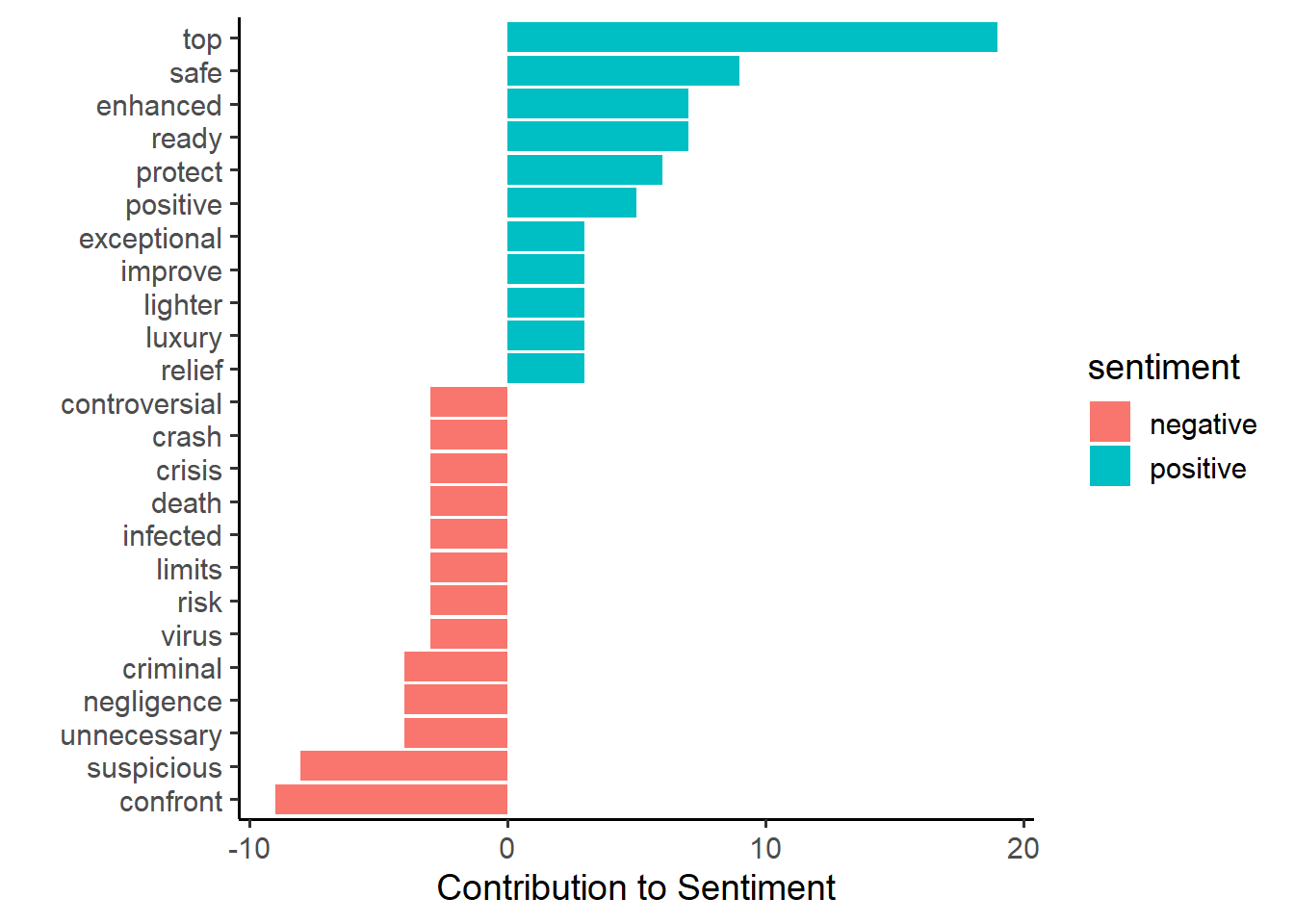
| **id** | **screenname** | **created** | **text** |
| --- | --- | --- | --- |
| 1269016938715533318 | reaListwhiteb0y | 2020-06-05 17:23:10 | @realDonaldTrump and just like that, the #coronavirus isn’t a priority for the media.. |
| 1269012098090180610 | MrUntactical | 2020-06-05 17:03:55 | @WUDeptMedicine @WUICTS @WUSTLmed Shouldn't you folks be in the lab working on a #coronavirus vaccine? |
| 1268993163500367879 | greekisland | 2020-06-05 15:48:41 | RT @gtpgr: Eiriana Luxury Suites Ready to Welcome Visitors to #Milos – Opening June 20 https://t.co/5VrBrGmX34 @EirianaM #EirianaLuxurySuit… |
| 1268988447609573377 | nytowns | 2020-06-05 15:29:57 | RT @generalcode: New Normal in Local Gov: @TownofAuburn to hold town meeting on football field https://t.co/u7MtIiNyNi via @telegramdotcom… |
| 1268982169029902338 | generalcode | 2020-06-05 15:05:00 | New Normal in Local Gov: @TownofAuburn to hold town meeting on football field https://t.co/u7MtIiNyNi via… https://t.co/gQ8ZYIZFHa |
| 1268961640407543810 | shirleyvols | 2020-06-05 13:43:25 | RT @bamanewsnow: #Governor Ivey Announces Development of #Coronavirus Relief Fund Expenditure Request Form #Governor announces the developm… |

#### Word Frequency top 6

#### Word frequency is generated through a table as understood by associated negative or positive sentiment. The top word frequency is understood as positive, but there is still prevalence of negative keywords in tweets.

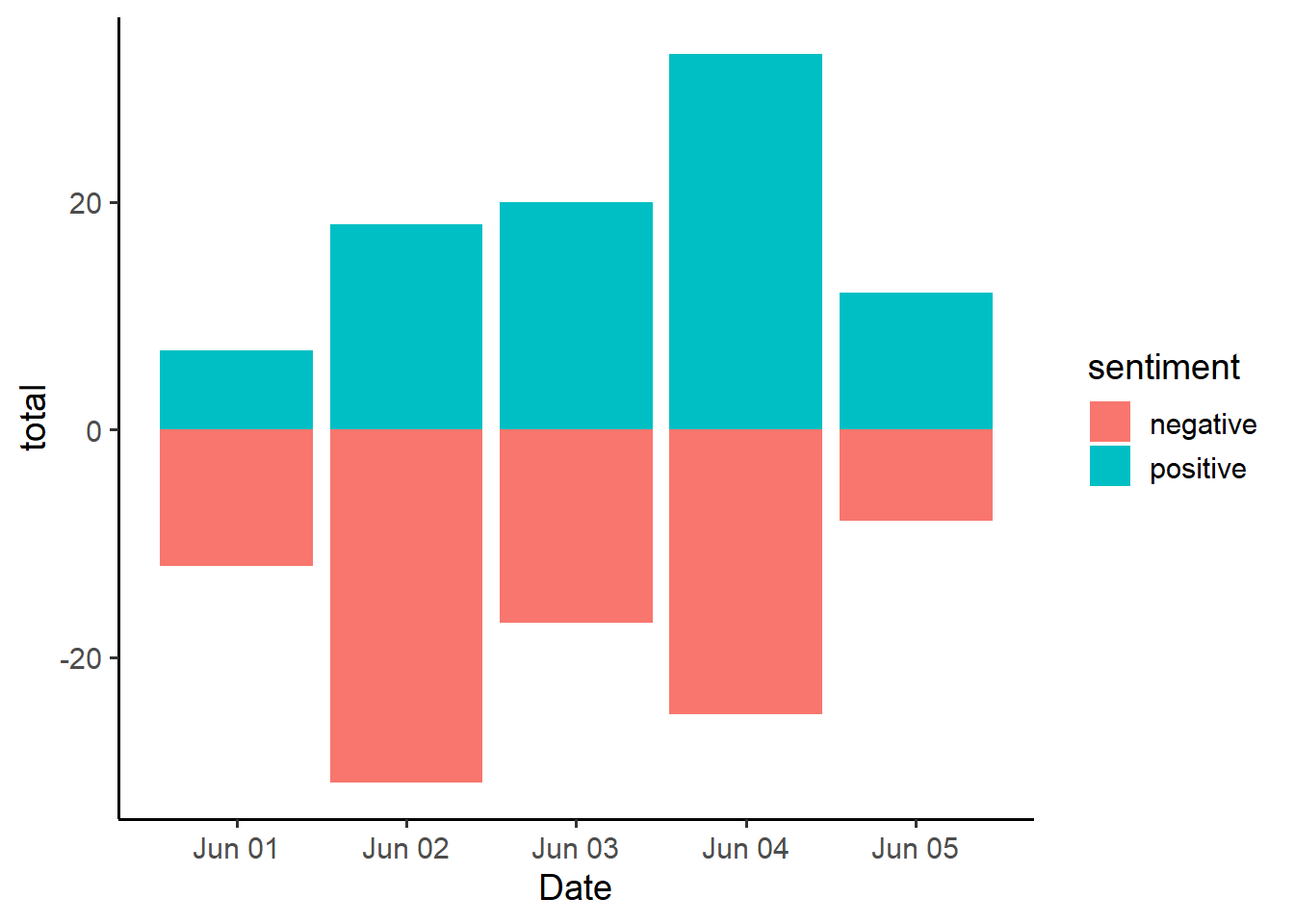
| **word** | **sentiment** | **n** |
| --- | --- | --- |
| top | positive | 19 |
| confront | negative | 9 |
| safe | positive | 9 |
| suspicious | negative | 8 |
| enhanced | positive | 7 |
| ready | positive | 7 |

#### Contribution to Sentiment



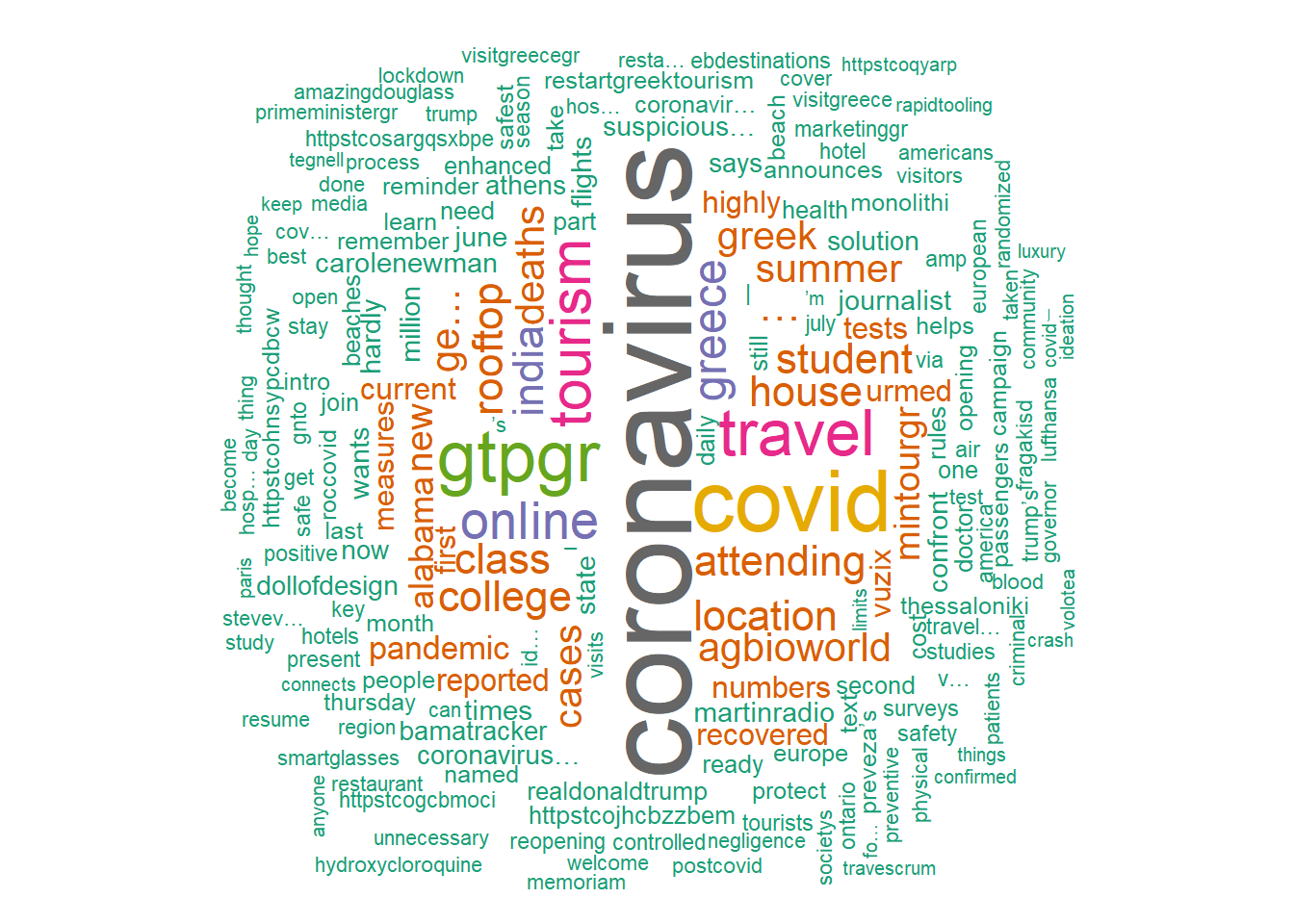
#### Pos/Neg Sentiment

#### The following graph is generated as a means of understanding the everchanging tone and positive/negative association of words as time progresses from COVID-19 prevalence.



#### WordCloud

#### A word cloud is also generated to give a visual understanding of “hot” words and the most common phrases tweeted relative to the COVID-19 outbreak and pandemic.



## Biomarker Modeling

### Modeling Overview

The sudden increase of COVID-19 cases put high pressure on our health-care services worldwide. It is vital that fast, accurate and early clinical assessments are made in order to reduce the disease severity. To support decision making and logistical planning in health care systems, this study leverages the data of a prior study.

In the study, blood samples from 485 infected patients in the region of Wuhan, China were used to identify crucial predictive biomarkers of case fatality rates associated with Covid-19. The new study here will include 375 samples of the aforementioned 485 patients. Machine learning tools will be used to identify key biomarkers that can be used to predict mortality of the 375 individual patients sampled. This study will attempt to improve upon the accuracy of the predictions in the original study by using enhanced machine learning tools associated with the language R.

### Motivation

The study is motivated by the fact that patients can potentially survive Covid-19, even with preconditions, if the biomarkers associated with their preconditions are exposed early enough in the process. Machine learning models may be the best way to detect these preconditions, so patients diagnosed with Covid-19 can be routed to expedited care with specific treatments.

### Problem Statement

The challenge is to find which Covid-19 biomarkers, when leveraged with machine learning, yield case fatality predictions with the high accuracy.

### About Model Data

The data tracks several patients over their hospital stay and keeps records on the patients 74 bio markers. The admission time, discharge time and outcome make a unique patient.

* Data set dimensions are 6120 rows by 81 columns
* Dependent variable is outcome with 0 death and 1 survived

The github location for the original study can be found here: Data Sourcing: <https://github.com/HAIRLAB/Pre_Surv_COVID_19>

### Data Loading

The github location for the original data can be found here: <https://github.com/HAIRLAB/Pre_Surv_COVID_19/tree/master/data>

#### Biomarker definitions:

The most important biomarkers for the analysis are listed with brief definitions. The three primary categories of biomarkers fall into cholesterol, inflammation, and clotting.

percent\_lymphocyte: Lymphocytes are a type of white blood cell, consisting of both B cells and T cells. B cells are responsible for the creation of antibodies, while T cells are responsible for controlling immune response to foreign substances. This marker indicates the percent of lymphocytes present relative to a patient’s white blood cell count.

lactate\_dehydrogenase: LDH is an enzyme released when there is tissue damage present in a patient’s body tissues. This marker measures the amount of LDH in blood or other body fluids and is vital in monitoring viral conditions or cancer.

lymphocyte\_count lymphocytes are a type of white blood cell, consisting of both B cells and T cells. B cells are responsible for the creation of antibodies, while T cells are responsible for controlling immune response to foreign substances. This marker indicates the count of lymphocytes present in a patient’s blood count.

albumin: Albumin is a protein created by your liver. This protein is primarily responsible for maintaining and keeping fluid levels in your bloodstream so that it does not leak into other tissues. This protein also helps carry hormones, vitamins, and enzymes throughout the body. Higher levels can indicate a patient fighting infections. Low levels can indicate problems with liver or inflammatory diseases.

neutrophils\_percent: Neutrophils are another type of white blood cell that makes up 55 to 70 percent of the white blood cell count. They consist of the majority of cells that make up the immune system response. Neutrophils help fight antigens and are not limited to a specific area of circulation. This marker shows the percent of neutrophils present relative to a patient’s white blood cell count.

eosinophils\_percent: Eosinophils are another disease-fighting white blood cell. The cell is primarily responsible for destroying antigens, particularly viruses. They also have a role in the inflammatory response. This marker shows the percent of eosinophils present relative to a patient’s white blood cell count.

eosinophil\_count: Eosinophils are another disease-fighting white blood cell. The cell is primarily responsible for destroying antigens, particularly viruses. They also have a role in the inflammatory response. This marker shows the count of eosinophils present in a patient’s blood count.

mean\_corpuscular\_hemoglobin\_concentration: MCHC is a measure of the concentration of hemoglobin present in a given volume of packed red blood cell.

x2019\_n\_co\_v\_nucleic\_acid\_detection: This marker is a specific test result from a detection test responsible for identifying the presence of nucleic acid from SARS-CoV-2 in humans. This test shows the presence or absence of viral infection by SARS-CoV-2.

basophil\_percent: Basophil is naturally produced by several types of white blood cells. Basophil contains heparin which naturally prevents blood clotting, while also releasing histamine during allergic reactions. It is also responsible for helping the body produce an antibody called immunoglobulin E, which help mediate inflammatory responses in the body. This marker shows the percent of basophil present relative to a patient’s white blood cell count.

total\_cholesterol: Cholesterol is a waxy substance found in blood that assists in building healthy cells. High levels of cholesterol can coat arteries and restrict blood flow. Total cholesterol is measured as a count in a patient’s blood panel.

procalcitonin: Procalcitonin is a protein produced by many cells in the body in response to both bacterial infections and tissue injury. The count of procalcitonin can increase based on systemic infections and sepsis.

thrombin\_time: Thrombin time or thrombin clotting time is a blood test that measures the time It takes for a clot to form in the plasma of a blood sample. This assists in helping doctors investigate excessive bleeding or abnormal blood clot formation.

red\_blood\_cell\_count: Red blood cell count is a test conducted to find out how many red blood cells are present in your blood. It is also known as an erythrocyte count. RBC’s contain hemoglobin, which is responsible for carrying oxygen to body tissues. This count is vital in understanding how tissues are functioning and healing in response to viral infections.

indirect\_bilirubin: Bilirubin tests measure the amount of bilirubin in blood, an orange-yellow pigment that forms when red blood cells break down. Indirect bilirubin is specifically bound to albumin. High levels of bilirubin indicate that the body is destroying too many red blood cells, which can also indicate liver or tissue damage.

h\_bs\_ag: HBsAg is the antigen of the hepatitis B virus. Presence of this indicates current HBV infection.

hco3: Bicarbonate is a byproduct of the body’s metabolic function. Blood contains bicarbonate which is brought to the lungs and exhaled as carbon dioxide. Kidneys excrete and reabsorb bicarbonate and maintain appropriate acid balance in the body.

platelet\_count: Platelets are small fragments of cells present for creating normal and protective blood clots. The platelet count test determined the number of platelets present in a sample of blood.

serum\_sodium: Sodium tests are conducted to see how much sodium is present in the blood. Sodium is an essential mineral in the body that is important for healthy nerve and muscle function.

hemoglobin: Hemoglobin is a red blood cell protein responsible for transporting oxygen in the body’s blood. A low count of hemoglobin can indicate deficiencies and anemia, which can hinder healing and recovery in the body from virus or infection.

lactate\_dehydrogenase: LDH is an enzyme released when there is tissue damage present in a patient’s body tissues. This marker measures the amount of LDH in blood or other body fluids and is vital in monitoring viral conditions or cancer.

monocytes\_percent: Monocytes are another type of white blood cell. They assist in fighting bacteria, viruses, and other infections. They are a key component of white blood cell immune response. High levels indicate the body fighting infection.

international\_standard\_ratio International Normalized Ratio is a preferred test for patients to assess the risk of bleeding or coagulation status in patients. This ratio shows how well blood clots, and if there are any abnormalities present.

percent\_lymphocyte: Lymphocytes are a type of white blood cell, consisting of both B cells and T cells. B cells are responsible for the creation of antibodies, while T cells are responsible for controlling immune response to foreign substances. This marker indicates the percent of lymphocytes present relative to a patient’s white blood cell count.

eosinophil\_count: Eosinophils are another disease-fighting white blood cell. The cell is primarily responsible for destroying antigens, particularly viruses. They also have a role in inflammatory response. This marker shows the count of eosinophils present in a patient’s blood count.

eosinophils\_percent: Eosinophils are another disease-fighting white blood cell. The cell is primarily responsible for destroying antigens, particularly viruses. They also have a role in inflammatory response. This marker shows the percent of eosinophils present relative to a patient’s white blood cell count.

neutrophils\_percent: Neutrophils are another type of white blood cell, that makes up 55 to 70 percent of the white blood cell count. They consist of the majority of cells that make up the immune system response. Neutrophils help fight antigens and are not limited to a specific area of circulation. This marker shows the percent of neutrophils present relative to a patient’s white blood cell count.

amino\_terminal\_brain\_natriuretic\_peptide\_precursor\_nt\_pro\_bnp: Natriuretic peptide tests measure the presence of brain natriuretic peptide and B-terminal pro b-type natriuretic peptides present. These are substances created by the heart. Small levels are normally found in the bloodstream. High levels of BNP, NT-proBNP indicate that the heart is failing to pump the appropriate amount of blood needed.

thrombocytocrit: This device measures the platelet content of the blood. Platelets are small fragments of cells present for creating normal and protective blood clots. The platelet count test determined the number of platelets present in a sample of blood.

hypersensitive\_cardiac\_troponin\_i: This is the preferred biomarker for acute myocardial infarction. This helps diagnose acute coronary syndromes in those who are critically ill.

fibrin\_degradation\_products: Fibrin degradation products are components of the blood produced by clot degeneration. This substance remains in your bloodstream after your body dissolves a blood clot, which in return regulates and manages clot dissolving. This specific test determines the amount of FDPs present in blood to ensure that the body is healing correctly, particularly in response to clots and blood illnesses.

procalcitonin: Procalcitonin is a protein produced by many cells in the body in response to both bacterial infections and tissue injury. The count of procalcitonin can increase based on systemic infections and sepsis.

albumin: Albumin is a protein created by your liver. This protein is primarily responsible for maintaining and keeping fluid levels in your bloodstream so that it does not leak into other tissues. This protein also helps carry hormones, vitamins, and enzymes throughout the body. Higher levels can indicate a patient fighting infections. Low levels can indicate problems with liver or inflammatory diseases

prothrombin\_activity: Prothrombin time is a test used to help detect and diagnose a bleeding or excessivee clotting disorder. This helps doctors evaluate a patient’s ability to appropriately form blood clots.

d\_d\_dimer: A D-dimer test is used to help rule out the presence of serious life-threatening blood clots. High levels of D-dimer, the protein leftover from clotting, can indicate deep vein thrombosis and other clotting disorders.

aspartate\_aminotransferase: Aspartate aminotransferase is an enzyme created by the liver but found in many cells throughout the body. High levels of AST are found in the body when liver damage is present, as the liver releases AST in response to distress of the liver.

neutrophils\_count: Neutrophils are another type of white blood cell, that makes up 55 to 70 percent of the white blood cell count. They consist of the majority of cells that make up the immune system response. Neutrophils help fight antigens and are not limited to a specific area of circulation. This marker shows the count of neutrophils present in a patient’s blood cell count.

direct\_bilirubin: Bilirubin tests measure the amount of bilirubin in blood, an orange-yellow pigment that forms when red blood cells break down. Direct bilirubin specifically travels from the liver to the small intestines. High levels of bilirubin indicate that the body is destroying too many red blood cells, which can also indicate liver or tissue damage.

thrombin\_time: Thrombin time or thrombin clotting time is a blood test that measures the time it takes for a clot to form in the plasma of a blood sample. This assists in helping doctors investigate excessive bleeding or abnormal blood clot formation.

high\_sensitivity\_c\_reactive\_protein: High sensitivity C-reactive protein is a biomarker used in determining coronary heart disease. The protein itself is produced when there is damage or inflammation in blood vessels, particularly in the heart.

## [1] 6120 81

## [1] 6120 81

### View Loaded data

## [1] 6120 81

## # A tibble: 6 x 81

## patient\_id re\_date age gender admission\_time

## <dbl> <dttm> <dbl> <dbl> <dttm>

## 1 1 2020-01-31 01:09:00 73 1 2020-01-30 22:12:47

## 2 NA 2020-01-31 01:25:00 73 1 2020-01-30 22:12:47

## 3 NA 2020-01-31 01:44:00 73 1 2020-01-30 22:12:47

## 4 NA 2020-01-31 01:45:00 73 1 2020-01-30 22:12:47

## 5 NA 2020-01-31 01:56:00 73 1 2020-01-30 22:12:47

## 6 NA 2020-01-31 01:59:00 73 1 2020-01-30 22:12:47

## # ... with 76 more variables: discharge\_time <dttm>, outcome <dbl>,

## # hypersensitive\_cardiac\_troponin\_i <dbl>, hemoglobin <dbl>,

## # serum\_chloride <dbl>, prothrombin\_time <dbl>, procalcitonin <dbl>,

## # eosinophils\_percent <dbl>, interleukin\_2\_receptor <dbl>,

## # alkaline\_phosphatase <dbl>, albumin <dbl>, basophil\_percent <dbl>,

## # interleukin\_10 <dbl>, total\_bilirubin <dbl>, platelet\_count <dbl>,

## # monocytes\_percent <dbl>, antithrombin <dbl>, interleukin\_8 <dbl>,

## # indirect\_bilirubin <dbl>, red\_blood\_cell\_distribution\_width <dbl>,

## # neutrophils\_percent <dbl>, total\_protein <dbl>,

## # quantification\_of\_treponema\_pallidum\_antibodies <dbl>,

## # prothrombin\_activity <dbl>, h\_bs\_ag <dbl>, mean\_corpuscular\_volume <dbl>,

## # hematocrit <dbl>, white\_blood\_cell\_count <dbl>,

## # tumor\_necrosis\_factor <dbl>,

## # mean\_corpuscular\_hemoglobin\_concentration <dbl>, fibrinogen <dbl>,

## # interleukin\_1 <dbl>, urea <dbl>, lymphocyte\_count <dbl>, ph\_value <dbl>,

## # red\_blood\_cell\_count <dbl>, eosinophil\_count <dbl>,

## # corrected\_calcium <dbl>, serum\_potassium <dbl>, glucose <dbl>,

## # neutrophils\_count <dbl>, direct\_bilirubin <dbl>,

## # mean\_platelet\_volume <dbl>, ferritin <dbl>,

## # rbc\_distribution\_width\_sd <dbl>, thrombin\_time <dbl>,

## # percent\_lymphocyte <dbl>, hcv\_antibody\_quantification <dbl>,

## # d\_d\_dimer <dbl>, total\_cholesterol <dbl>, aspartate\_aminotransferase <dbl>,

## # uric\_acid <dbl>, hco3 <dbl>, calcium <dbl>,

## # amino\_terminal\_brain\_natriuretic\_peptide\_precursor\_nt\_pro\_bnp <dbl>,

## # lactate\_dehydrogenase <dbl>, platelet\_large\_cell\_ratio <dbl>,

## # interleukin\_6 <dbl>, fibrin\_degradation\_products <dbl>,

## # monocytes\_count <dbl>, plt\_distribution\_width <dbl>, globulin <dbl>,

## # glutamyl\_transpeptidase <dbl>, international\_standard\_ratio <dbl>,

## # basophil\_count\_number <dbl>, x2019\_n\_co\_v\_nucleic\_acid\_detection <dbl>,

## # mean\_corpuscular\_hemoglobin <dbl>,

## # activation\_of\_partial\_thromboplastin\_time <dbl>,

## # high\_sensitivity\_c\_reactive\_protein <dbl>,

## # hiv\_antibody\_quantification <dbl>, serum\_sodium <dbl>,

## # thrombocytocrit <dbl>, esr <dbl>, glutamic\_pyruvic\_transaminase <dbl>,

## # e\_gfr <dbl>, creatinine <dbl>

### Create Patient Master

## [1] 6120 3

### View Patient Master

## # A tibble: 6 x 3

## patient\_id admission\_time discharge\_time

## <dbl> <dttm> <dttm>

## 1 1 2020-01-30 22:12:47 2020-02-17 12:40:09

## 2 2 2020-02-04 21:39:03 2020-02-19 12:59:01

## 3 3 2020-01-23 10:59:36 2020-02-08 17:52:31

## 4 4 2020-01-31 23:03:59 2020-02-18 12:59:12

## 5 5 2020-02-01 20:59:54 2020-02-18 10:33:06

## 6 6 2020-01-24 10:47:10 2020-02-07 09:06:58

A date frame is created with corrected patient\_id label. The data is still a sparse data frame at this point with the sparse data being NA (as opposed to 0). Dimension still 6120,81 here.

## [1] 6120 81

## # A tibble: 1 x 5

## patient\_id admission\_time discharge\_time re\_date age

## <dbl> <dttm> <dttm> <dttm> <dbl>

## 1 1 2020-01-30 22:12:47 2020-02-17 12:40:09 2020-01-31 01:09:00 73

### Aggregate Data

Taking the data from being sparse to denser or at the patient level. Takes the “max” of all metrics and groups by patient. This approach assumes the worst case of the patients’ biomarkers. Additionally, the NA and infinite numbers are set to zero.

Note: This reduces the dataset to the number of patients, which is 375. Dimension is bow 371,80 here. We lost one column on purpose as it was aggregate out of the mix. It was the time the bio marker was taken. We aren’t interested in the time component with this analysis, as the objective is to simply determine if we can predict a a mortality outcome with the biomarkers and patient attributes.

New/Changed Features: \* duration\_days - is the differences in days the patient was admitted to discharged \* factorized - patient\_id,gender,age,outcome

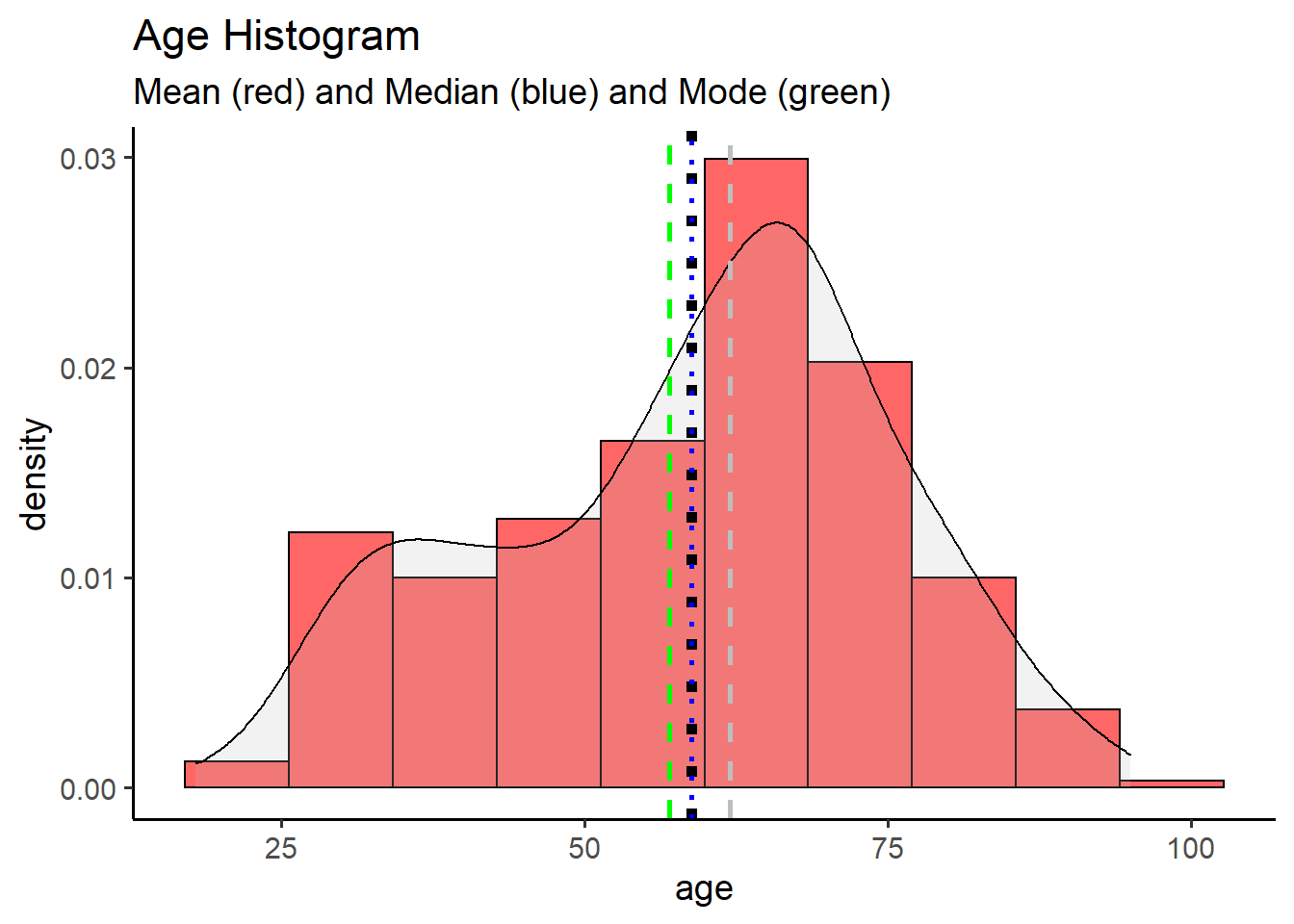
### Age and Gender Analysis

The clean data set will set to new analysis data set so it can be changed to facilitate the analysis of age and gender.

### Age Histogram

The histogram showing distribution of age across the 375 patients is very normal looking.

Hist Line Color Legend: \* Mean (red) \* Median (gray) \* Mode (green)

 ### Age Discretized

## Age Percentage

## 1 young 0.05866667

## 2 middle 0.30400000

## 3 older 0.23466667

## 4 old 0.40266667

### Age Outliers

Checking how many patients greater than or equal to 90 years old.

## # A tibble: 5 x 1

## # Groups: age [5]

## age

## <dbl>

## 1 95

## 2 90

## 3 94

## 4 92

## 5 91

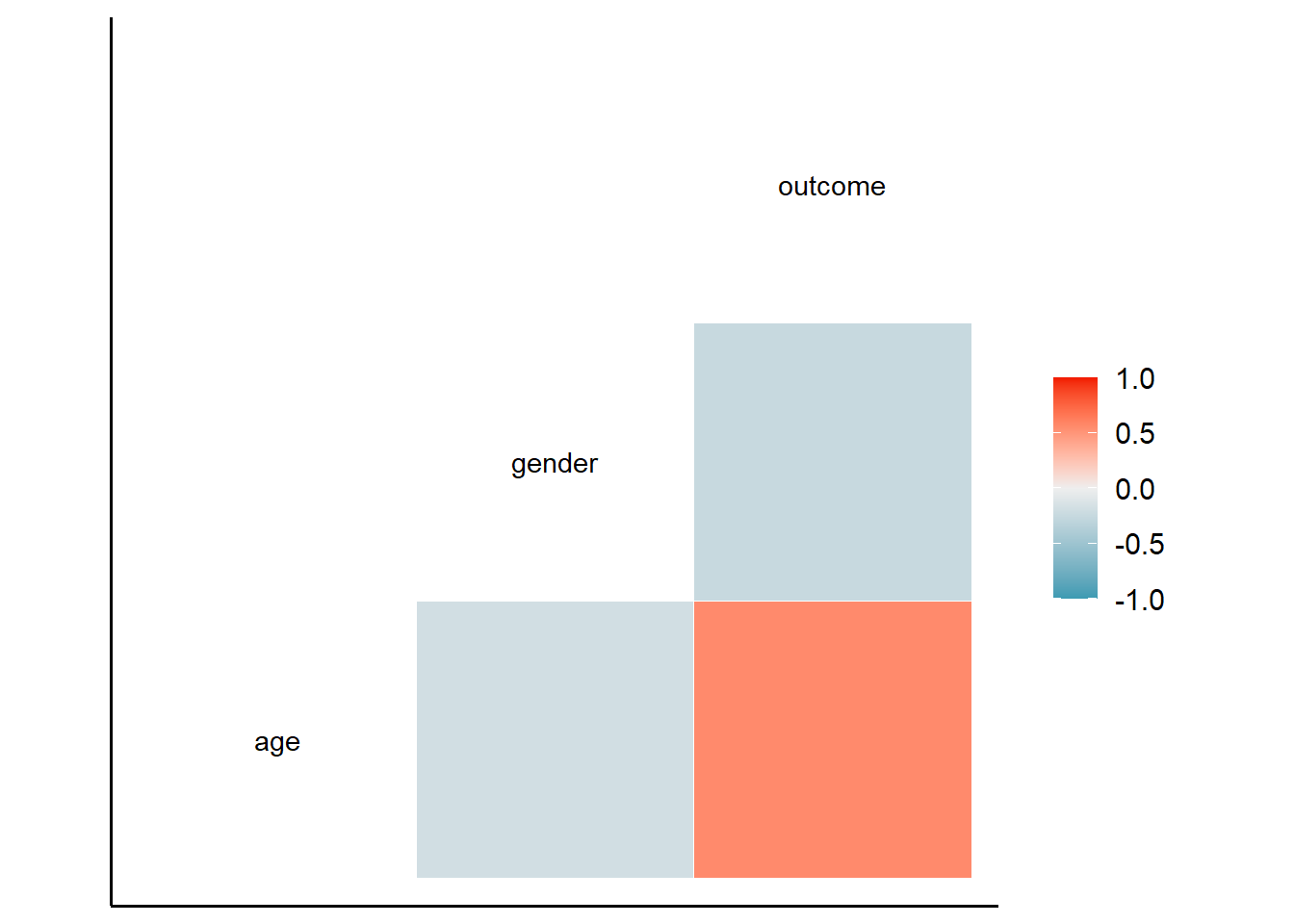
### Gender Analysis

Gender isn’t significantly correlated with negative correlation whereas age does show significant positive correlation to the binary fatality outcome.

## Gender Percentage

## 1 Female 0.5973333

## 2 Male 0.4026667



### Correlation Analysis

Visual analysis of the outcome (dependent feature) with all others (independent features).

During the analysis, the following features were found to be important. They are list in order of relative importance.

* lactate\_dehydrogenase 100.00
* monocytes\_percent 83.50
* international\_standard\_ratio 78.47
* percent\_lymphocyte 64.12
* eosinophil\_count 59.75
* eosinophils\_percent 50.99
* neutrophils\_percent 49.15
* amino\_terminal\_brain\_natriuretic\_peptide\_precursor\_nt\_pro\_bnp 48.62
* thrombocytocrit 46.40
* hypersensitive\_cardiac\_troponin\_i 45.83
* fibrin\_degradation\_products 43.42
* procalcitonin 41.40
* albumin 40.76
* prothrombin\_activity 37.53
* d\_d\_dimer 33.37
* aspartate\_aminotransferase 32.88
* neutrophils\_count 31.01
* direct\_bilirubin 30.13
* thrombin\_time 29.82
* high\_sensitivity\_c\_reactive\_protein 28.38

Figure 1 - “outcome”,“percent\_lymphocyte”,“lactate\_dehydrogenase”,“lymphocyte\_count”,“albumin”, “neutrophils\_percent”,“eosinophils\_percent”,“eosinophil\_count”, “x2019\_n\_co\_v\_nucleic\_acid\_detection”,“direct\_bilirubin”,“thrombin\_time”, “monocytes\_percent”,“international\_standard\_ratio”

Figure 2 - “outcome”,“amino\_terminal\_brain\_natriuretic\_peptide\_precursor\_nt\_pro\_bnp”, “thrombocytocrit”,“hypersensitive\_cardiac\_troponin\_i”,“fibrin\_degradation\_products”,“procalcitonin”, “prothrombin\_activity”,“d\_d\_dimer”,“aspartate\_aminotransferase”,“neutrophils\_count”, “high\_sensitivity\_c\_reactive\_protein”

Figure 1

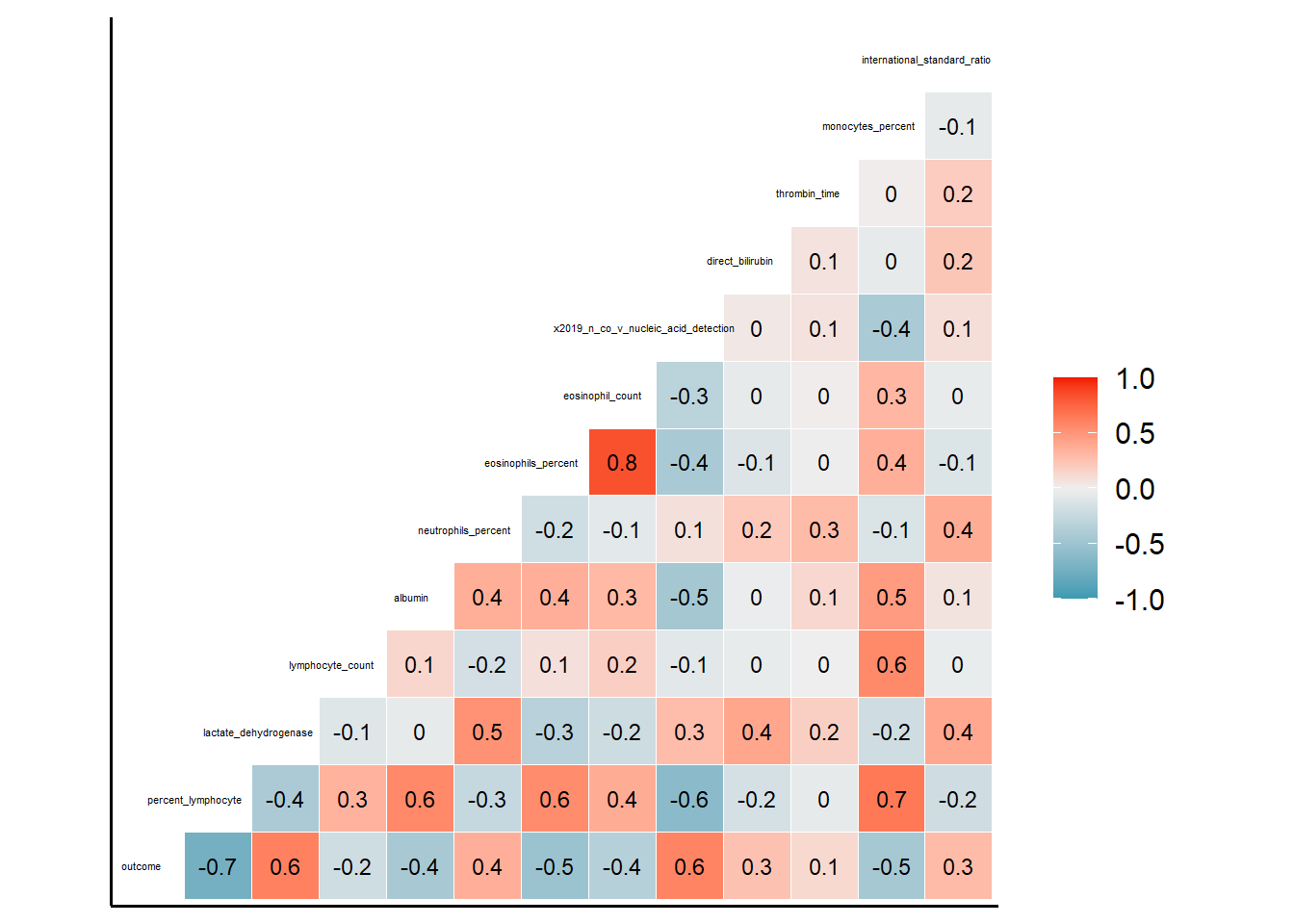
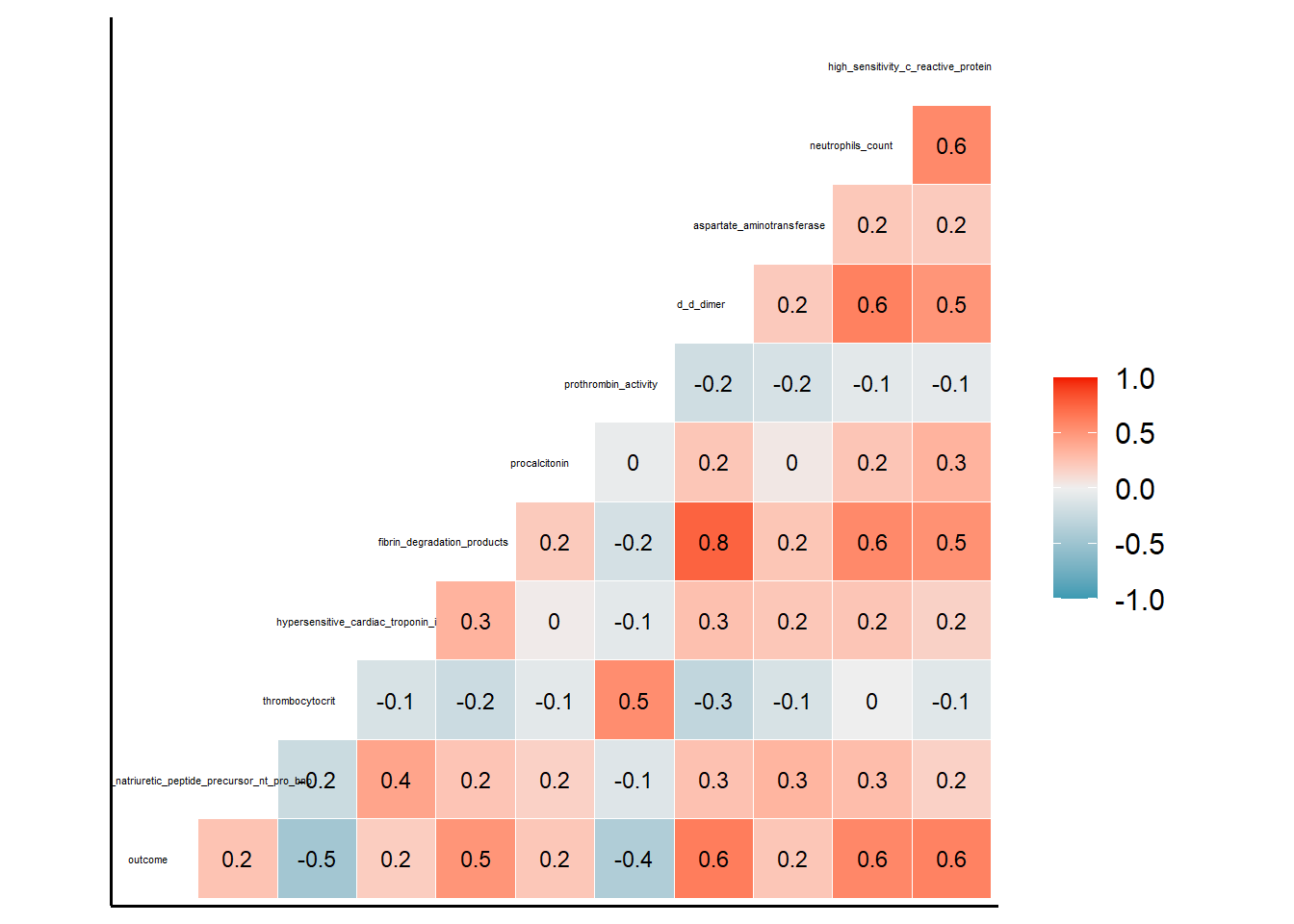


Figure 2



## [1] ""

## Kmeans Clustering

kmeans() function returns a list of components, including:

\*cluster: A vector of integers (from 1:k) indicating the cluster to which each point is allocated

\*centers: A matrix of cluster centers (cluster means)

\*totss: The total sum of squares (TSS), i.e ∑(xi−x¯)2. TSS measures the total variance in the data.

\*withinss: Vector of within-cluster sum of squares, one component per cluster

\*tot.withinss: Total within-cluster sum of squares, i.e. sum(withinss)

\*betweenss: The between-cluster sum of squares, i.e. totss−tot.withinss

\*size: The number of observations in each cluster

## # A tibble: 6 x 5

## hypersensitive\_cardi~ hemoglobin serum\_chloride prothrombin\_time procalcitonin

## <dbl> <dbl> <dbl> <dbl> <dbl>

## 1 19.9 140 103. 14.1 0.09

## 2 16.9 151 100. 14.3 0.09

## 3 0 126 103. 13.6 0.06

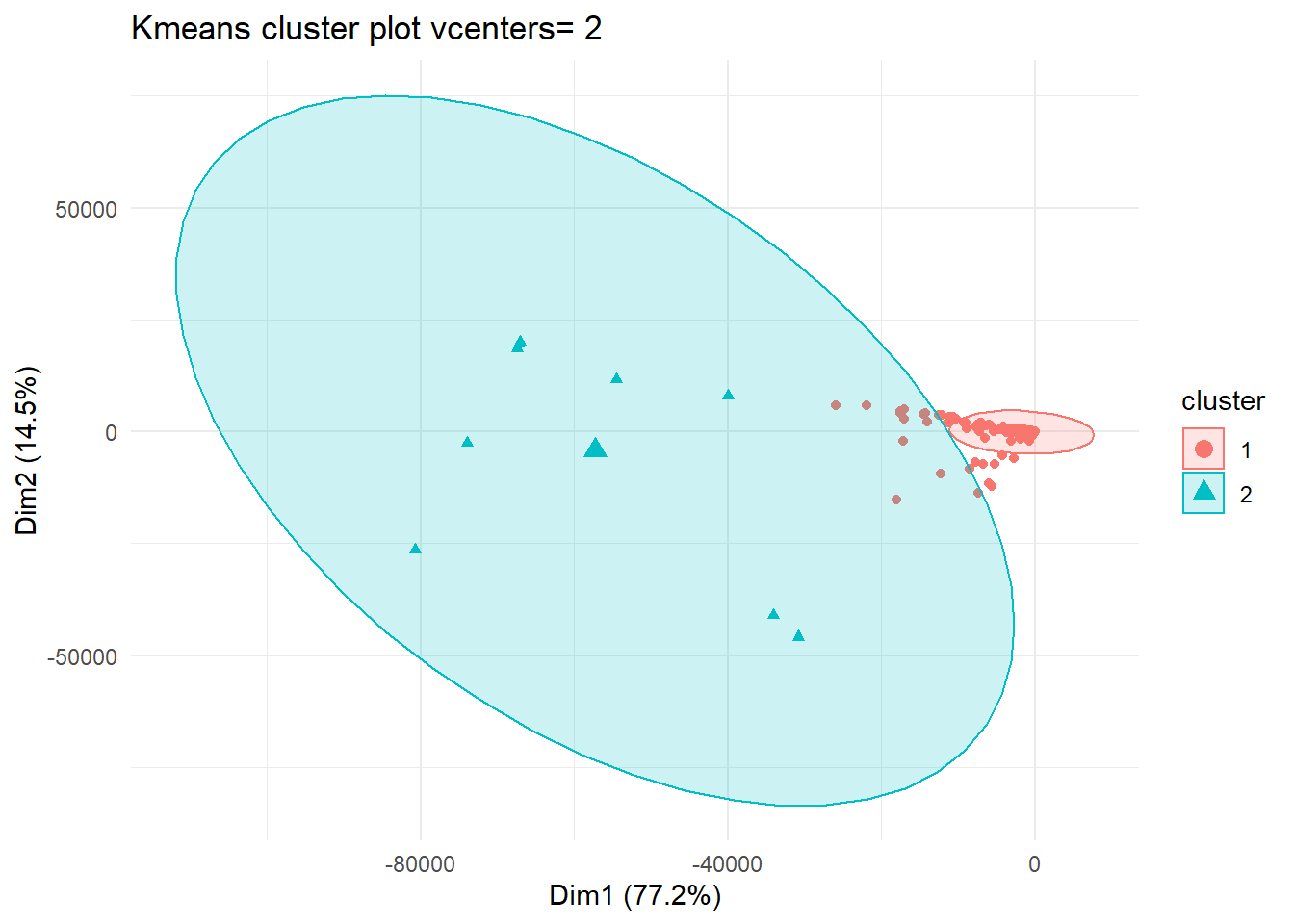
## 4 4.8 110 103. 16.3 0.38

## 5 5.6 134 102. 14.6 0.02

## 6 19.7 108 106. 12.4 0.1

### Cluster Build

## [[1]]



##

## [[2]]

## Length Class Mode

## cluster 375 -none- numeric

## centers 148 -none- numeric

## totss 1 -none- numeric

## withinss 2 -none- numeric

## tot.withinss 1 -none- numeric

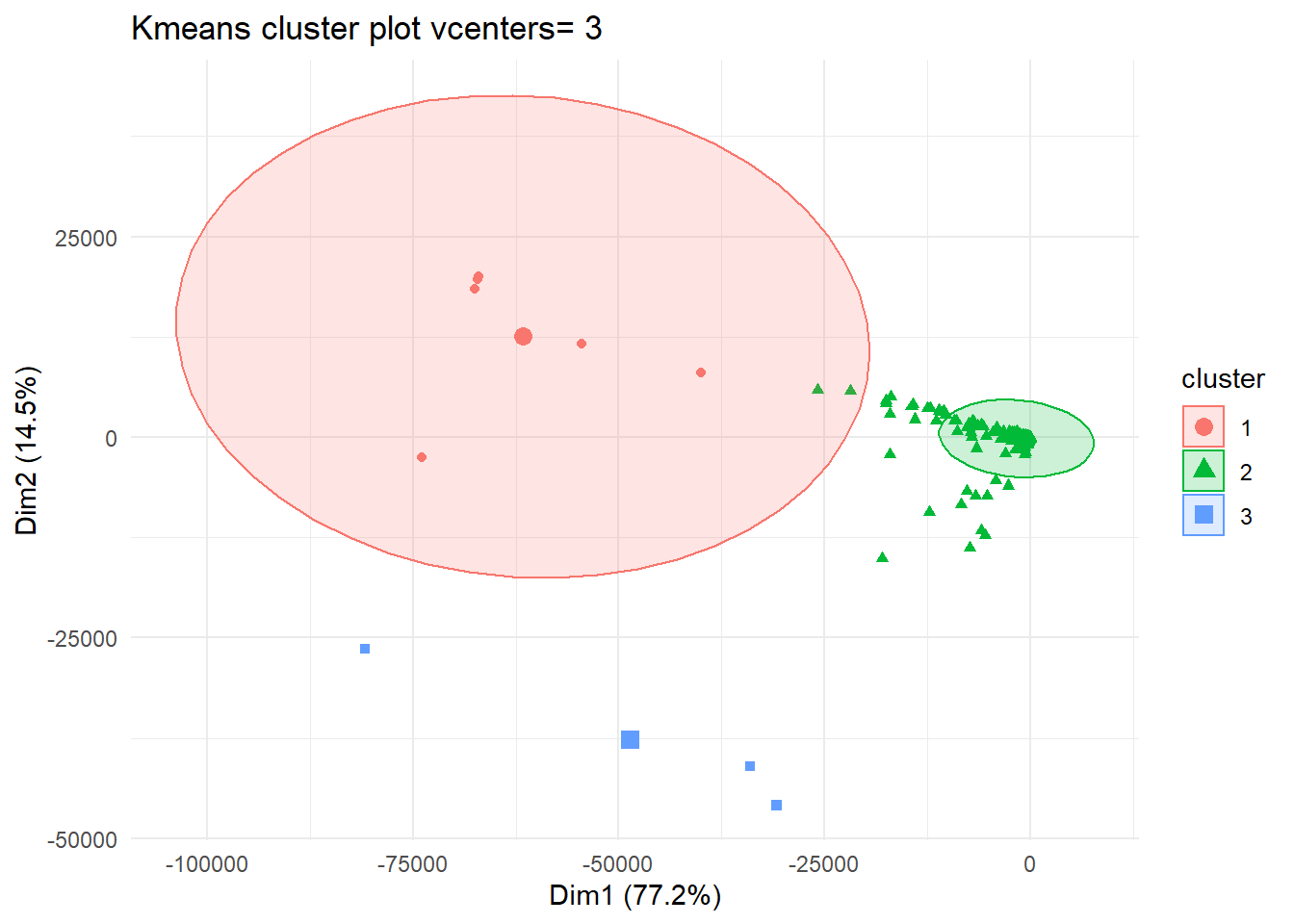
## betweenss 1 -none- numeric

## size 2 -none- numeric

## iter 1 -none- numeric

## ifault 1 -none- numeric

## [[1]]



##

## [[2]]

## Length Class Mode

## cluster 375 -none- numeric

## centers 222 -none- numeric

## totss 1 -none- numeric

## withinss 3 -none- numeric

## tot.withinss 1 -none- numeric

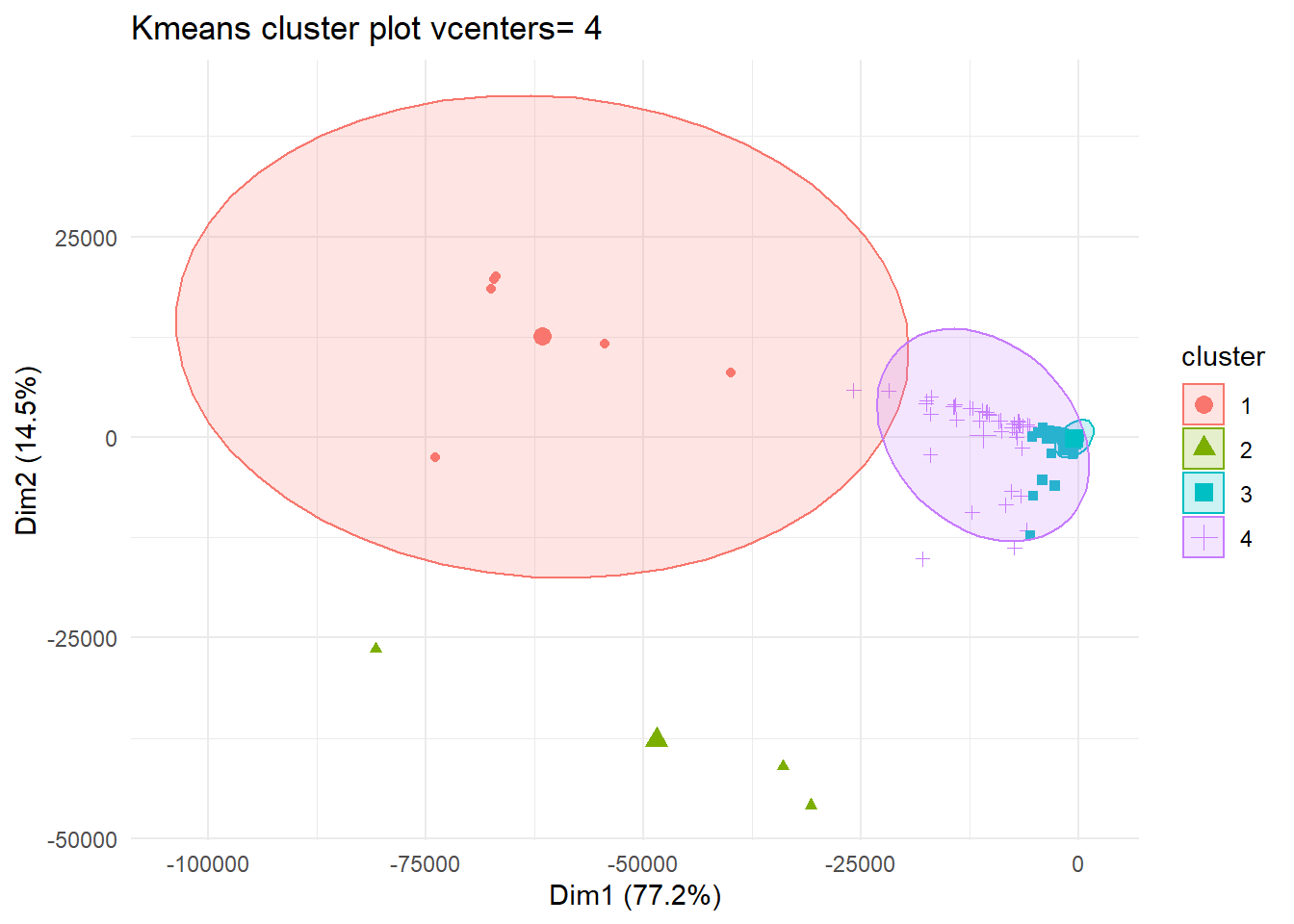
## betweenss 1 -none- numeric

## size 3 -none- numeric

## iter 1 -none- numeric

## ifault 1 -none- numeric

## [[1]]



##

## [[2]]

## Length Class Mode

## cluster 375 -none- numeric

## centers 296 -none- numeric

## totss 1 -none- numeric

## withinss 4 -none- numeric

## tot.withinss 1 -none- numeric

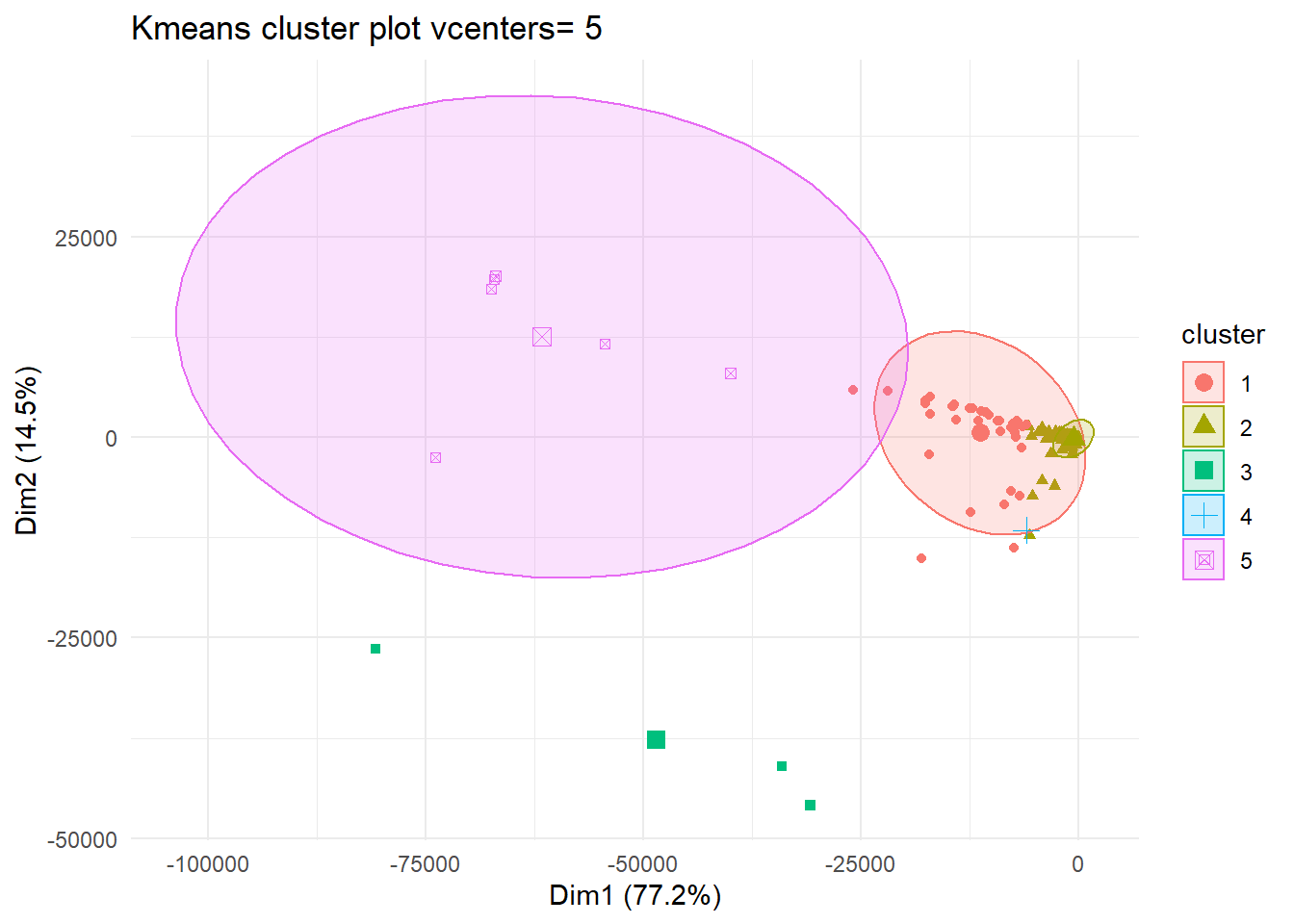
## betweenss 1 -none- numeric

## size 4 -none- numeric

## iter 1 -none- numeric

## ifault 1 -none- numeric

## [[1]]



##

## [[2]]

## Length Class Mode

## cluster 375 -none- numeric

## centers 370 -none- numeric

## totss 1 -none- numeric

## withinss 5 -none- numeric

## tot.withinss 1 -none- numeric

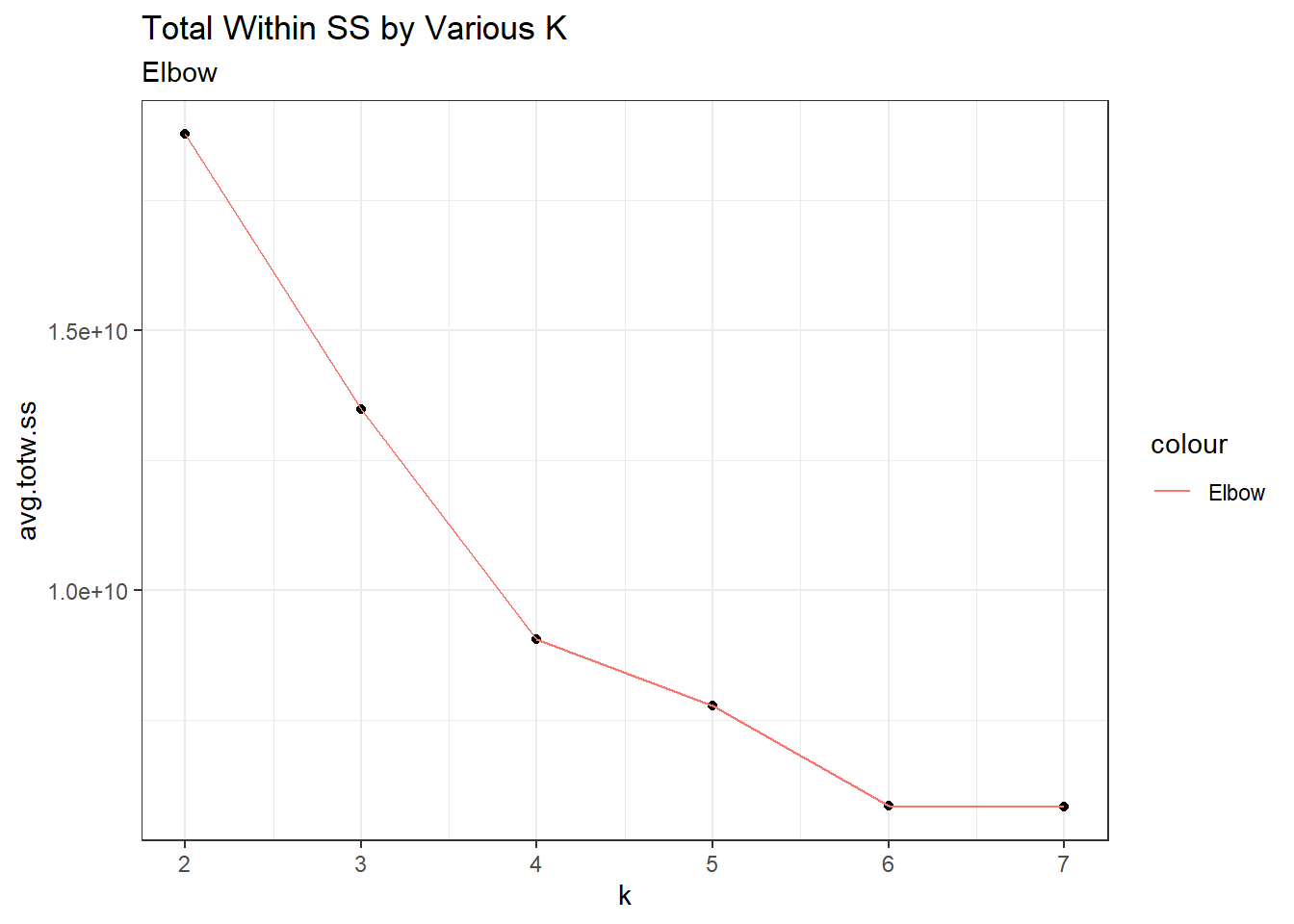
## betweenss 1 -none- numeric

## size 5 -none- numeric

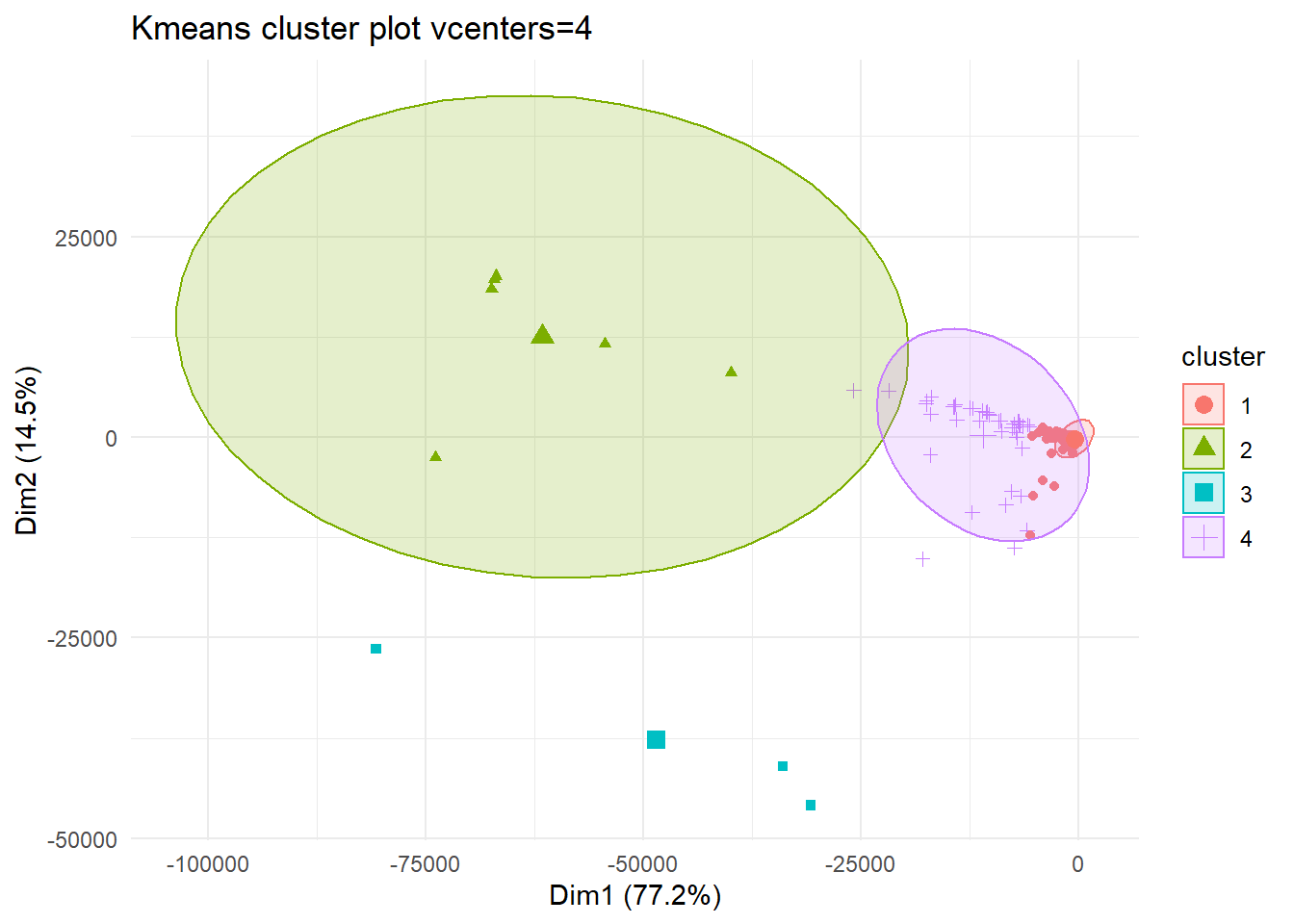
## iter 1 -none- numeric

## ifault 1 -none- numeric

### Elbow Plot Result

The plot represents the variance within the clusters. It decreases as k increases, and a bend (or “elbow”) can be seen at k = 4. This bend indicates that additional clusters beyond the fourth have little value. In the next section, we’ll classify the observations into 4 clusters.  ### K-means Results

Results using k=4 clustering with patient age and 74 biomarkers

 Samples from the 4 clusters are displayed below in descending order. Only 5 of the 74 biomarkers are displayed. Clusters 1 and 4 had the most observations and we know there were not that many important features out of the 74 biomarkers. This may mean clusters 3 and 2 were defined by how the important features set them apart from the rest.

* **hypersensitive\_cardiac\_troponin\_i**: This is the preferred biomarker for acute myocardial infarction. This helps diagnose acute coronary syndromes in those who are critically ill.
* **amino\_terminal\_brain\_natriuretic\_peptide\_precursor\_nt\_pro\_bnp**: Natriuretic peptide tests measure the presence of brain natriuretic peptide and B-terminal pro b-type natriuretic peptides present. These are substances created by the heart. Small levels are normally found in the bloodstream. High levels of BNP, NT-proBNP indicate that the heart is failing to pump the appropriate amount of blood needed.
* **lactate\_dehydrogenase**: LDH is an enzyme released when there is tissue damage present in a patient’s body tissues. This marker measures the amount of LDH in blood or other body fluids and is vital in monitoring viral conditions or cancer.
* **ferritin**: Ferritin is a major intracellular iron storage protein in all organisms. It binds free ions of the trace element, neutralizing its toxic properties and increasing its solubility. In the soluble form, the body can expend iron as needed, in particular for regulation of cellular oxygen metabolism. Elevated levels of ferritin, or hyperferritinemia, indicate the presence of viruses and bacteria into the body.

k=4 cluster dimensions:

* [1] 324, 12 - Cluster one is the largest
* [2] 3, 12 - Cluster two is the smallest
* [3] 42, 12 - Cluster three is the second largest
* [4] 6, 12 - Cluster four is very small as well

The smaller clusters have elevated markers, while the larger clusters do not. Note: long biomarker names have been abbreviated to 25 characters.

| **cluster** | **lactate\_dehydrogenase** | **amino\_terminal\_brain\_natr** | **hypersensitive\_cardiac\_tr** | **high\_sensitivity\_c\_reacti** |
| --- | --- | --- | --- | --- |
| 1 | 306 | 60 | 19.9 | 43.1 |
| 1 | 738 | 173 | 16.9 | 27.4 |
| 1 | 328 | 0 | 0.0 | 42.3 |
| 1 | 338 | 152 | 4.8 | 108.2 |
| 1 | 195 | 5 | 5.6 | 7.0 |
| 1 | 268 | 645 | 19.7 | 29.3 |
| **cluster** | **lactate\_dehydrogenase** | **amino\_terminal\_brain\_natr** | **hypersensitive\_cardiac\_tr** | **high\_sensitivity\_c\_reacti** |
| 2 | 284 | 70000 | 85.5 | 81.7 |
| 2 | 1752 | 40096 | 2.6 | 84.6 |
| 2 | 1867 | 70000 | 523.1 | 36.7 |
| 2 | 1126 | 55300 | 3404.7 | 138.8 |
| 2 | 1867 | 70000 | 21953.4 | 259.1 |
| 2 | 702 | 70000 | 1810.1 | 0.0 |

| **cluster** | **lactate\_dehydrogenase** | **amino\_terminal\_brain\_natr** | **hypersensitive\_cardiac\_tr** | **high\_sensitivity\_c\_reacti** |
| --- | --- | --- | --- | --- |
| 3 | 1867 | 20906 | 50000 | 113.1 |
| 3 | 1500 | 70000 | 50000 | 87.3 |
| 3 | 785 | 16243 | 50000 | 192.3 |
| **cluster** | **lactate\_dehydrogenase** | **amino\_terminal\_brain\_natr** | **hypersensitive\_cardiac\_tr** | **high\_sensitivity\_c\_reacti** |
| 4 | 208 | 11017 | 0.0 | 24.3 |
| 4 | 187 | 13006 | 124.0 | 2.1 |
| 4 | 310 | 22525 | 617.0 | 160.2 |
| 4 | 1772 | 9310 | 46.3 | 274.1 |
| 4 | 890 | 7447 | 304.2 | 86.1 |
| 4 | 492 | 11534 | 1317.8 | 196.9 |

Statistical summary of most important biomarkers.

## [1] 375 6

## [1] "cluster" "lactate\_dehydrogenase"

## [3] "amino\_terminal\_brain\_natr" "hypersensitive\_cardiac\_tr"

## [5] "high\_sensitivity\_c\_reacti" "ferritin"

## [1] "==============================================================="

## [1] "---------------------------------------------------------------"

## [1] "cluster"

## [1] "---------------------------------------------------------------"

## [1] "mean: 1.368"

## [1] "median: 1"

## [1] "min: 1"

## [1] "max: 4"

## [1] "range: 3"

## [1] "sd: 0.960748371403814"

## [1] "quantile: 1" "quantile: 1" "quantile: 1" "quantile: 1" "quantile: 4"

## [1] "IQR: 0"

## [1] "---------------------------------------------------------------"

## [1] "lactate\_dehydrogenase"

## [1] "---------------------------------------------------------------"

## [1] "mean: 519.592"

## [1] "median: 355"

## [1] "min: 0"

## [1] "max: 1867"

## [1] "range: 1867"

## [1] "sd: 436.03199416537"

## [1] "quantile: 0" "quantile: 239.5" "quantile: 355" "quantile: 661"

## [5] "quantile: 1867"

## [1] "IQR: 421.5"

## [1] "---------------------------------------------------------------"

## [1] "amino\_terminal\_brain\_natr"

## [1] "---------------------------------------------------------------"

## [1] "mean: 2811.68533333333"

## [1] "median: 99"

## [1] "min: 0"

## [1] "max: 70000"

## [1] "range: 70000"

## [1] "sd: 9387.95662943333"

## [1] "quantile: 0" "quantile: 0" "quantile: 99" "quantile: 941"

## [5] "quantile: 70000"

## [1] "IQR: 941"

## [1] "---------------------------------------------------------------"

## [1] "hypersensitive\_cardiac\_tr"

## [1] "---------------------------------------------------------------"

## [1] "mean: 887.5552"

## [1] "median: 4.8"

## [1] "min: 0"

## [1] "max: 50000"

## [1] "range: 50000"

## [1] "sd: 4933.42370112608"

## [1] "quantile: 0" "quantile: 1.9" "quantile: 4.8" "quantile: 47.2"

## [5] "quantile: 50000"

## [1] "IQR: 45.3"

## [1] "---------------------------------------------------------------"

## [1] "high\_sensitivity\_c\_reacti"

## [1] "---------------------------------------------------------------"

## [1] "mean: 86.6608"

## [1] "median: 54.1"

## [1] "min: 0"

## [1] "max: 320"

## [1] "range: 320"

## [1] "sd: 87.9859750272583"

## [1] "quantile: 0" "quantile: 10.1" "quantile: 54.1" "quantile: 144.4"

## [5] "quantile: 320"

## [1] "IQR: 134.3"

## [1] "---------------------------------------------------------------"

## [1] "ferritin"

## [1] "---------------------------------------------------------------"

## [1] "mean: 883.942933333333"

## [1] "median: 147.3"

## [1] "min: 0"

## [1] "max: 50000"

## [1] "range: 50000"

## [1] "sd: 3052.68673704692"

## [1] "quantile: 0" "quantile: 0" "quantile: 147.3"

## [4] "quantile: 881.95" "quantile: 50000"

## [1] "IQR: 881.95"

|  |  |
| --- | --- |
|  |  |
|  |  |

## Principal Component Analysis

Principal Component Analysis (PCA) is a multivariate technique that allows us to summarize the systematic patterns of variations in the data.

From a data analysis standpoint, PCA is used for studying one table of observations and variables with the main idea of transforming the observed variables into a set of new variables, the principal components, which are uncorrelated and explain the variation in the data. For this reason, PCA allows us to reduce a “complex” data set to a lower dimension in order to reveal the structures or the dominant types of variations in both the observations and the variables. Note: starts by setting with clean data set.

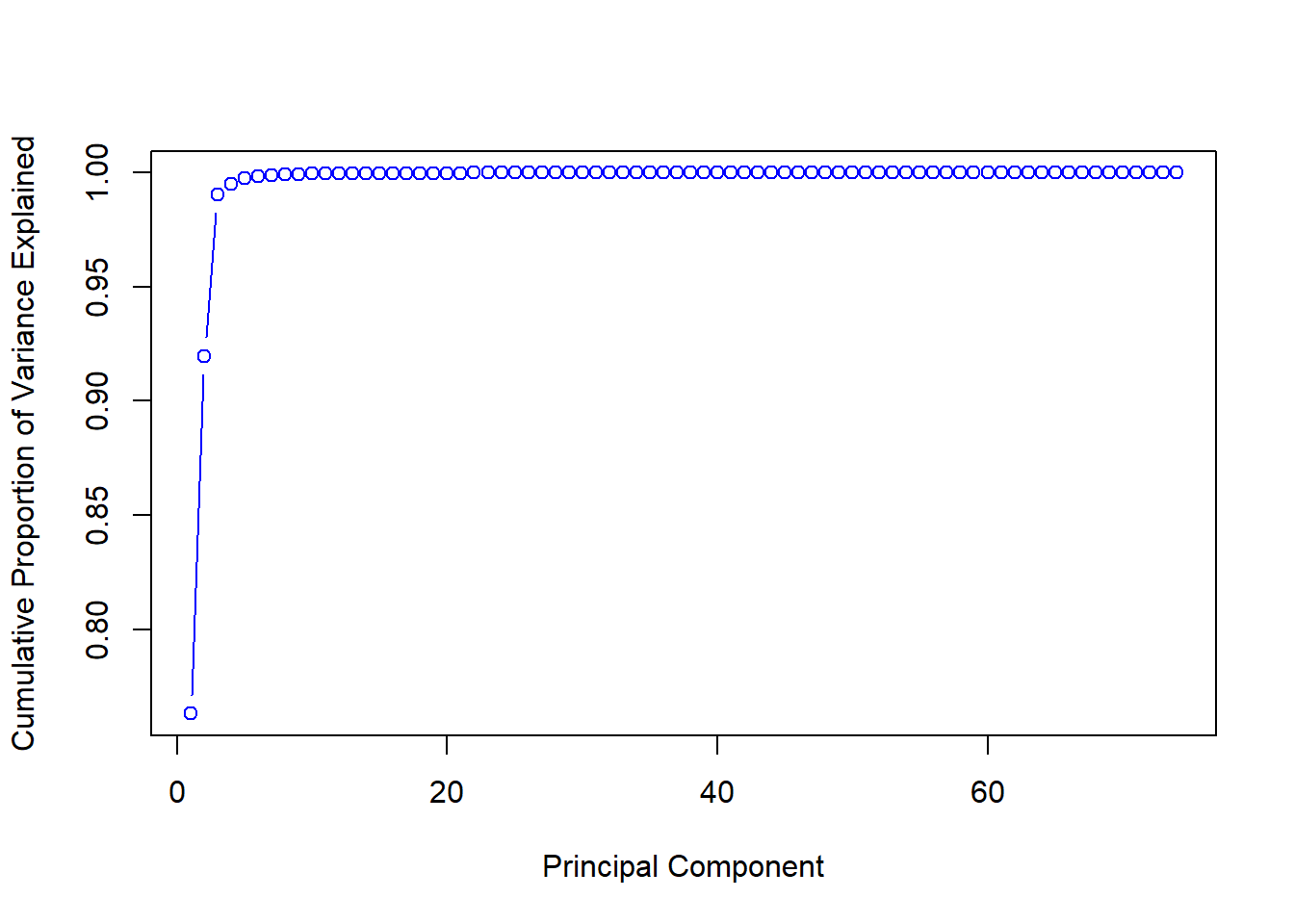
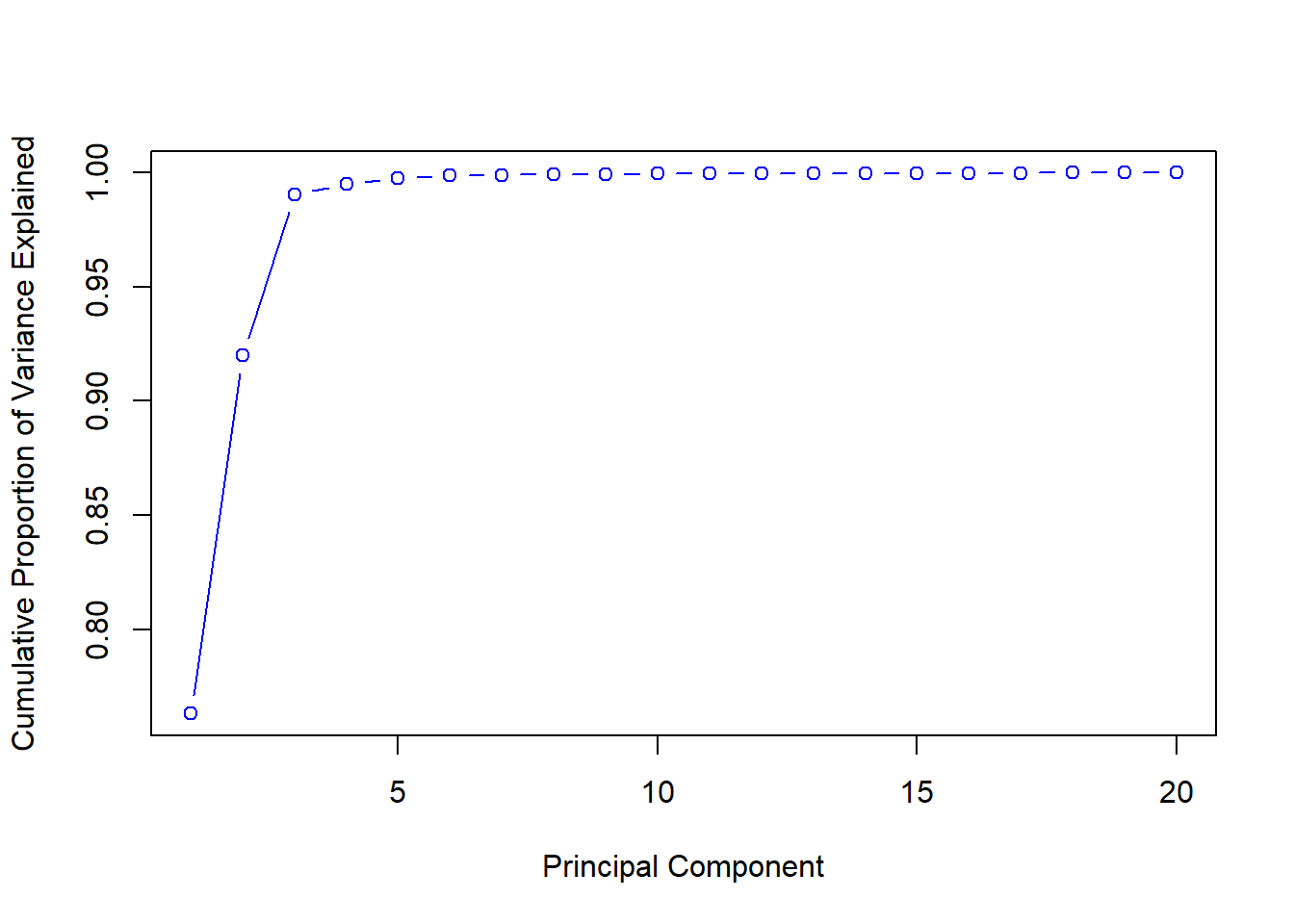
| **Survived** | **hypersensitive\_cardiac\_troponin\_i** | **hemoglobin** | **serum\_chloride** | **prothrombin\_time** |
| --- | --- | --- | --- | --- |
| Deceased | 19.9 | 140 | 103.1 | 14.1 |
| Deceased | 16.9 | 151 | 100.5 | 14.3 |
| Deceased | 0.0 | 126 | 102.9 | 13.6 |
| Deceased | 4.8 | 110 | 103.1 | 16.3 |
| Deceased | 5.6 | 134 | 102.2 | 14.6 |
| Deceased | 19.7 | 108 | 105.8 | 12.4 |

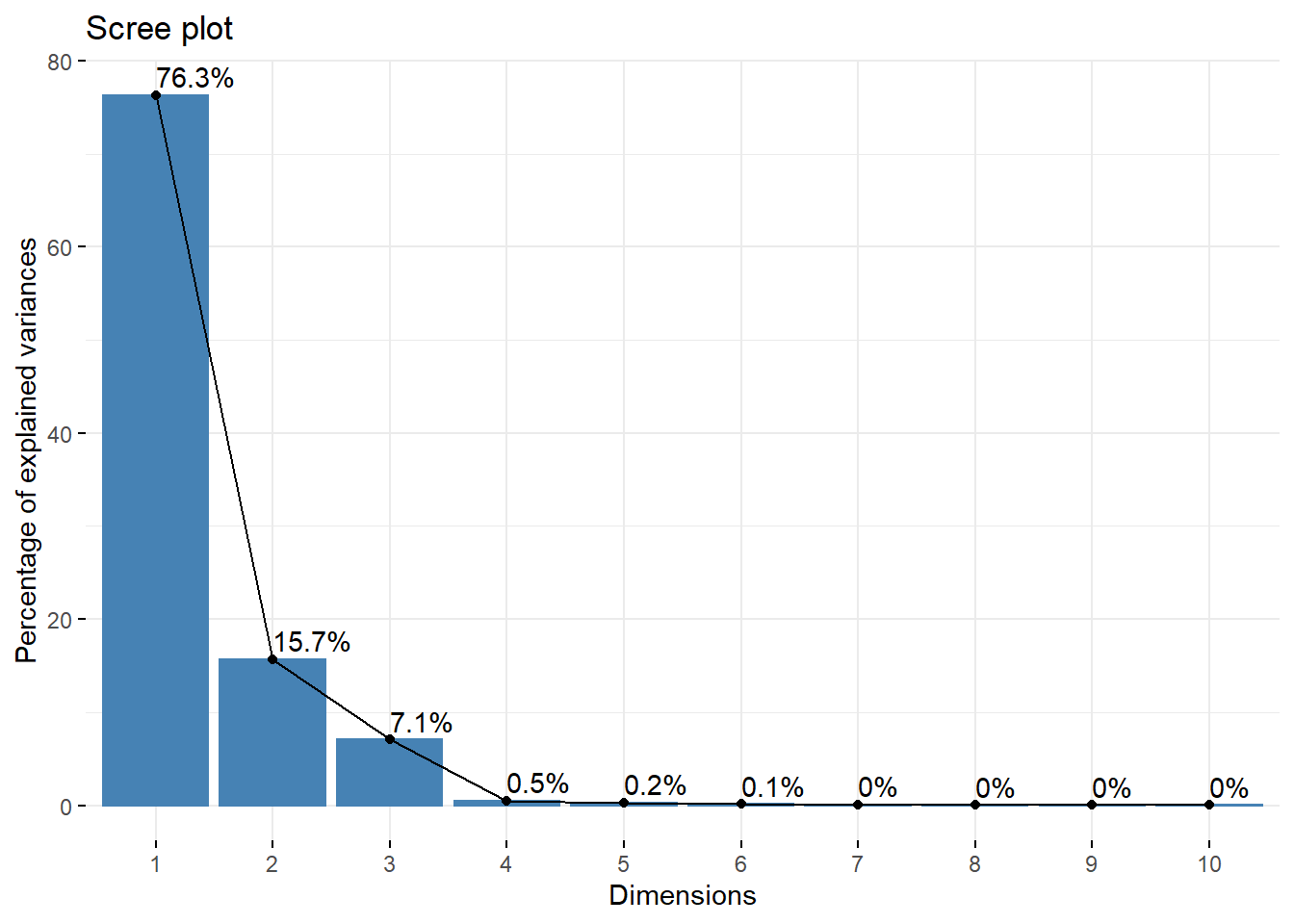
There are several methods for doing PCA using R, but we’re going use princomp method, since it's in base R stats.

Typical PCA results should consist of the following:

* A set of eigenvalues
* A table with the scores or Principal Components (PCs)
* A table of loadings (or correlations between variables and PCs).

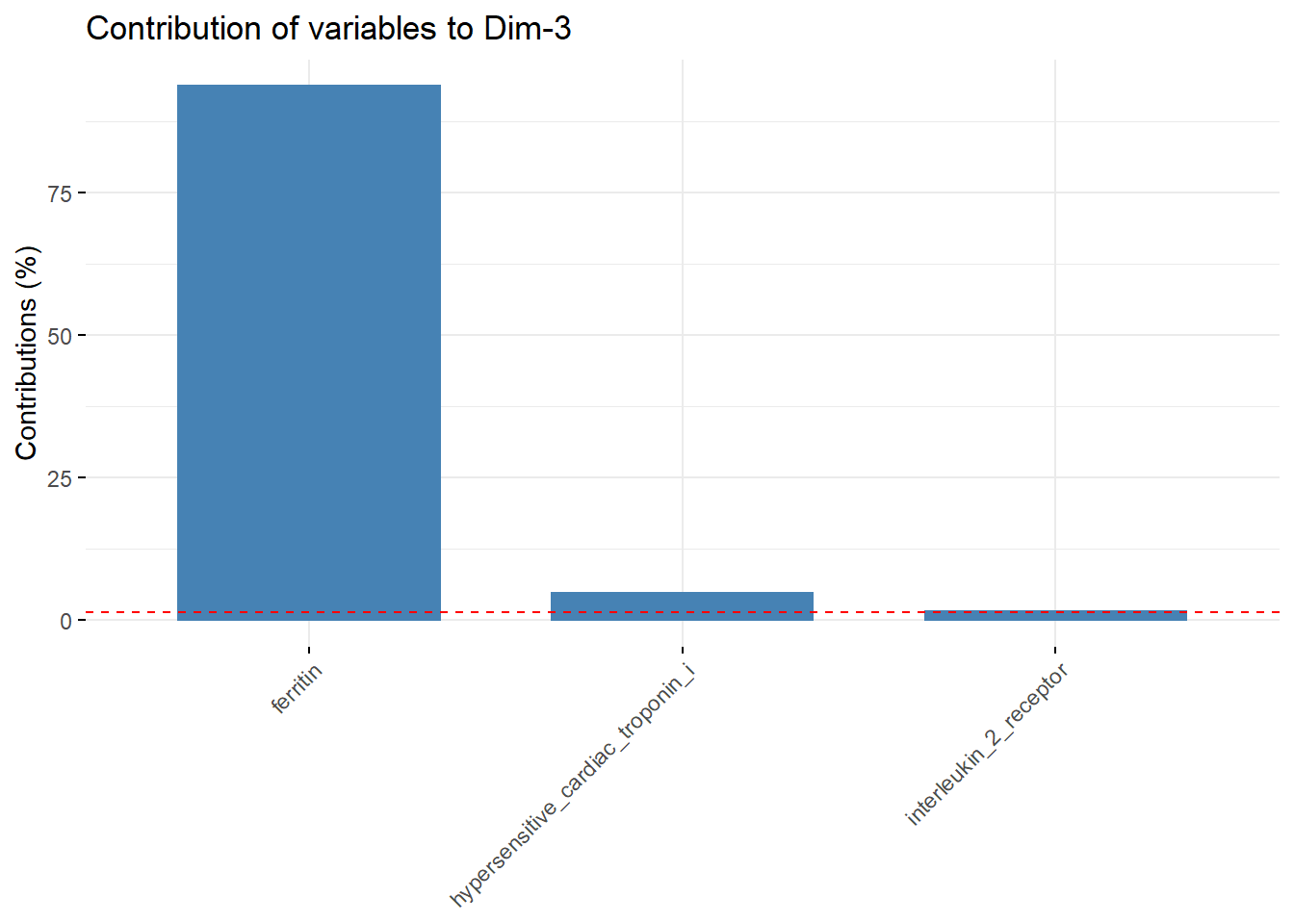
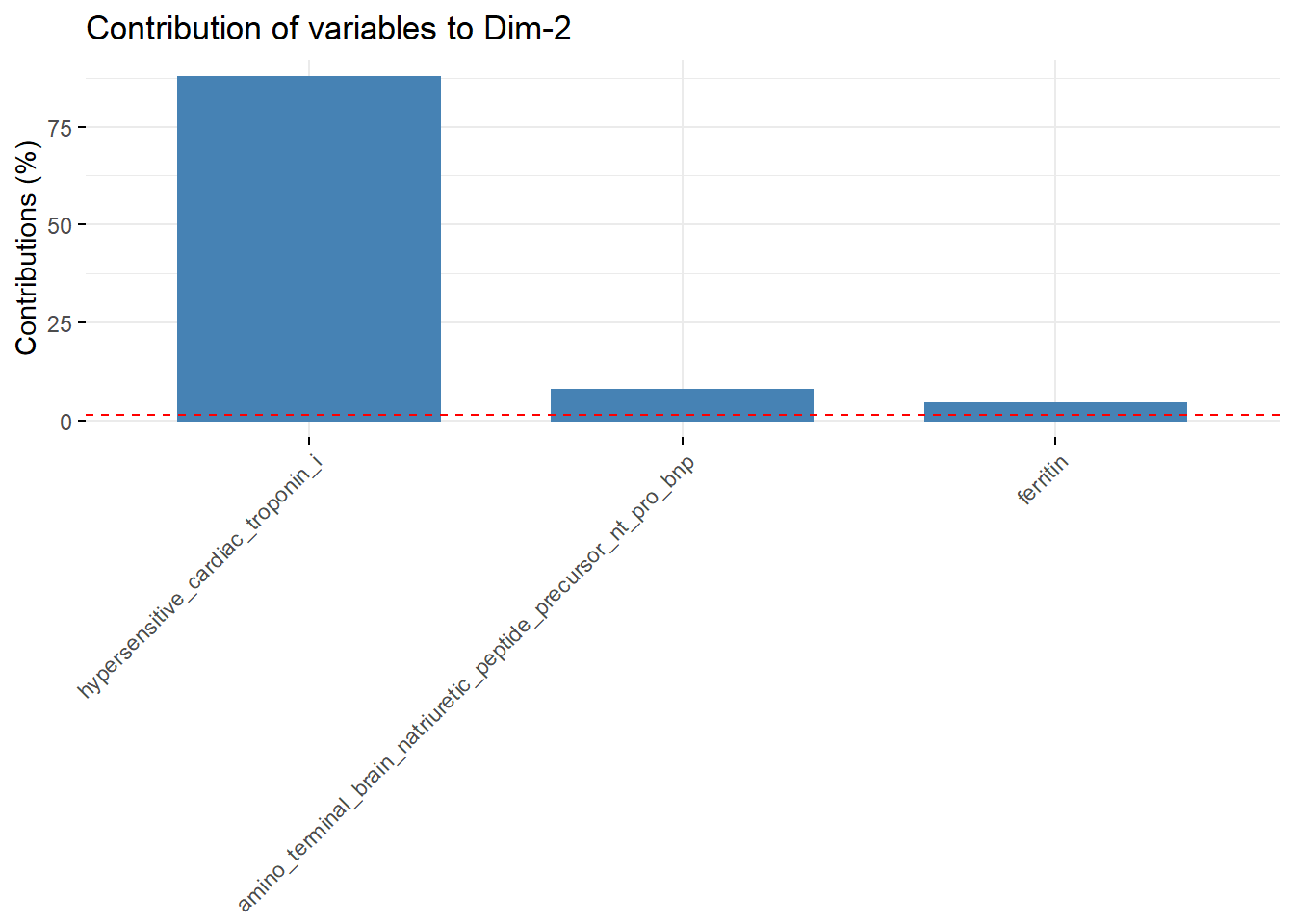
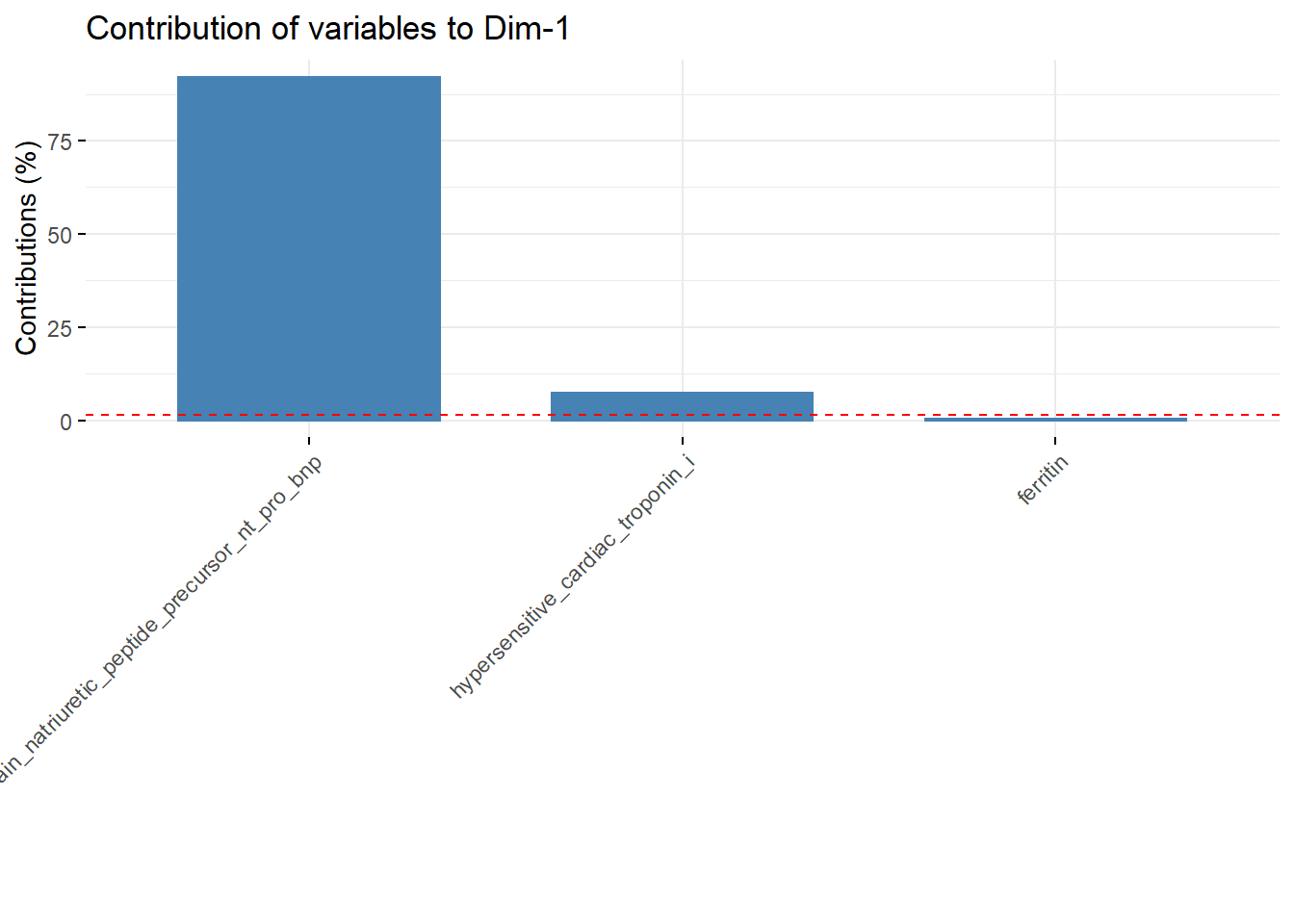
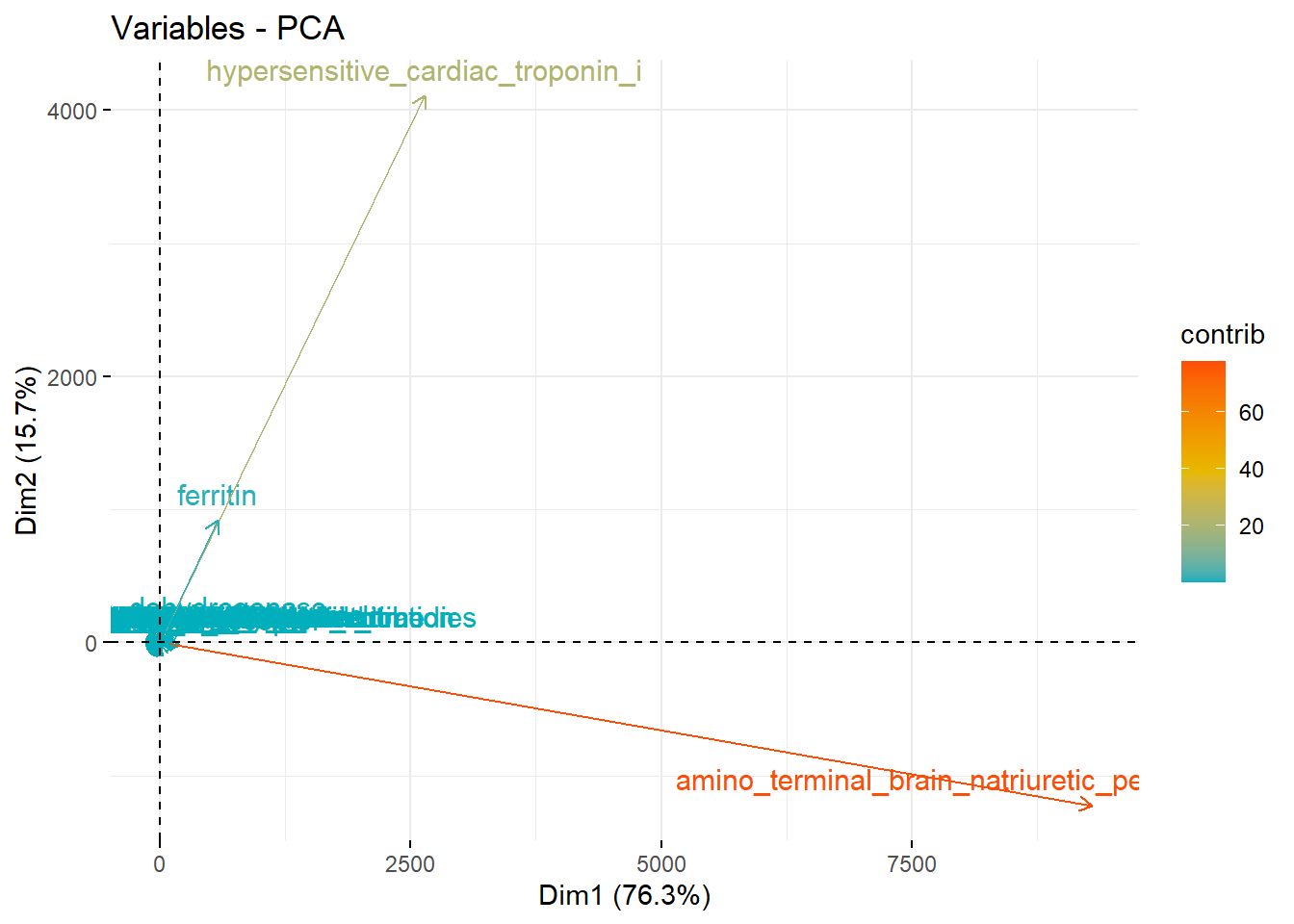
The eigenvalues provide information of the variability in the data. The scores provide information about the structure of the observations. The loadings (or correlations) allow you to get a sense of the relationships between variables, as well as their associations with the extracted PCs.

As seen in the plot below, nearly first 15 components explain most variation in the data. The second plot indicates 15-20 components might be adequate for the modeling data sets.  

The scree plot shows the first component is explaining 76.3% of the variances. The line drops of quickly from there, which bolsters the case that we can use far less than 74 features to get a good prediction.  Since the majority of tha variance are explained by the first three components a contribution analysis is performed on them here.

Top contributors:

* **hypersensitive\_cardiac\_troponin\_i**: This is the preferred biomarker for acute myocardial infarction. This helps diagnose acute coronary syndromes in those who are critically ill.
* **amino\_terminal\_brain\_natriuretic\_peptide\_precursor\_nt\_pro\_bnp**: Natriuretic peptide tests measure the presence of brain natriuretic peptide and B-terminal pro b-type natriuretic peptides present. These are substances created by the heart. Small levels are normally found in the bloodstream. High levels of BNP, NT-proBNP indicate that the heart is failing to pump the appropriate amount of blood needed.
* **lactate\_dehydrogenase**: LDH is an enzyme released when there is tissue damage present in a patient’s body tissues. This marker measures the amount of LDH in blood or other body fluids and is vital in monitoring viral conditions or cancer.
* **ferritin**: Ferritin is a major intracellular iron storage protein in all organisms. It binds free ions of the trace element, neutralizing its toxic properties and increasing its solubility. In the soluble form, the body can expend iron as needed, for regulation of cellular oxygen metabolism. Elevated levels of ferritin, or hyperferritinemia, indicate the presence of viruses and bacteria into the body.



Checking for imbalance with respect to the dependent variable “Survived.” This should not be a problem with default method of creating the model training and test sets since there is good balance of fatalities versus survived in our data set.

## Survived Percentage

## 1 Deceased 0.536

## 2 Survived 0.464

## Classification Models

Several classification models will be leveraged and compared while using the Covid-19, patient, biomarker data. Classification in this case predicts a binary outcome of case fatality or survival. The models will be trained in supervised mode using the dependent variable we will call label, which holds values 0 for fatal and 1 for survived. The models will then be tested to determine if they can classify case fatality and survival without having the benefit of the labels.

The list of classification models to be used can be viewed below here:

* Decision Tree(rpart)
* Random Forest(rf)
* Naive Bayes(nb)
* K Nearest Neighbors(knn)
* Support Vector Machines(svm)
* Gradient Boosting Machines(gbm)
* Classification Based On Association Rules (cba)

The models will each be described further as they is used.

### Data Splitting

Using the first 20 components for the model splitting after determining that gets the best predictions without losing predictive quality using all 74 features.

## [1] 375 75

## [1] 300 21

## [1] 75 21

The data will be split into 80/20 for training and test sets.

## dimension\_of\_training\_dataset dimension\_of\_testing\_dataset

## 1 300 75

## 2 21 21

## [1] 300 21

## [1] 75 21

## [1] 75 21

### Model Controls

Model controls will consist of cross validation set to 5-fold, and fit metric set to “Accuracy.”

### Model Tuning

Reviewing the models and their different parameters which can be tuned. Caret will be used with method model training controls to run algorithms u5-fold cross validation and various parameter tuning. The most important features are printed after each model is trained for additional information.

## model parameter label forReg forClass

## 1 rpart cp Complexity Parameter TRUE TRUE

## 2 rf mtry #Randomly Selected Predictors TRUE TRUE

## 3 nb fL Laplace Correction FALSE TRUE

## 4 nb usekernel Distribution Type FALSE TRUE

## 5 nb adjust Bandwidth Adjustment FALSE TRUE

## 6 knn k #Neighbors TRUE TRUE

## 7 svmRadial sigma Sigma TRUE TRUE

## 8 svmRadial C Cost TRUE TRUE

## 9 gbm n.trees # Boosting Iterations TRUE TRUE

## 10 gbm interaction.depth Max Tree Depth TRUE TRUE

## 11 gbm shrinkage Shrinkage TRUE TRUE

## 12 gbm n.minobsinnode Min. Terminal Node Size TRUE TRUE

## probModel

## 1 TRUE

## 2 TRUE

## 3 TRUE

## 4 TRUE

## 5 TRUE

## 6 TRUE

## 7 TRUE

## 8 TRUE

## 9 TRUE

## 10 TRUE

## 11 TRUE

## 12 TRUE

### Model Build

Running a similar cross validation experiment using decision trees to compare the results We are now ready to train and test using classifiers. Below we use a few different decision tree models with different parameters and pruning to get varied results.

Caret will be used with method used for model training controls to run algorithms 5-fold cross validation and various parameter tuning.

Note: The most important features are printed after each model is trained for additional information.

#### Decision Tree

* Decision Tree - A tree in which each internal (non-leaf) node is labeled with an input feature. The arcs coming from a node labeled with an input feature are labeled with each of the possible values of the target feature or the arc leads to a subordinate decision node on a different input feature.

## CART

##

## 300 samples

## 20 predictor

## 2 classes: 'Deceased', 'Survived'

##

## No pre-processing

## Resampling: Cross-Validated (5 fold)

## Summary of sample sizes: 240, 240, 241, 239, 240

## Resampling results across tuning parameters:

##

## cp Accuracy Kappa

## 0.04347826 0.8397610 0.6815205

## 0.13768116 0.7701436 0.5319054

## 0.62318841 0.7159063 0.4094905

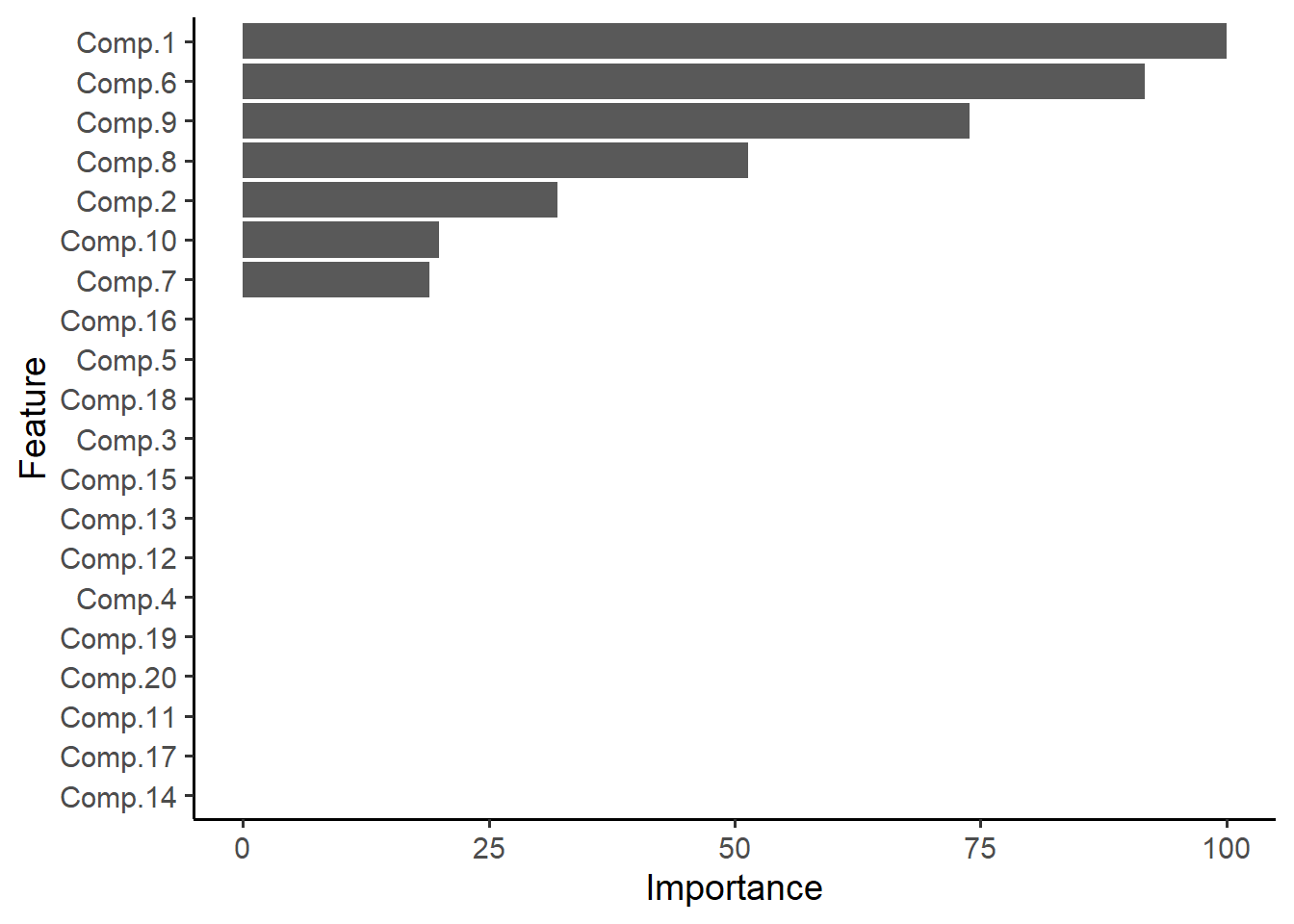
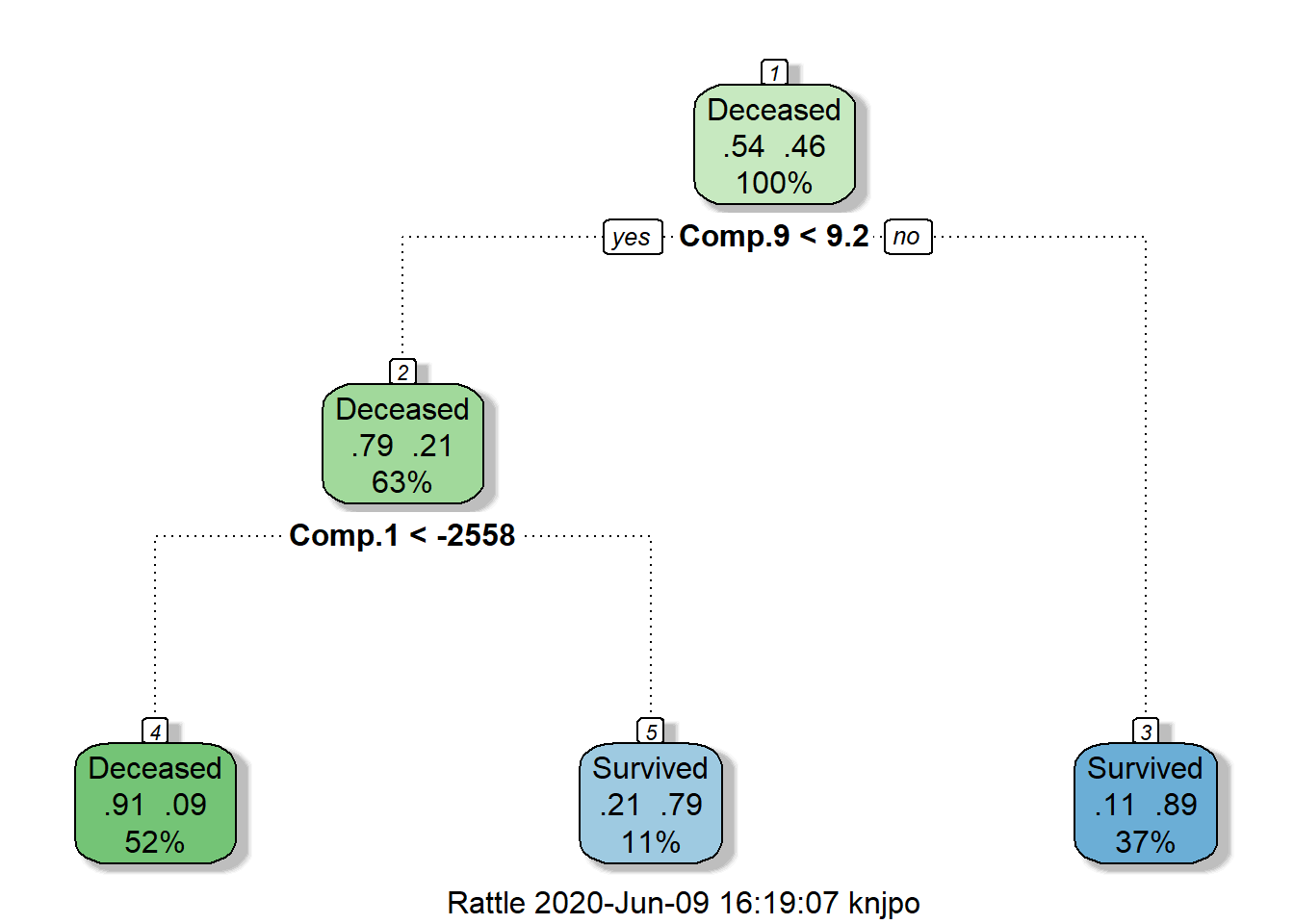
##

## Accuracy was used to select the optimal model using the largest value.

## The final value used for the model was cp = 0.04347826.

## user system elapsed

## 2.25 0.07 2.70



#### Gradient Boosting Machine

* Gradient Boosting Machines - Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees.

## [1] 90

## Stochastic Gradient Boosting

##

## 300 samples

## 20 predictor

## 2 classes: 'Deceased', 'Survived'

##

## No pre-processing

## Resampling: Cross-Validated (5 fold)

## Summary of sample sizes: 241, 239, 241, 240, 239

## Resampling results across tuning parameters:

##

## interaction.depth n.trees Accuracy Kappa

## 1 50 0.8869760 0.7722846

## 1 100 0.8934204 0.7850006

## 1 150 0.9133130 0.8257010

## 1 200 0.9000871 0.7982149

## 1 250 0.9065351 0.8118068

## 1 300 0.9132583 0.8254766

## 1 350 0.9068667 0.8121216

## 1 400 0.9232602 0.8457358

## 1 450 0.9034787 0.8054518

## 1 500 0.9133148 0.8258101

## 1 550 0.9099250 0.8189050

## 1 600 0.9165916 0.8323539

## 1 650 0.9232602 0.8456811

## 1 700 0.9232055 0.8455723

## 1 750 0.9297629 0.8588103

## 1 800 0.9133695 0.8253909

## 1 850 0.9232055 0.8455723

## 1 900 0.9133695 0.8253909

## 1 950 0.9199268 0.8389179

## 1 1000 0.9068121 0.8118238

## 1 1050 0.9068121 0.8118238

## 1 1100 0.9166481 0.8322628

## 1 1150 0.9098685 0.8186061

## 1 1200 0.8966426 0.7911592

## 1 1250 0.9067009 0.8116982

## 1 1300 0.9164259 0.8316007

## 1 1350 0.9034222 0.8048151

## 1 1400 0.9165370 0.8319092

## 1 1450 0.9067009 0.8116171

## 1 1500 0.9165370 0.8319092

## 5 50 0.9103130 0.8201424

## 5 100 0.9004770 0.7997087

## 5 150 0.9136464 0.8264013

## 5 200 0.9234824 0.8462389

## 5 250 0.9202056 0.8399207

## 5 300 0.9103695 0.8189685

## 5 350 0.9168723 0.8331420

## 5 400 0.9167611 0.8331009

## 5 450 0.9069251 0.8125138

## 5 500 0.9167611 0.8332795

## 5 550 0.9069251 0.8128074

## 5 600 0.9036464 0.8059687

## 5 650 0.9133713 0.8261328

## 5 700 0.9036464 0.8056365

## 5 750 0.9167611 0.8331009

## 5 800 0.9100926 0.8194051

## 5 850 0.9002566 0.7986684

## 5 900 0.9002566 0.7986684

## 5 950 0.9100926 0.8194051

## 5 1000 0.9035352 0.8055457

## 5 1050 0.9002566 0.7986684

## 5 1100 0.9100926 0.8194051

## 5 1150 0.9035352 0.8055457

## 5 1200 0.9133713 0.8261328

## 5 1250 0.9035352 0.8055457

## 5 1300 0.9035352 0.8055457

## 5 1350 0.9133713 0.8261328

## 5 1400 0.9001454 0.7986578

## 5 1450 0.9099815 0.8192450

## 5 1500 0.9133148 0.8259852

## 9 50 0.9002001 0.7997006

## 9 100 0.9102019 0.8192672

## 9 150 0.9333750 0.8665816

## 9 200 0.9202603 0.8396702

## 9 250 0.9235389 0.8460798

## 9 300 0.9268176 0.8530253

## 9 350 0.9137029 0.8260286

## 9 400 0.9268176 0.8530253

## 9 450 0.9234843 0.8464329

## 9 500 0.9136482 0.8258457

## 9 550 0.9103695 0.8189685

## 9 600 0.9069797 0.8121997

## 9 650 0.9168158 0.8332545

## 9 700 0.9069797 0.8121997

## 9 750 0.9102584 0.8190770

## 9 800 0.9167611 0.8331009

## 9 850 0.9167611 0.8331009

## 9 900 0.9167611 0.8331009

## 9 950 0.9134824 0.8263732

## 9 1000 0.9133713 0.8258490

## 9 1050 0.9035352 0.8051123

## 9 1100 0.9002566 0.7984271

## 9 1150 0.8968667 0.7910900

## 9 1200 0.9133713 0.8256077

## 9 1250 0.9067028 0.8123286

## 9 1300 0.9036464 0.8053364

## 9 1350 0.9069251 0.8118805

## 9 1400 0.9001454 0.7981035

## 9 1450 0.9100926 0.8190961

## 9 1500 0.9100926 0.8190961

##

## Tuning parameter 'shrinkage' was held constant at a value of 0.1

##

## Tuning parameter 'n.minobsinnode' was held constant at a value of 20

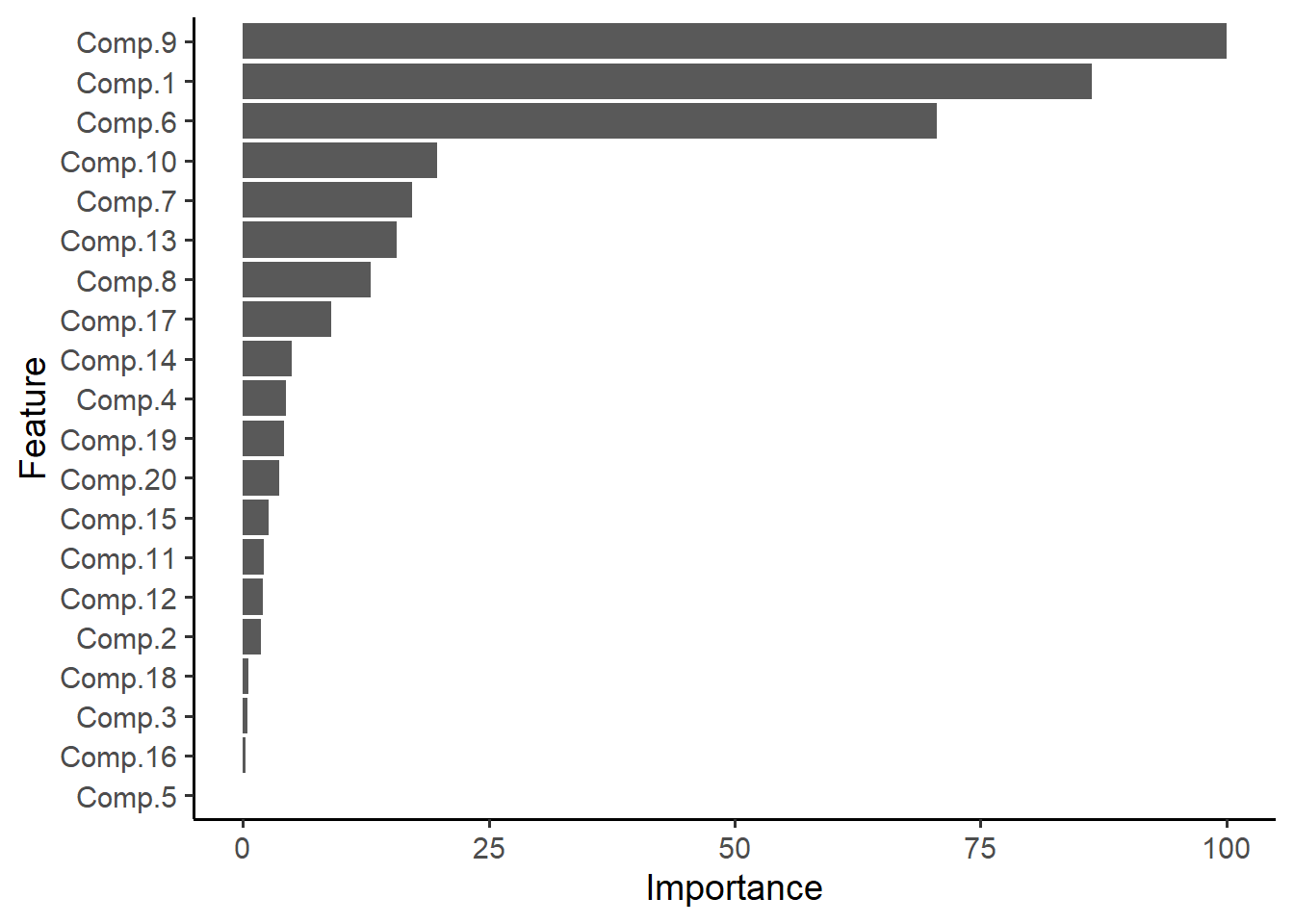
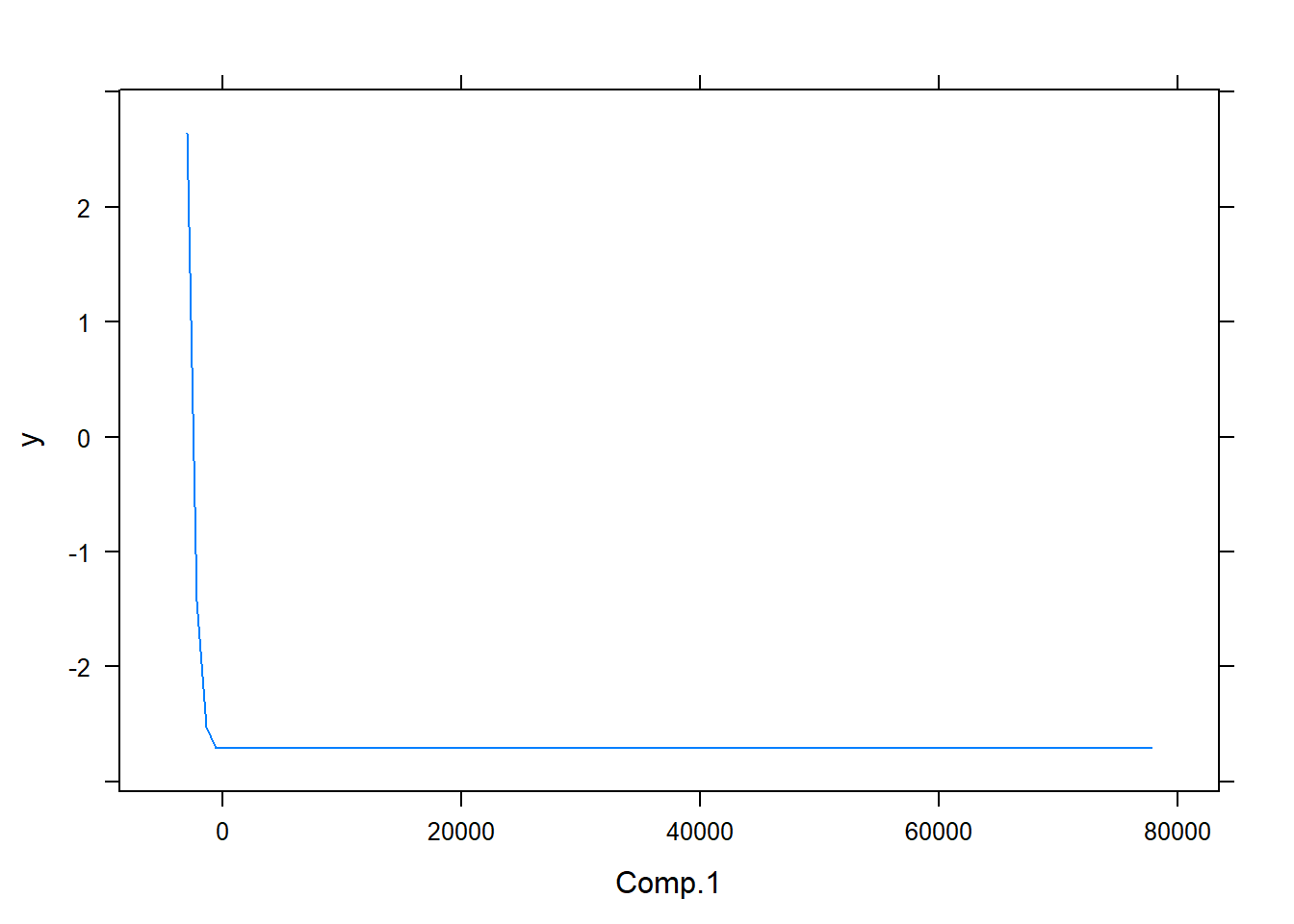
## Accuracy was used to select the optimal model using the largest value.

## The final values used for the model were n.trees = 150, interaction.depth =

## 9, shrinkage = 0.1 and n.minobsinnode = 20.

## user system elapsed

## 18.84 0.16 23.08



#### Random Forest

* Random Forest - An ensemble classifier that consists of many decision trees and outputs the class that is the mode of the classes output by individual trees.

## Random Forest

##

## 300 samples

## 20 predictor

## 2 classes: 'Deceased', 'Survived'

##

## No pre-processing

## Resampling: Cross-Validated (5 fold)

## Summary of sample sizes: 240, 240, 240, 240, 240

## Resampling results across tuning parameters:

##

## mtry Accuracy Kappa

## 2 0.8966667 0.7939077

## 11 0.9166667 0.8328082

## 20 0.9000000 0.7994747

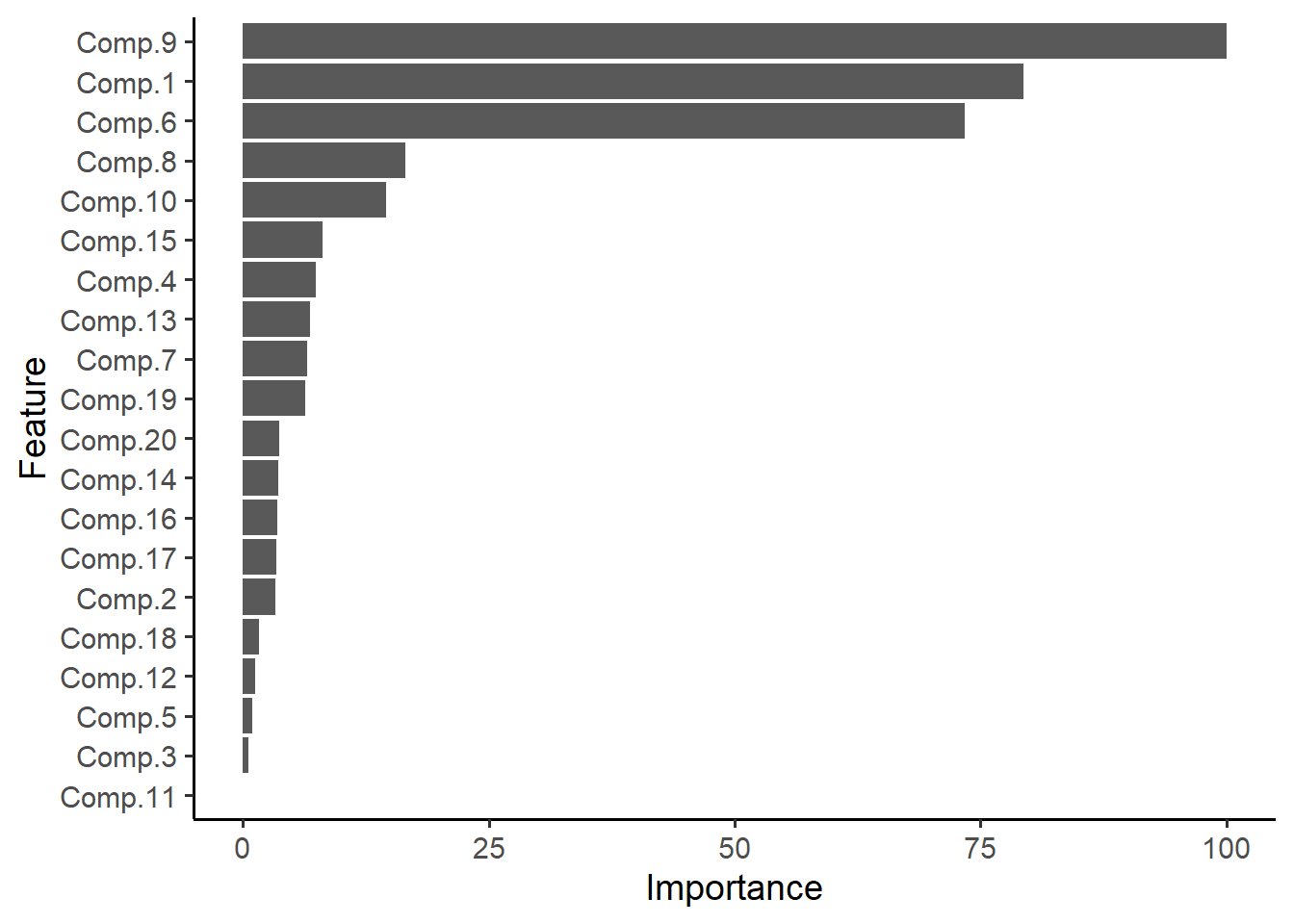
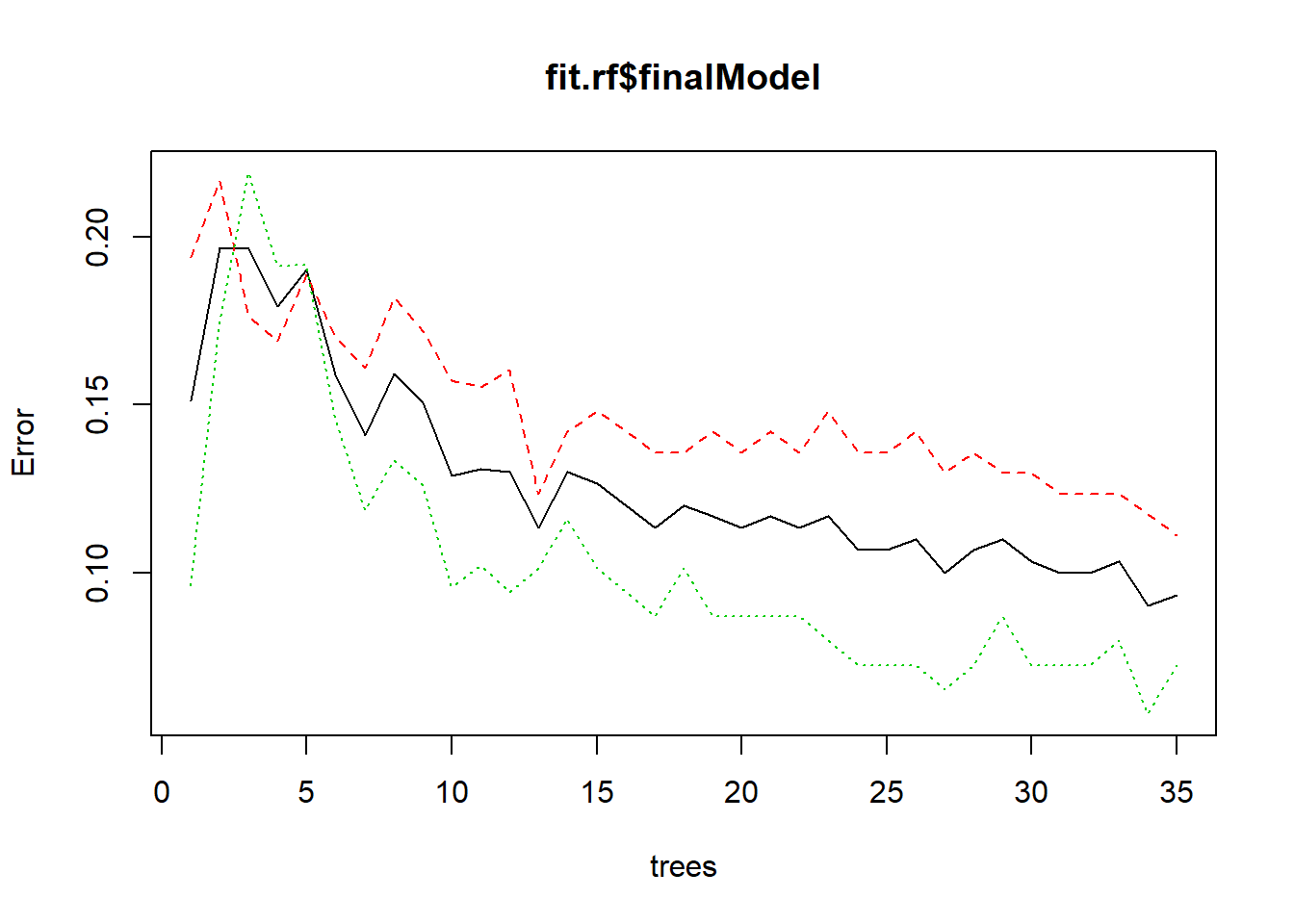
##

## Accuracy was used to select the optimal model using the largest value.

## The final value used for the model was mtry = 11.

## user system elapsed

## 1.78 0.02 1.97



#### Support Vector Machines

* Support Vector Machines (radial) - A discriminative classifier formally defined by a separating hyperplane. The algorithm creates a line or a hyperplane, which separates the data into classes.

## Support Vector Machines with Radial Basis Function Kernel

##

## 300 samples

## 20 predictor

## 2 classes: 'Deceased', 'Survived'

##

## Pre-processing: centered (20), scaled (20)

## Resampling: Cross-Validated (5 fold)

## Summary of sample sizes: 240, 241, 240, 239, 240

## Resampling results across tuning parameters:

##

## C Accuracy Kappa

## 0.25 0.8764351 0.7533862

## 0.50 0.8763786 0.7532587

## 1.00 0.8730453 0.7462505

## 2.00 0.8697666 0.7398198

## 4.00 0.8665444 0.7335186

## 8.00 0.8598759 0.7202493

## 16.00 0.8631546 0.7274778

## 32.00 0.8664879 0.7336732

## 64.00 0.8664879 0.7339260

## 128.00 0.8664879 0.7339260

##

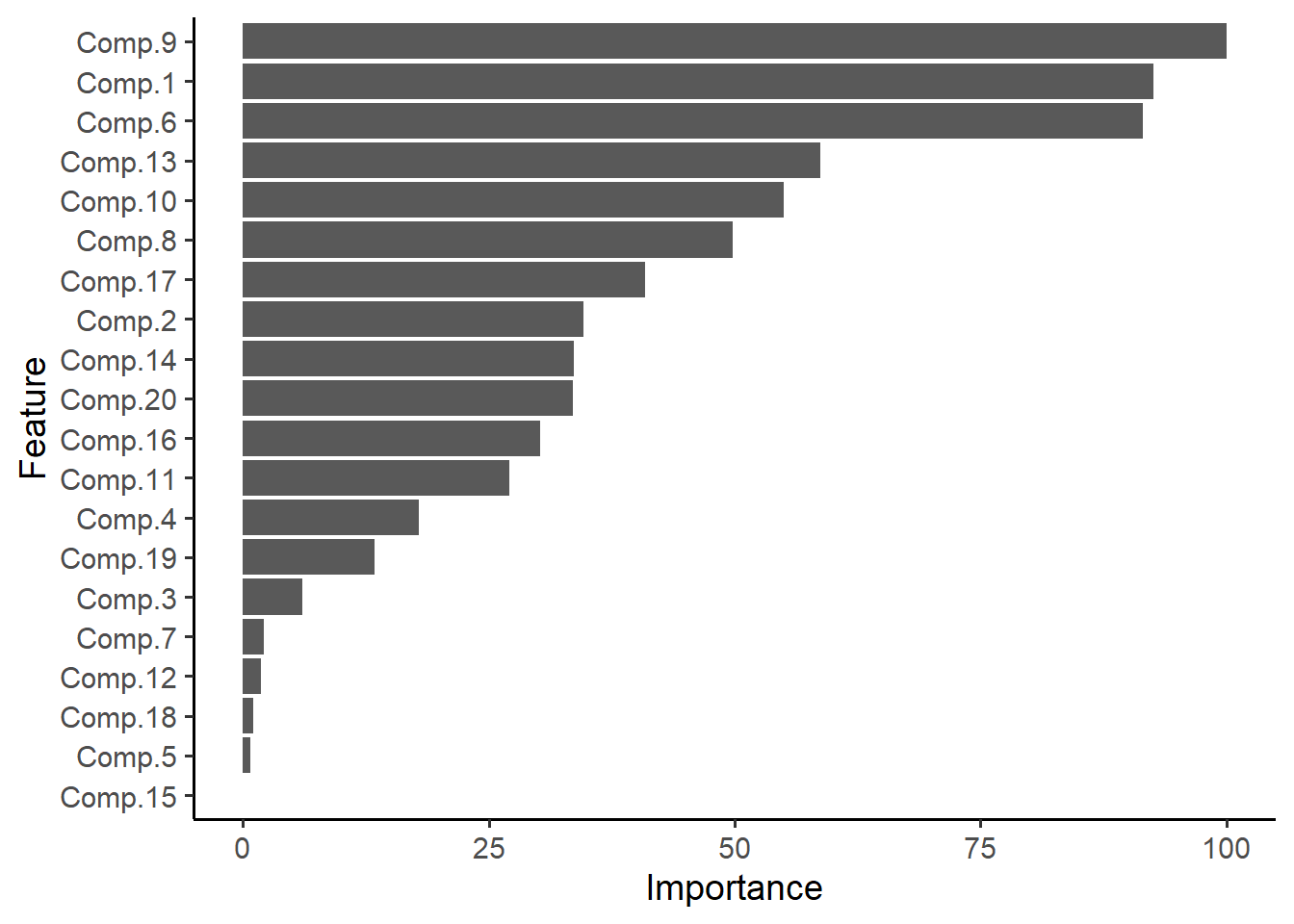
## Tuning parameter 'sigma' was held constant at a value of 0.1594806

## Accuracy was used to select the optimal model using the largest value.

## The final values used for the model were sigma = 0.1594806 and C = 0.25.

## user system elapsed

## 11.11 0.25 13.41



#### Naive Bayes

* Naive Bayes - An algorithm that uses Bayes’ theorem to classify objects. Naive Bayes classifiers assume strong, or naive, independence between attributes of data points.

## Naive Bayes

##

## 300 samples

## 20 predictor

## 2 classes: 'Deceased', 'Survived'

##

## No pre-processing

## Resampling: Cross-Validated (5 fold)

## Summary of sample sizes: 241, 240, 240, 239, 240

## Resampling results across tuning parameters:

##

## usekernel laplace adjust Accuracy Kappa

## FALSE 0.0 0.75 0.7296934 0.4377316

## FALSE 0.0 1.00 0.7296934 0.4377316

## FALSE 0.0 1.25 0.7296934 0.4377316

## FALSE 0.0 1.50 0.7296934 0.4377316

## FALSE 0.5 0.75 0.7296934 0.4377316

## FALSE 0.5 1.00 0.7296934 0.4377316

## FALSE 0.5 1.25 0.7296934 0.4377316

## FALSE 0.5 1.50 0.7296934 0.4377316

## FALSE 1.0 0.75 0.7296934 0.4377316

## FALSE 1.0 1.00 0.7296934 0.4377316

## FALSE 1.0 1.25 0.7296934 0.4377316

## FALSE 1.0 1.50 0.7296934 0.4377316

## TRUE 0.0 0.75 0.8433769 0.6817564

## TRUE 0.0 1.00 0.8366518 0.6682333

## TRUE 0.0 1.25 0.8265370 0.6458675

## TRUE 0.0 1.50 0.8298138 0.6519463

## TRUE 0.5 0.75 0.8433769 0.6817564

## TRUE 0.5 1.00 0.8366518 0.6682333

## TRUE 0.5 1.25 0.8265370 0.6458675

## TRUE 0.5 1.50 0.8298138 0.6519463

## TRUE 1.0 0.75 0.8433769 0.6817564

## TRUE 1.0 1.00 0.8366518 0.6682333

## TRUE 1.0 1.25 0.8265370 0.6458675

## TRUE 1.0 1.50 0.8298138 0.6519463

##

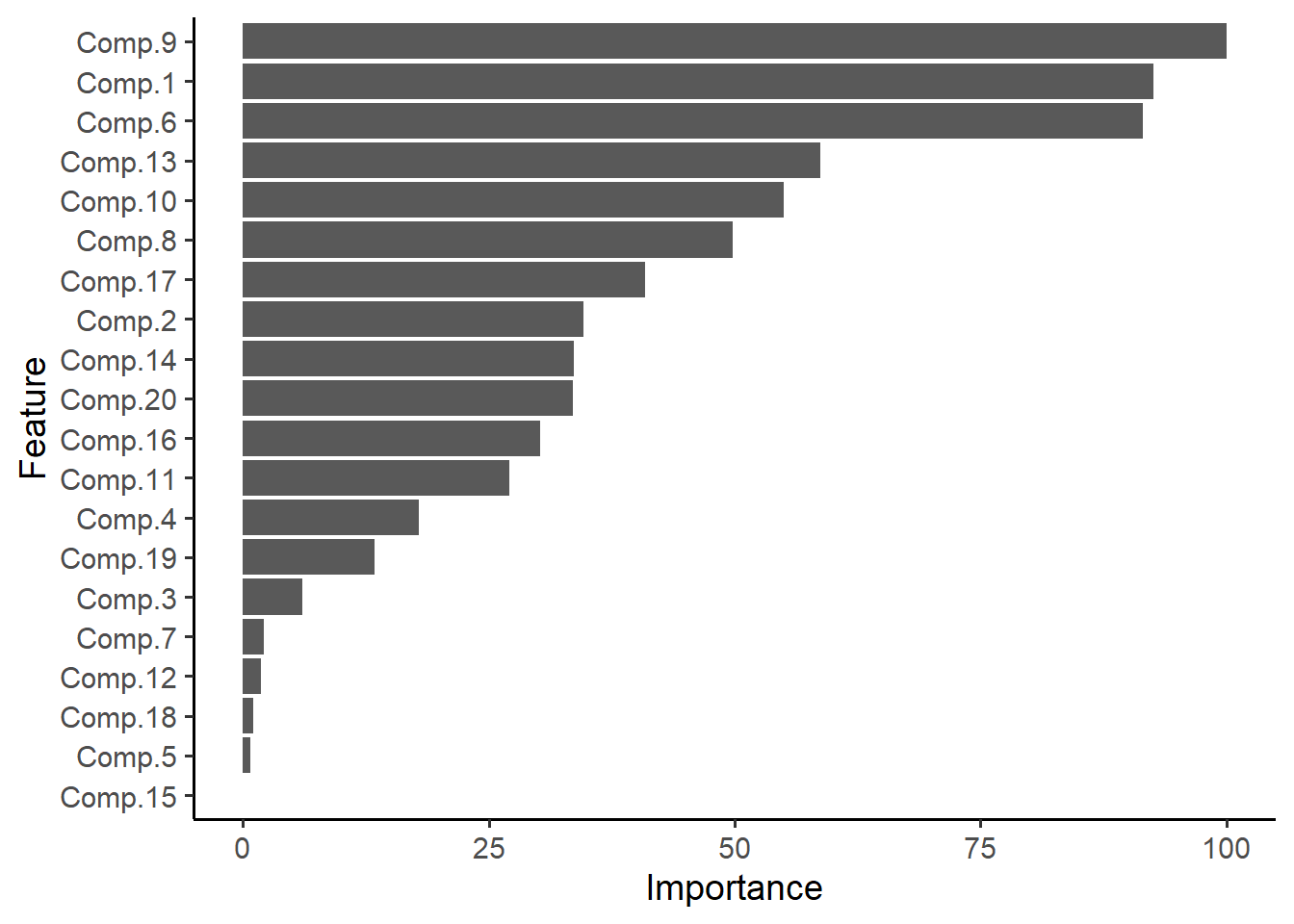
## Accuracy was used to select the optimal model using the largest value.

## The final values used for the model were laplace = 0, usekernel = TRUE

## and adjust = 0.75.

## user system elapsed

## 10.93 0.18 13.21



#### K Nearest Neighbor

* K Nearest Neighbors - KNN has been used in statistical estimation and pattern recognition already in the beginning of 1970’s as a non-parametric technique. It estimates how likely a data point is to be a member of one group or the other depending on what group in which the data points nearest to it. The k-nearest-neighbor is an example of a “lazy learner” algorithm, meaning that it does not build a model using the training set until a query of the data set is performed.

## k-Nearest Neighbors

##

## 300 samples

## 20 predictor

## 2 classes: 'Deceased', 'Survived'

##

## No pre-processing

## Resampling: Cross-Validated (5 fold)

## Summary of sample sizes: 241, 240, 240, 240, 239

## Resampling results across tuning parameters:

##

## k Accuracy Kappa

## 2 0.8667611 0.7312582

## 3 0.8700417 0.7370683

## 5 0.8667083 0.7305085

## 7 0.8534278 0.7028591

## 9 0.8568723 0.7097726

##

## Accuracy was used to select the optimal model using the largest value.

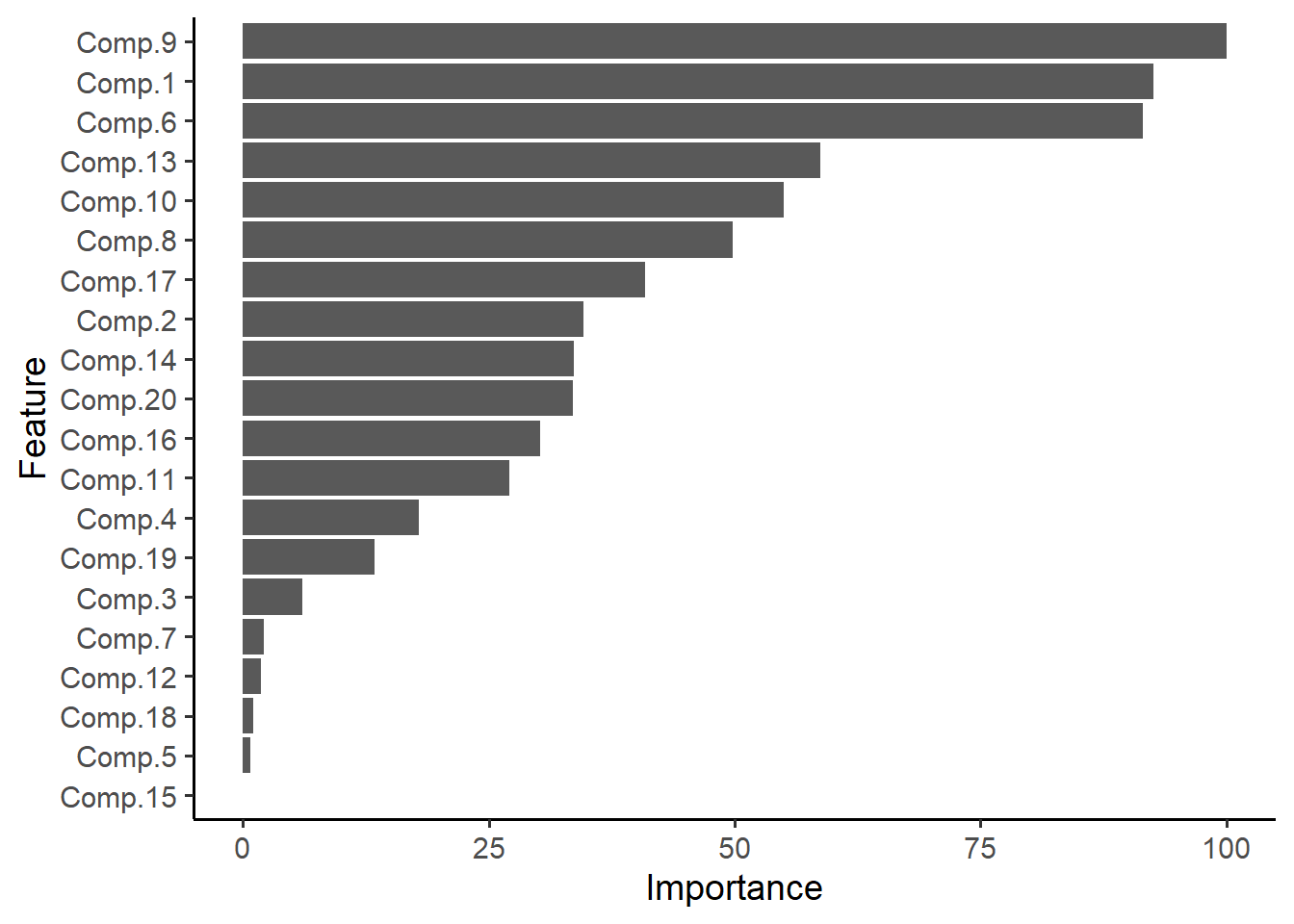
## The final value used for the model was k = 3.

## user system elapsed

## 1.96 0.03 2.28

## k

## 2 3



## Results

### Model Fit Results

The graph below shows two evaluation metrics for the models.

* Accuracy is the percentage of correctly classifies instances out of all instances. It is more useful on a binary classification than multi-class classification problems because it can be less clear exactly how the accuracy breaks down across those classes (e.g. you need to go deeper with a confusion matrix). Learn more about Accuracy here.
* Kappa or Cohen’s Kappa is like classification accuracy, except that it is normalized at the baseline of random chance on your data set. It is a more useful measure to use on problems that have an imbalance in the classes (e.g. 70-30 split for classes 0 and 1 and you can achieve 70% accuracy by predicting all instances are for class 0). Learn more about Kappa here.

The gradient algorithm denoted by gbm is the better model according to these fit metrics, but all models performed reasonably well on the training data set.

##

## Call:

## summary.resamples(object = results)

##

## Models: gbm, rf, tree, nb, svm, knn

## Number of resamples: 5

##

## Accuracy

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

## gbm 0.9016393 0.9166667 0.9322034 0.9333750 0.9491525 0.9672131 0

## rf 0.8666667 0.9166667 0.9166667 0.9166667 0.9333333 0.9500000 0

## tree 0.8000000 0.8135593 0.8500000 0.8397610 0.8500000 0.8852459 0

## nb 0.7666667 0.8166667 0.8500000 0.8433769 0.8852459 0.8983051 0

## svm 0.8000000 0.8666667 0.8813559 0.8764351 0.8833333 0.9508197 0

## knn 0.8166667 0.8500000 0.8688525 0.8700417 0.8813559 0.9333333 0

##

## Kappa

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

## gbm 0.8051118 0.8329621 0.8626310 0.8665816 0.8978650 0.9343380 0

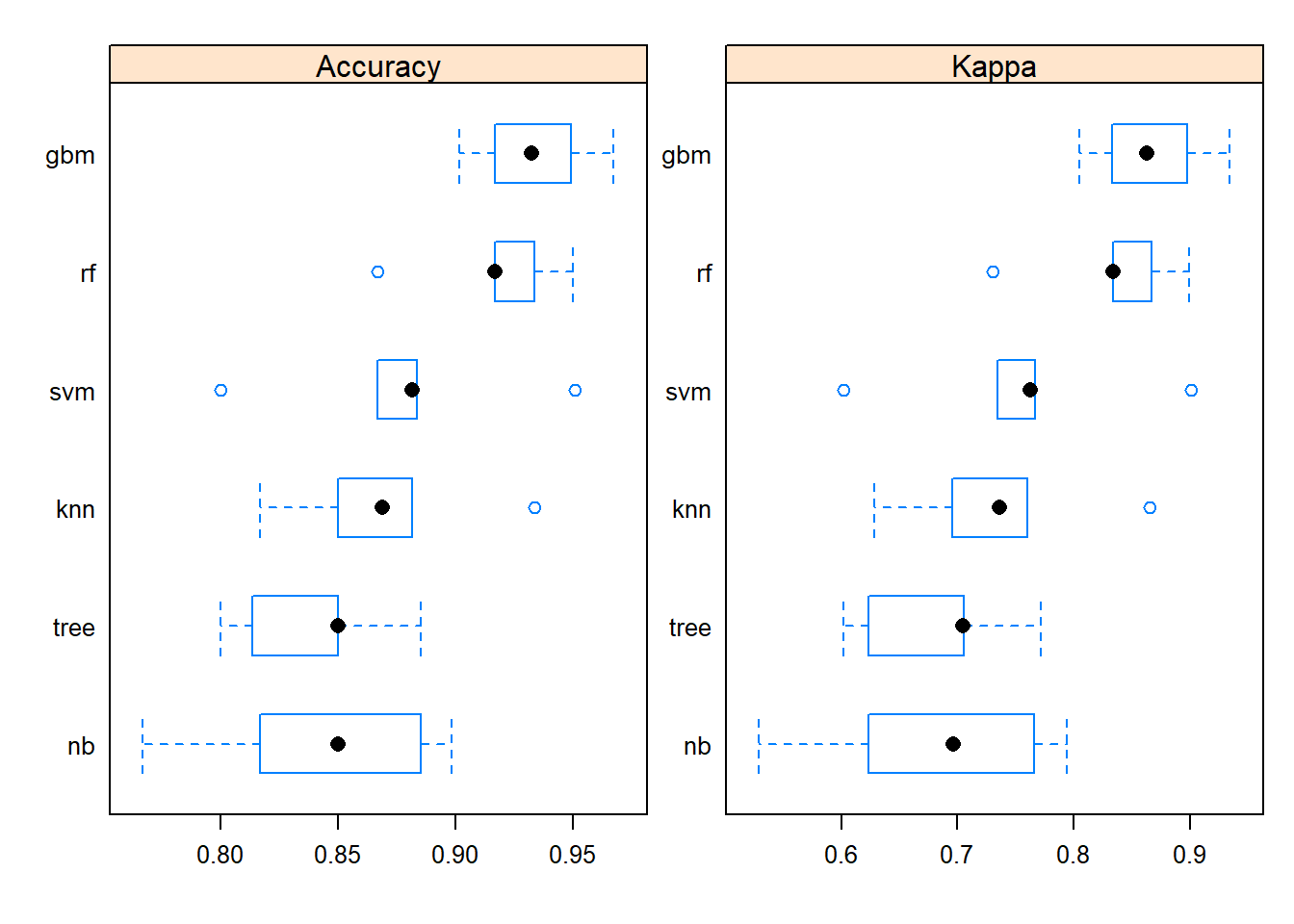
## rf 0.7306397 0.8337029 0.8337029 0.8328082 0.8666667 0.8993289 0

## tree 0.6017699 0.6233314 0.7045952 0.6815205 0.7058824 0.7720235 0

## nb 0.5291480 0.6232877 0.6966292 0.6817564 0.7657707 0.7939464 0

## svm 0.6017699 0.7342193 0.7630522 0.7533862 0.7671840 0.9007054 0

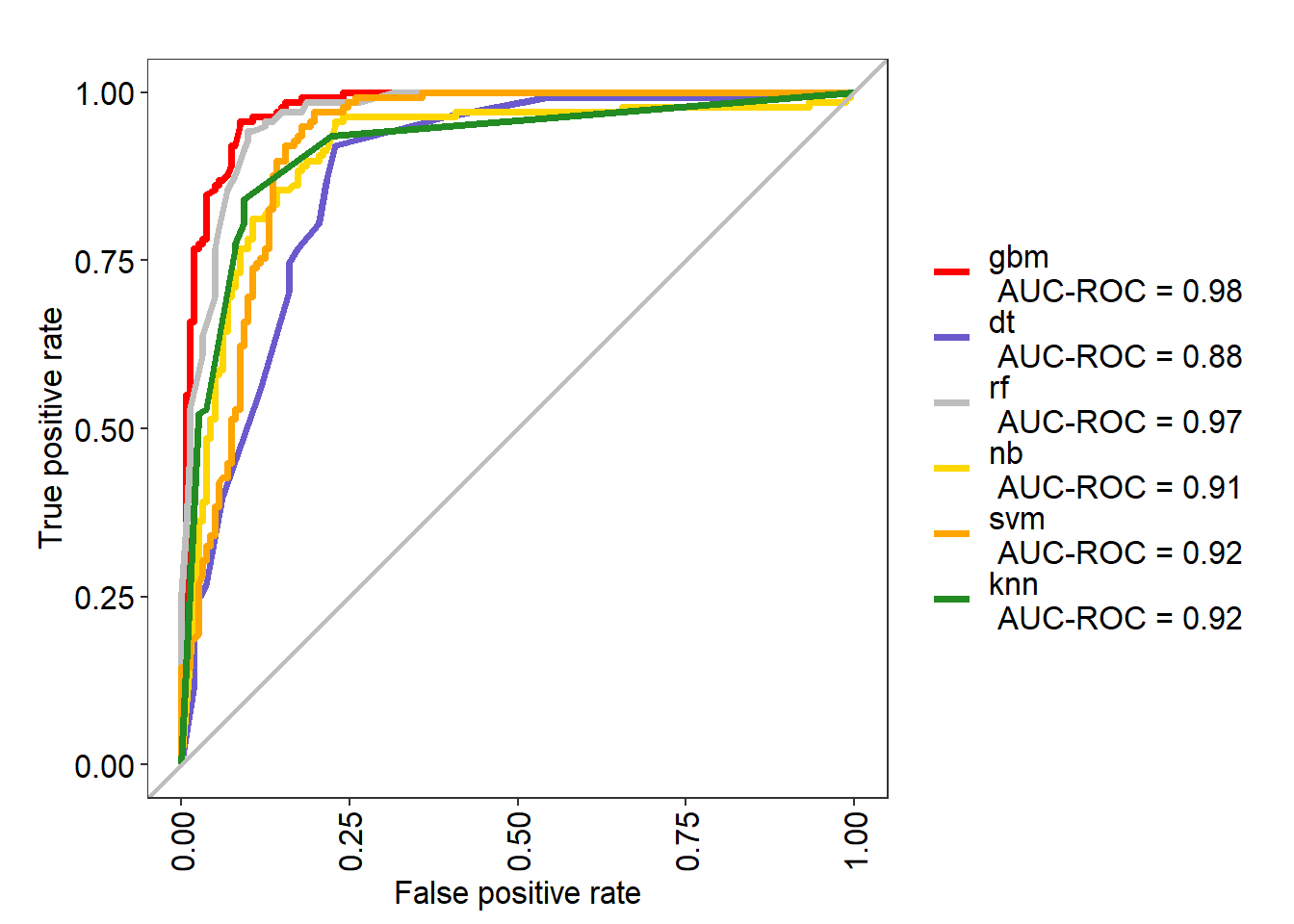
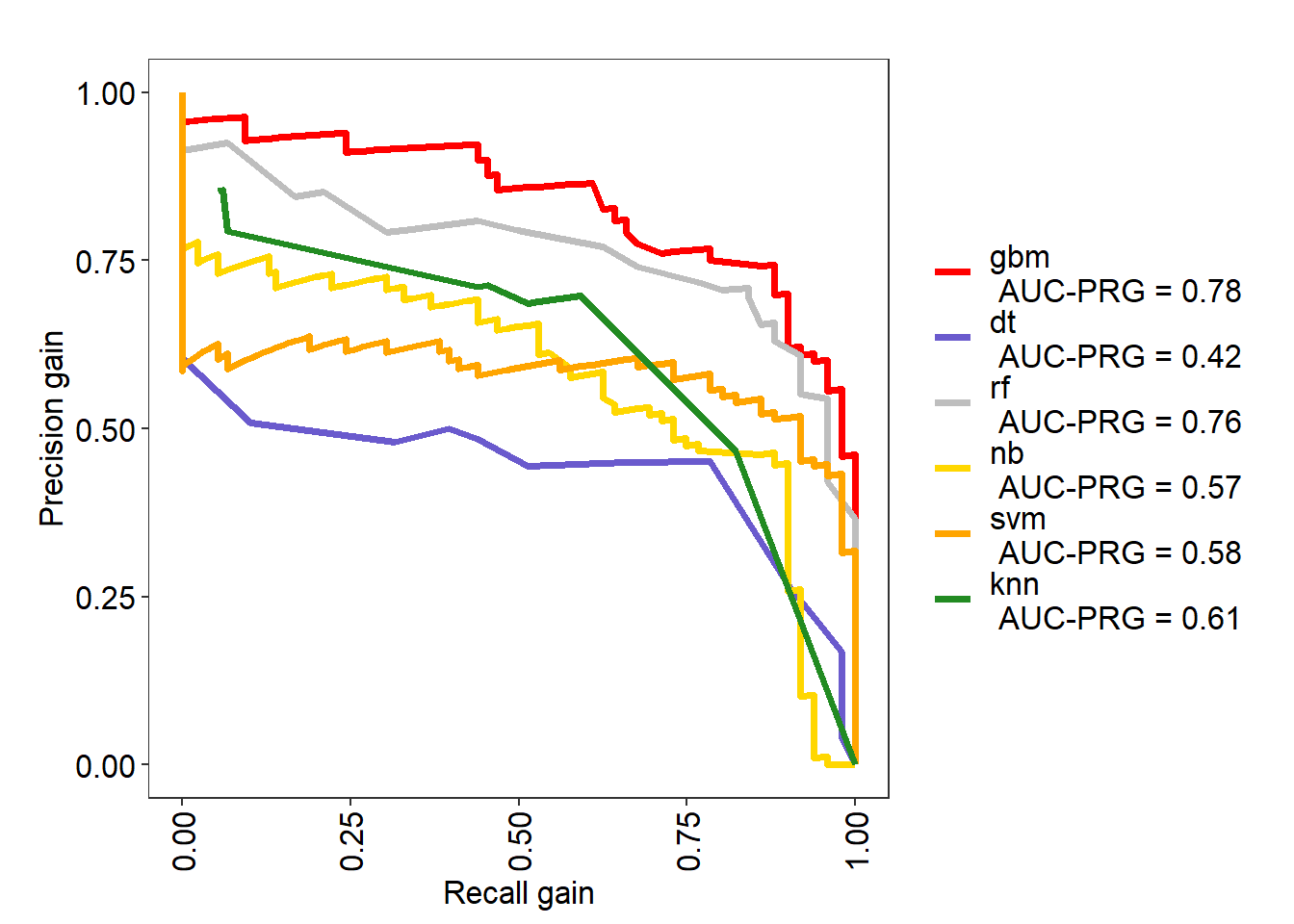
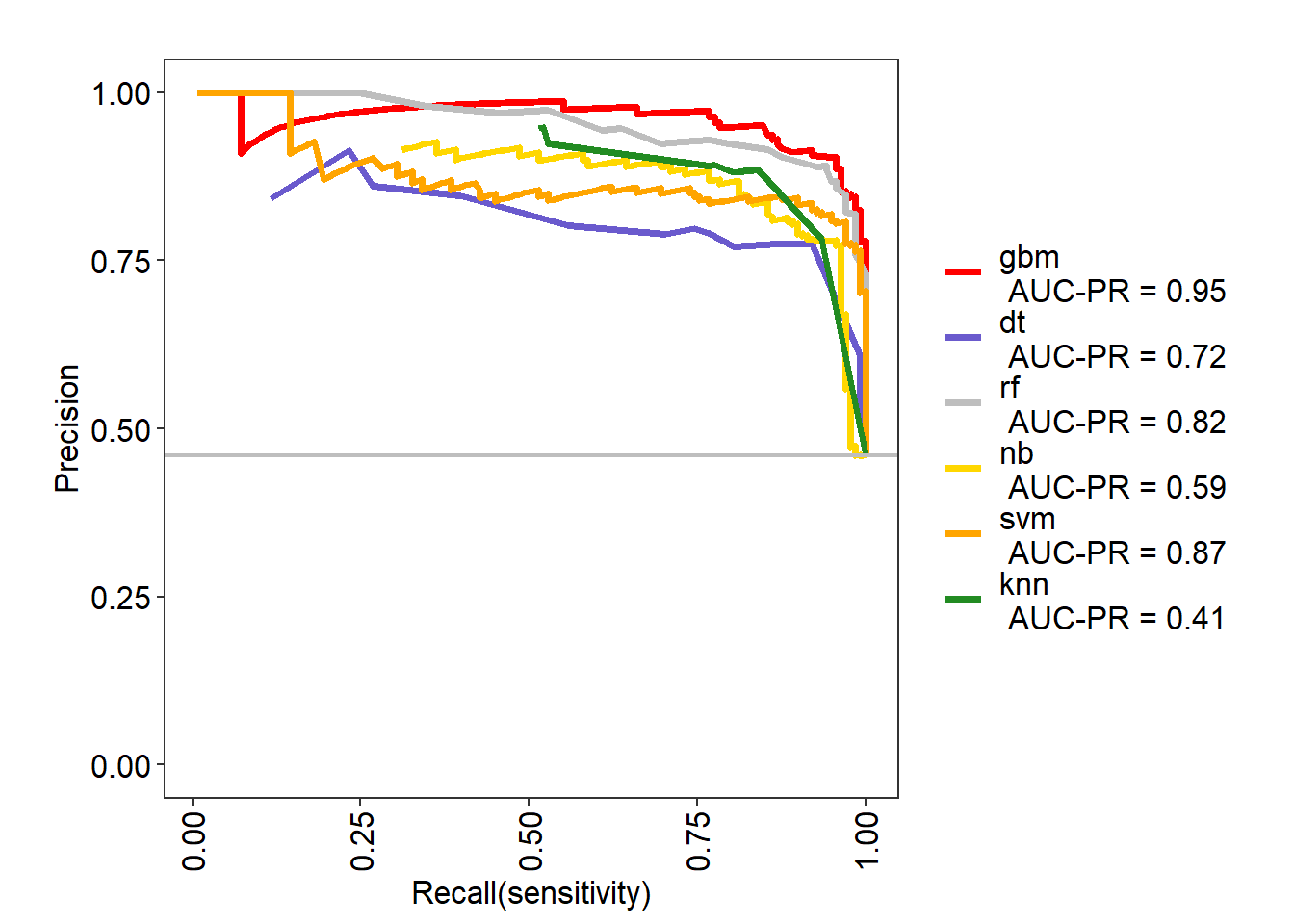
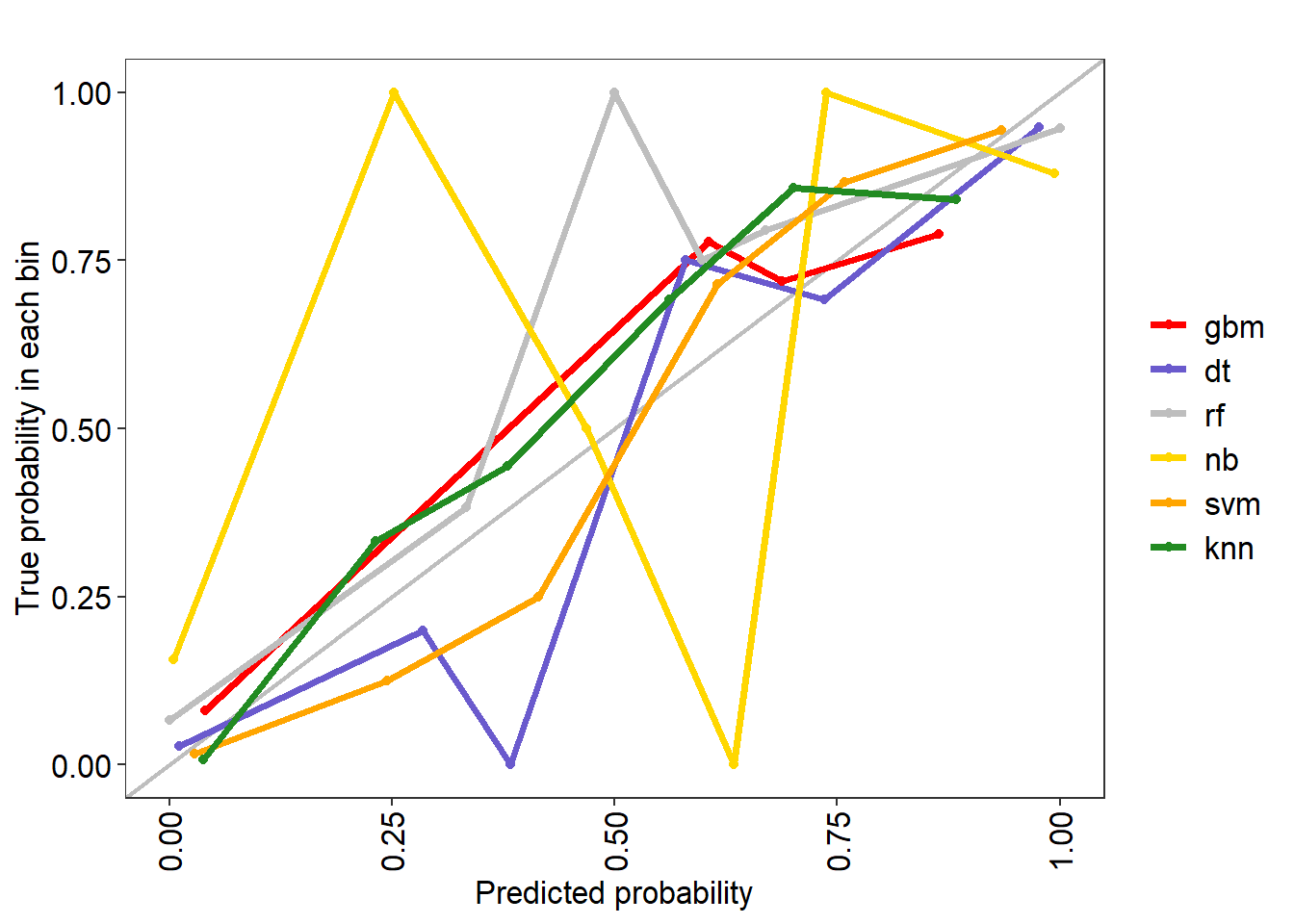
## knn 0.6283784 0.6952596 0.7359307 0.7370683 0.7603018 0.8654709 0



* **AUC-ROC** curve: This is a performance measurement for classification problem at various thresholds settings. ROC is a probability curve and AUC represents degree or measure of separability. It tells how much model is capable of distinguishing between classes. The **Higher the AUC, better the model** is at making predictions.

It should be noted that we fitted our models using “accuracy” to aide with finding the best tuning parameters. However, it is good to measure models with multiple metrics and AUC-ROC is one of the additional metrics to consider when making a decision.

The last graph indicates all the models look very good in terms of AUC-ROC, but gradient boosting, random forest are standouts. This means the gradient boosting machines resulted in a better fit than all other models, judging by the AUC-ROC metric. The Naive Bayes, Support Vector Machines, and K Nearest Neighbors were not far behind GBM, but the decision tree was trailing a bit more.

The recall sensitivity is import here too, since that is our ability to accurately discern all the fatalities. The gradient boosting machines method proving to be a good model for true positives. Random forests and support vector machines do reasonably here, but naive Bayes, decision tree, and kNN trail significantly on this metric. 

### Model Prediction Results

Definition of the Terms related to measuring classification models:

* Positive (P) : Observation is positive (for example: is an apple).
* Negative (N) : Observation is not positive (for example: is not an apple).
* True Positive (TP) : Observation is positive, and is predicted to be positive.
* False Negative (FN) : Observation is positive, but is predicted negative.
* True Negative (TN) : Observation is negative, and is predicted to be negative.
* False Positive (FP) : Observation is negative, but is predicted positive.

**Accuracy**: \* Classification Rate or Accuracy is given by the relation: (TP + TN) / (TP + TN + FP + FN) \* It assumes equal costs for both kinds of errors. A 99% accuracy can be excellent, good, mediocre, poor or terrible depending upon the problem.

**Recall**: Recall can be defined as the ratio of the total number of correctly classified positive examples divide to the total number of positive examples. High Recall indicates the class is correctly recognized (a small number of FN). Recall is given by the relation: TP / (TP + FN)

**AUC-ROC** curve: This is a performance measurement for classification problem at various thresholds settings. ROC is a probability curve and AUC represents degree or measure of separability. It tells how much model is capable of distinguishing between classes. **Higher the AUC, better the model** is at binary predictions.

Summary the test results for each of the following classifier models. The results vary slightly from run to run due to random sampling.

Before PCA:

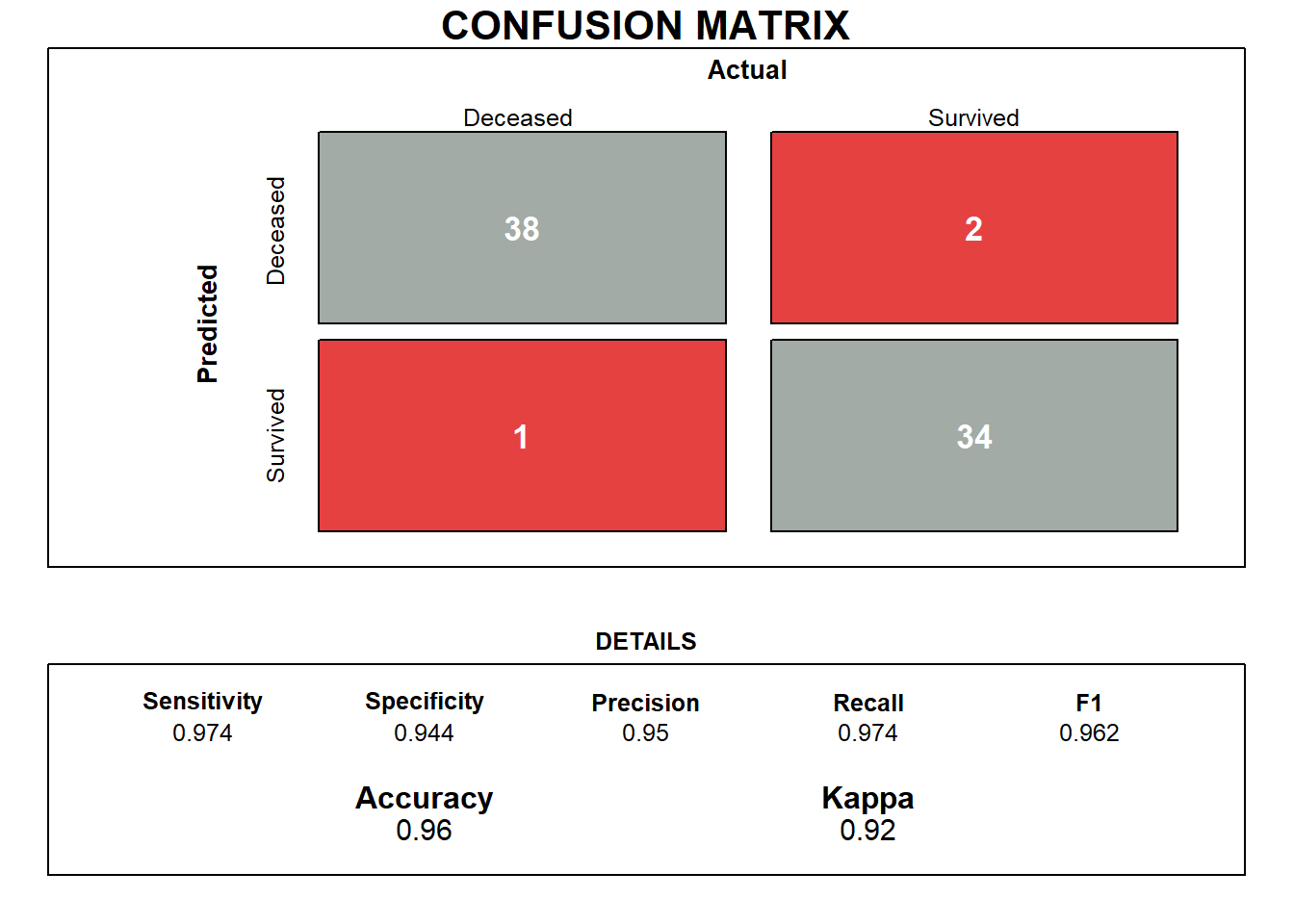
* Decision Tree - Accuracy : 0.90
* Random Forest - Accuracy : 0.94
* Naive Bayes - Accuracy : 0.90
* K Nearest Neighbors - Accuracy : 0.92
* Support Vector Machines (radial)- Accuracy : 0.89
* Classification Based On Association Rules (CBA) - Accuracy : .88 \*\*this was run with no PCA involved

After PCA:

* Decision Tree - Accuracy : 0.94
* Random Forest - Accuracy : 0.92
* Naive Bayes - Accuracy : 0.88
* K Nearest Neighbors - Accuracy : 0.96
* Support Vector Machines (radial)- Accuracy : 0.88
* Gradient Boosting Machines - Accuracy : .96 \*\*this was added after PCA, so there is no before PCA run

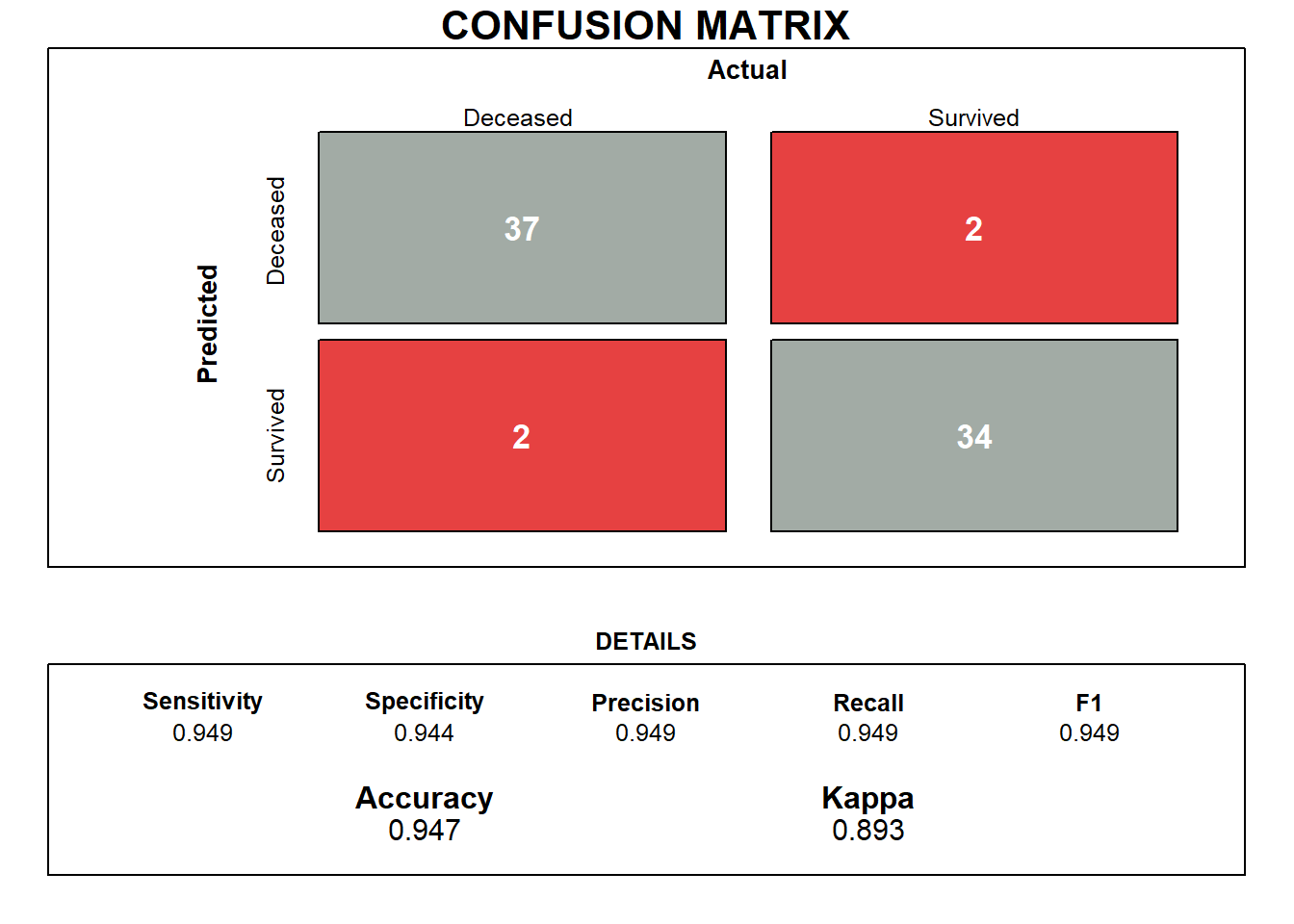
#### Gradient Boosting Machines

* Gradient Boosting Machines - Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. This model type was added to the analysis to challenge the accuracy of random forest. The results are 96% accuracy, and recall/sensitivity is 97%.



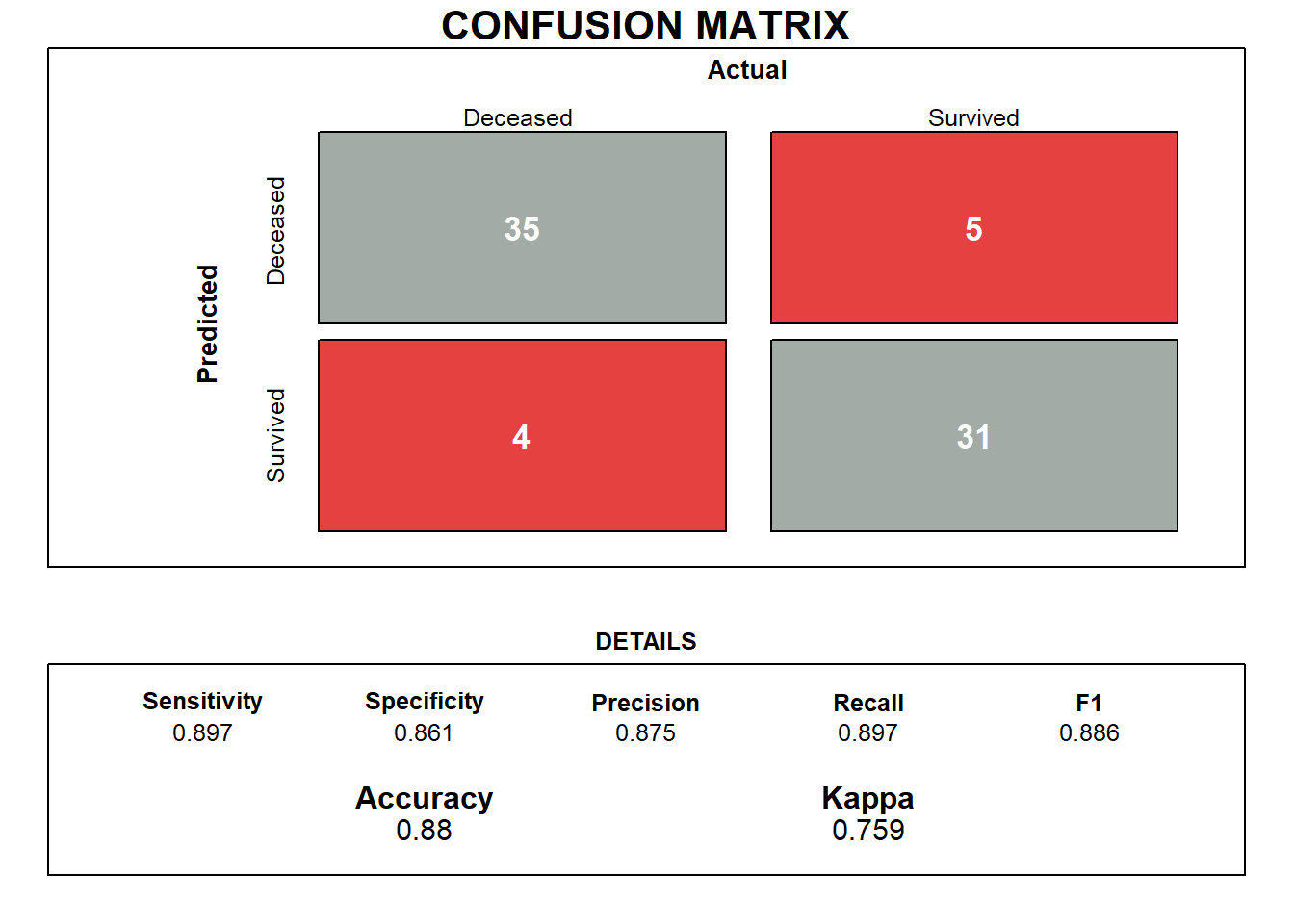
#### Decision Tree

* Decision Tree - A tree in which each internal (non-leaf) node is labeled with an input feature. The arcs coming from a node labeled with an input feature are labeled with each of the possible values of the target feature or the arc leads to a subordinate decision node on a different input feature. Decision tree tested at 94% accuracy, 95% recall/sensitivity, which is a solid test performance.



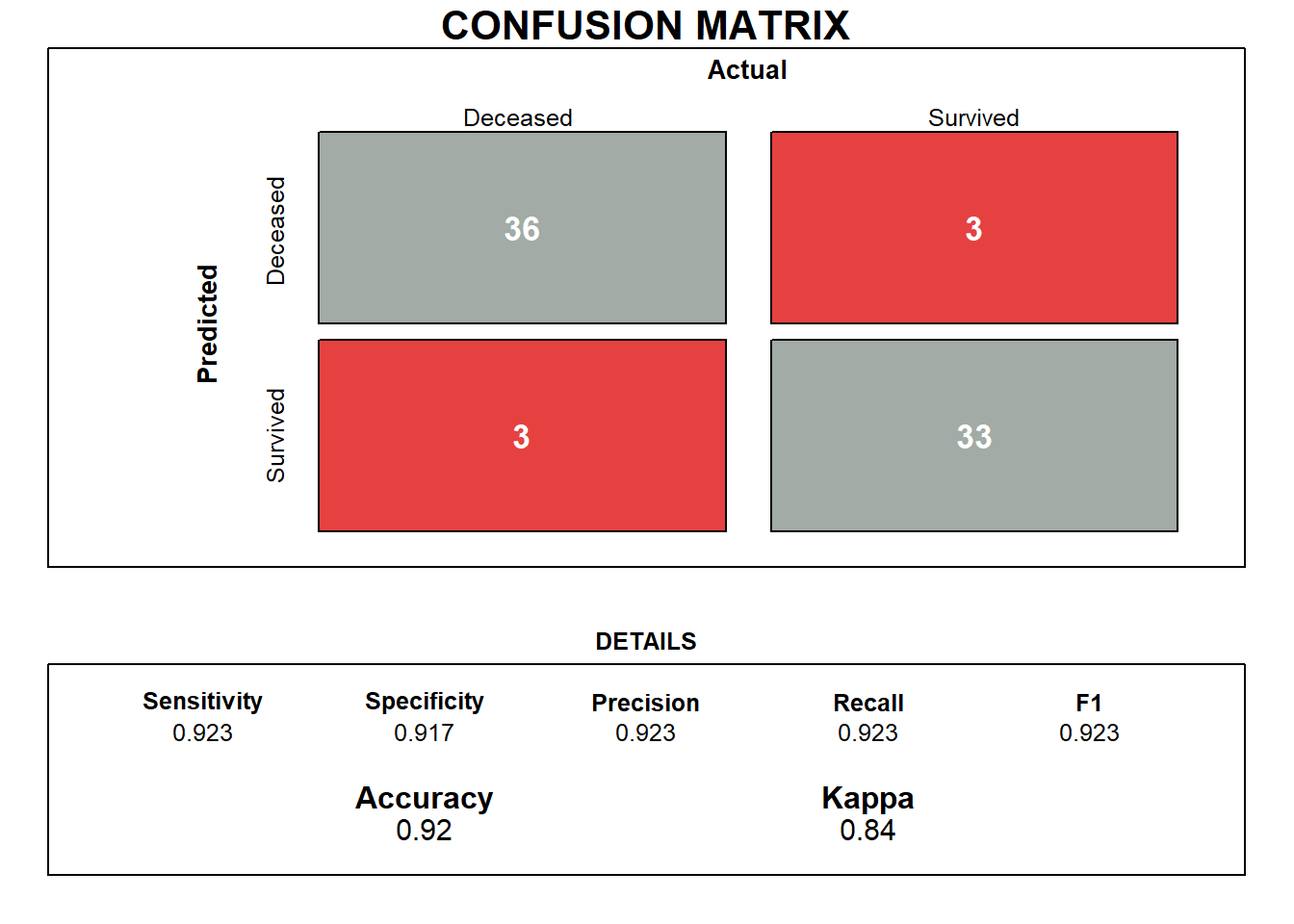
#### SVM

* Support Vector Machines (radial) - A discriminative classifier formally defined by a separating hyperplane. The algorithm creates a line or a hyperplane, which separates the data into classes. Support vector machines turned in an accuracy of 88% and 90% recall/sensitivity here.



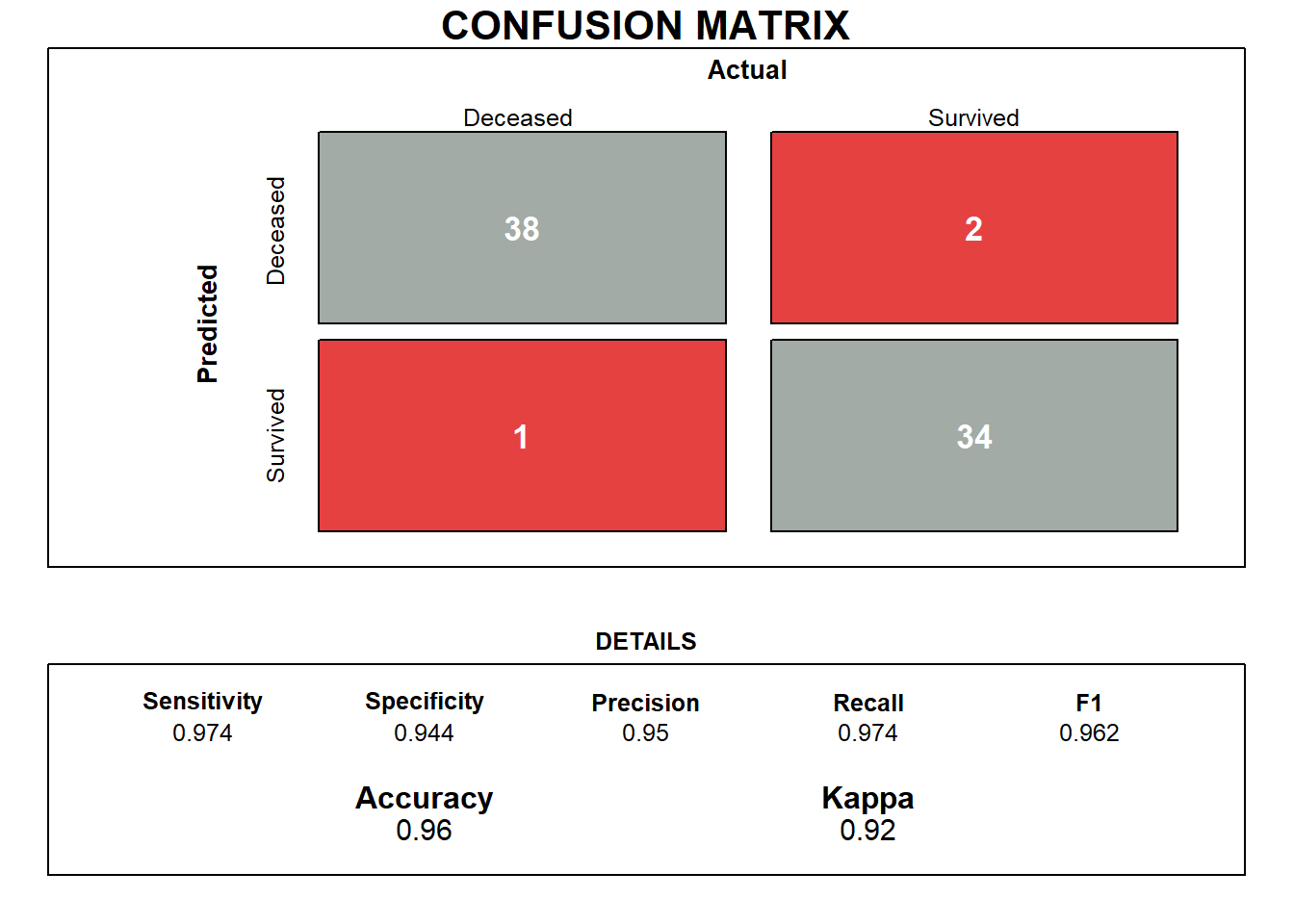
#### Random Forest

* Random Forest - An ensemble classifier that consists of many decision trees and outputs the class that is the mode of the classes output by individual trees. Our random forest testing produced accuracy of 92% and 92% recall/sensitivity. In comparison gradient boosting machines has done better in the testing compared to random forest.



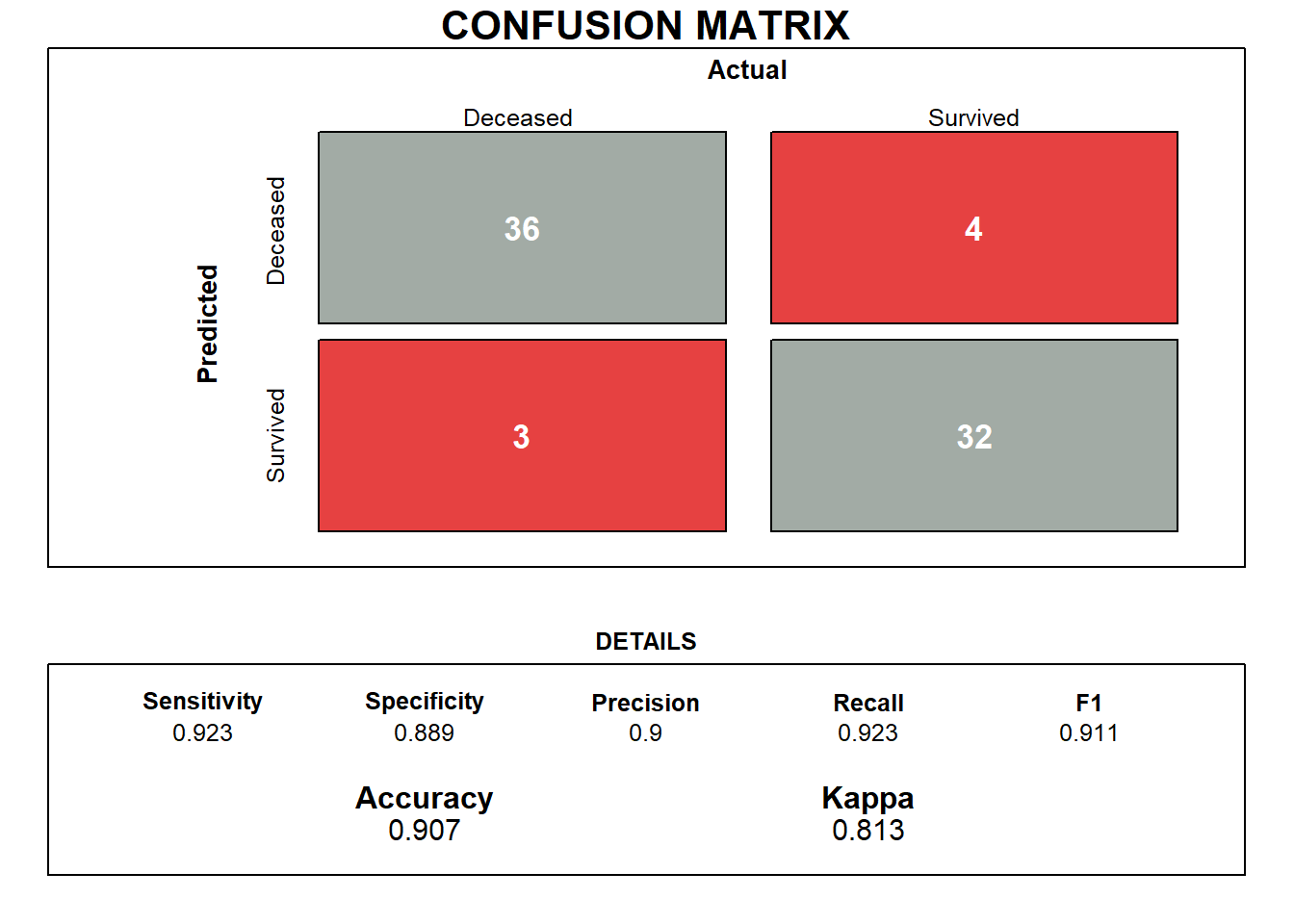
#### K Nearest Neighbors

* K Nearest Neighbors - KNN has been used in statistical estimation and pattern recognition already in the beginning of 1970’s as a non-parametric technique. It estimates how likely a data point is to be a member of one group or the other depending on what group in which the data points nearest to it. The k-nearest-neighbor is an example of a “lazy learner” algorithm, meaning that it does not build a model using the training set until a query of the data set is performed. The kNN results were very good with 96% accuracy and a high recall of 97%. It is a good candidate model, but it had 2 misclassified fatalities.



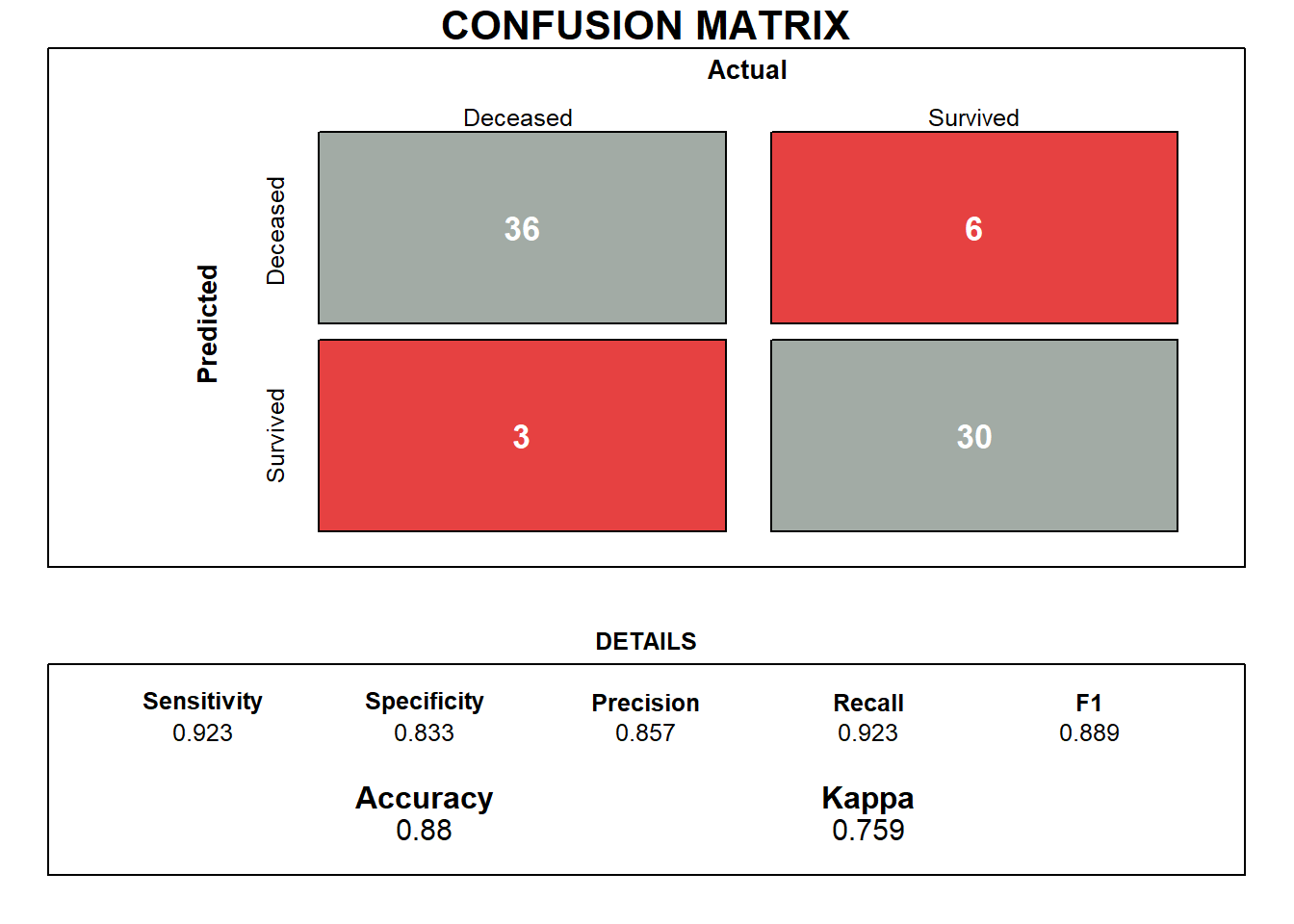
#### Naive Bayes

* Naive Bayes - An algorithm that uses Bayes’ theorem to classify objects. Naive Bayes classifiers assume strong, or naive, independence between attributes of data points. Naive Bayes has a reasonable accuracy at 90%, but it misclassified four fatalities.



#### CBA Model

* Classification Based on Association Rules - a classifier based on association rules mined for an input data set. CBA demonstrated good accuracy at 88%, and only misclassified six fatalities.



### Conclusions

After conducting the analysis on Covid-19 we discovered the number of Covid-19 cases can be misleading when used to compare countries. The ratio of death to cases, otherwise known as case fatality rate, is the best means to compare how countries are combating the virus. The analysis here fortunately shows the top eight countries, as measured by cumulative cases, show the key CFR measure decreasing. This is a good sign that the world’s challenges with Covid-19 are trending in the right direction.

However, the Twitter sentiment analysis indicate fears still exist. And these fears are certainly warranted given some citizens are highly susceptible to the virus based upon existing health issues. When a Covid-19 patient is admitted for treatment their blood is studied and documented according to medical standards. As part of the process, biomarkers are measured to help indicate phenomenon such as disease, infection, or environmental exposures.

The biomarkers of Covid-19 patients are important as this study concluded some of them can be used to predict mortality rates with high accuracy. During this study several machine learning classification models were developed and shown to predict mortality rates with accuracy rate of 90% and better. These results are promising, and if used as early detection mechanisms, for patients with Covid-19, they could greatly assist in reducing global case fatality rates (CFR) now or in the future.

### References

Yan, L., Zhang, H., Goncalves, J. et al. An interpretable mortality prediction model for COVID-19 patients. Nat Mach Intell 2, 283–288 (2020). <https://doi.org/10.1038/s42256-020-0180-7>

Related Links:

<https://www.nature.com/articles/s42256-020-0180-7.pdf> <https://github.com/HAIRLAB/Pre_Surv_COVID_19>

Drugs.com <https://www.drugs.com/medical-answers/covid-19-symptoms-progress-death-3536264/>

The Lancet <https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(20)30566-3/fulltext>

MedicalXpress.com <https://medicalxpress.com/news/2020-05-markers-covid-outcome.html>