**Covid-19 Survival**

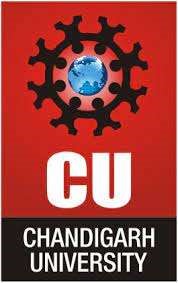
**Project Report**

**Balkar Singh (21MCI1007)**

**Sugandha Tanwar (21MCI1021)**

**Karan Bammi (21MCI1037)**

**Vishvajeet Singh (21MCI1023)**



# University Institute of Computing,

# Chandigarh University, Garhaun, Mohali

**Academic Year: 2021-2023**

**Abstract**

Nexoid1, a software company located in London, has created a large global medical dataset focusing on COVID-19 infection. This dataset consists of rich geographical, behaviour, segmentation, health conditions, and risk values to calculate the covid19 survival calculation. The calculation is totally based on two main factors risk infection and risk mortality.

1. **Introduction**

COVID-19 research dataset is created to predict COVID-19 risk infection and risk mortality rate, data was gathered by a software company located in London, has created a large global medical dataset focusing on COVID-19 infection. This dataset consists of 1023426 rows and 60 columns having geographical region, health diseases, factors used to reduce the spread of COVID-19. The initial World Health Organisation estimates put the chances of dying from COVID-19 between 4% and 7%. We now know that figure is significantly lower with current estimates putting it between 0.3% and 0.6%. The controversy around this figure comes from not knowing how many people have been infected. With some reports saying as many as 80% of infected people are asymptomatic. Either way vaccines are just artificial ways of creating antibodies. In the event that we are wrong and a vaccine is discovered the same herd immunity figure would apply, just it would be reached artificially. Regardless of how we get there, the virus will not stop until 60% to 70% of the population has antibodies.

1. **Technology Used**

***Jupyter Notebook (Google Colab)*** – Jupyter Notebook is an open-source web application that allows to create and share documents that contain live code, equations, visualizations and narrative text.

***Pandas***– Pandas is a fast, powerful, flexible and easy to use opensource data analysis and manipulation tool, built on top of the python programming language.

***NumPy*** – NumPy is a library for the python programming language, adding support for large multi-dimensional arrays and matrices, along with a large collection of high- level mathematical functions to operate on these arrays.

***Matplotlib*** – Matplotlib is a plotting library which we have used in our project for data visualisation.

***Seaborn*** – Seaborn is a python data visualisation library based on Matplotlib.

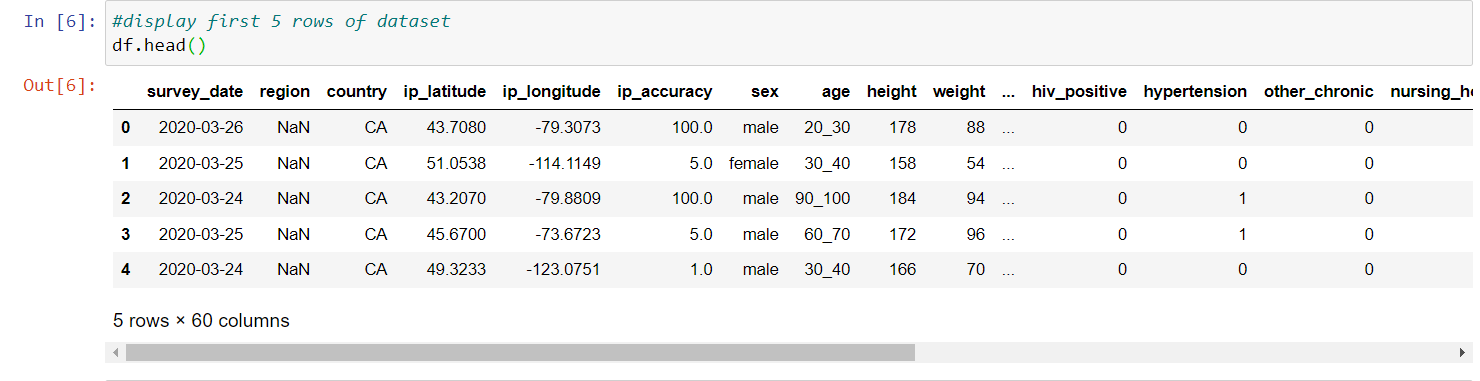
***Scipy*** – Scipy is a free and open-source python library used for scientific computing. It contains modules for optimisation, linear algebra, integration, interpolation etc.

1. **Methodology**

**Dataset description**

The dataset consists of 1023426 rows and 60 columns and the following columns are:

* Survey\_date: Recorded date of the data
* Region: Region
* Country: Country
* Ip\_latitude: Latitude coordinate of the location
* Ip\_longitude: Longitude coordinate of the location
* Ip\_accuracy: Accuracy of the location in meters
* Sex: Male, Female or other
* Age: Age quantile
* Height: Height of the person in cm
* Weight: Weight of the person in cm
* Bmi: Body Mass Index
* Blood\_type: Type of the person’s blood
* Insurance: If the person has insurance or not?
* Income: Type of income. For example, low, medium, high, or gov
* Race: Race of the person
* Immigrant: If the person is immigrant or not?
* Smoke: Information on how often the person smokes
* Contacts\_count: Number of people the person has contacted
* House\_count: Number of peoples living in a person’s house
* Public\_transport\_count: Number of people contacted by the person during public transportation
* Working: Status of the person’s work
* Worried: On a scale of 1 to 5, how much worried is the patient?
* Covid19\_positive: 0 and 1 stating existence0 and 1 stating existence
* Covid19\_symptoms: 0 and 1 stating existence
* Covid19\_contact: 0 and 1 stating existence
* Asthma: 0 and 1 stating existence
* Kidney\_disease: 0 and 1 stating existence
* Liver\_disease: 0 and 1 stating existence
* Compromised\_immune: 0 and 1 stating existence
* Heart\_disease: 0 and 1 stating existence
* Lung\_disease: 0 and 1 stating existence
* Diabetes: 0 and 1 stating existence
* Hiv\_positive: 0 and 1 stating existence
* Hypertension: 0 and 1 stating existence
* Other\_chronic: 0 and 1 stating existence
* Nursing\_home: 0 and 1 stating existence
* Health\_worker: 0 and 1 stating existence
* Risk\_infection: Risk of the person to get infected
* Risk\_mortality: Risk of the person to die



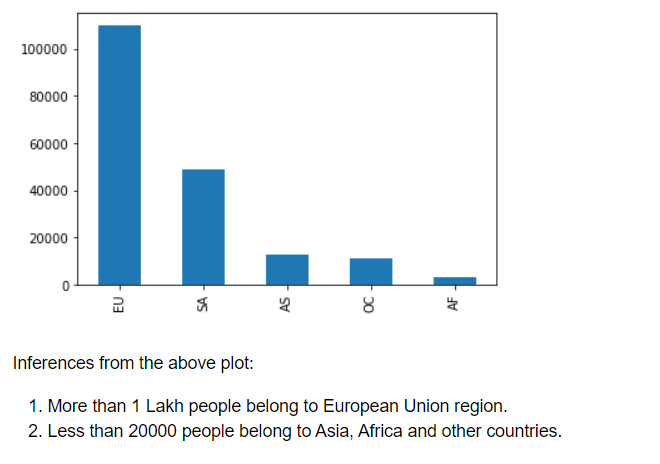
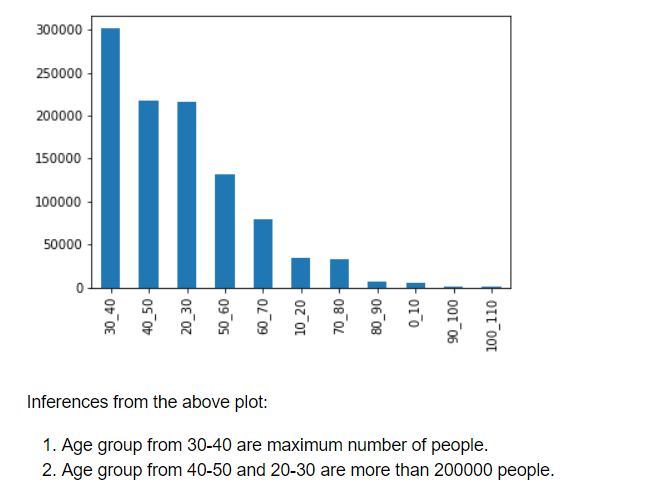
**Exploratory Data Analysis**

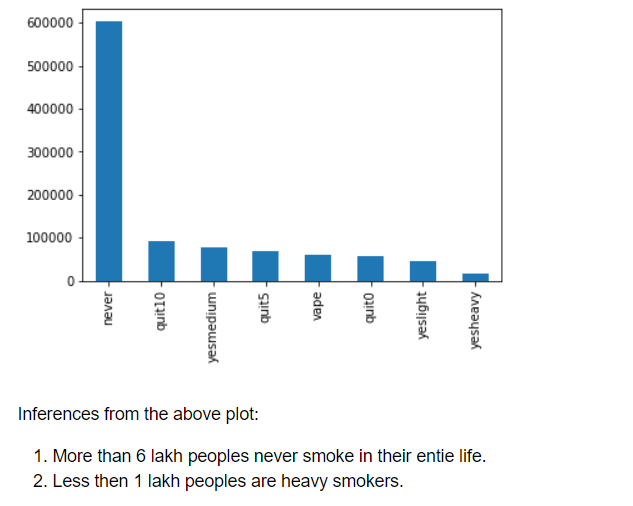
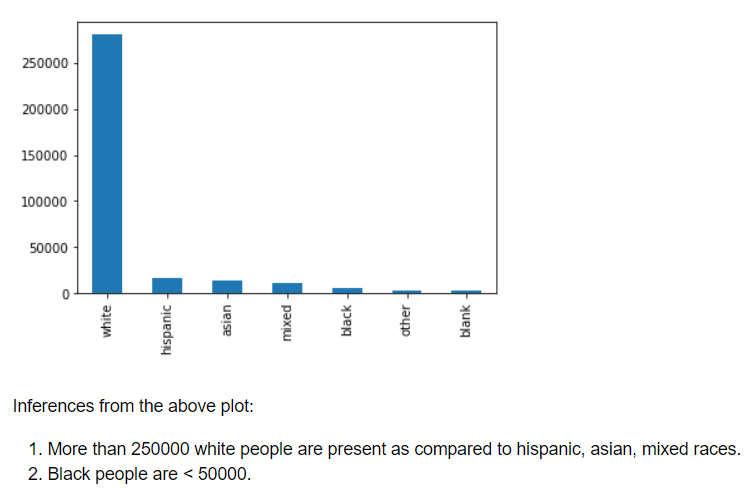
Exploratory data analysis is an approach to analysing the datasets to summarise their main characteristics, often with visual methods.

***Numerical columns:*** ip\_latitude, ip\_longitude, ip\_accuracy, height, weight, bmi, alcohol, cannabis, amphetamines, cocaine, lsd, mdma, contacts\_count, house\_count, public\_transport\_count, opinion\_infection, opinion\_mortality, risk\_infection, risk\_mortality.

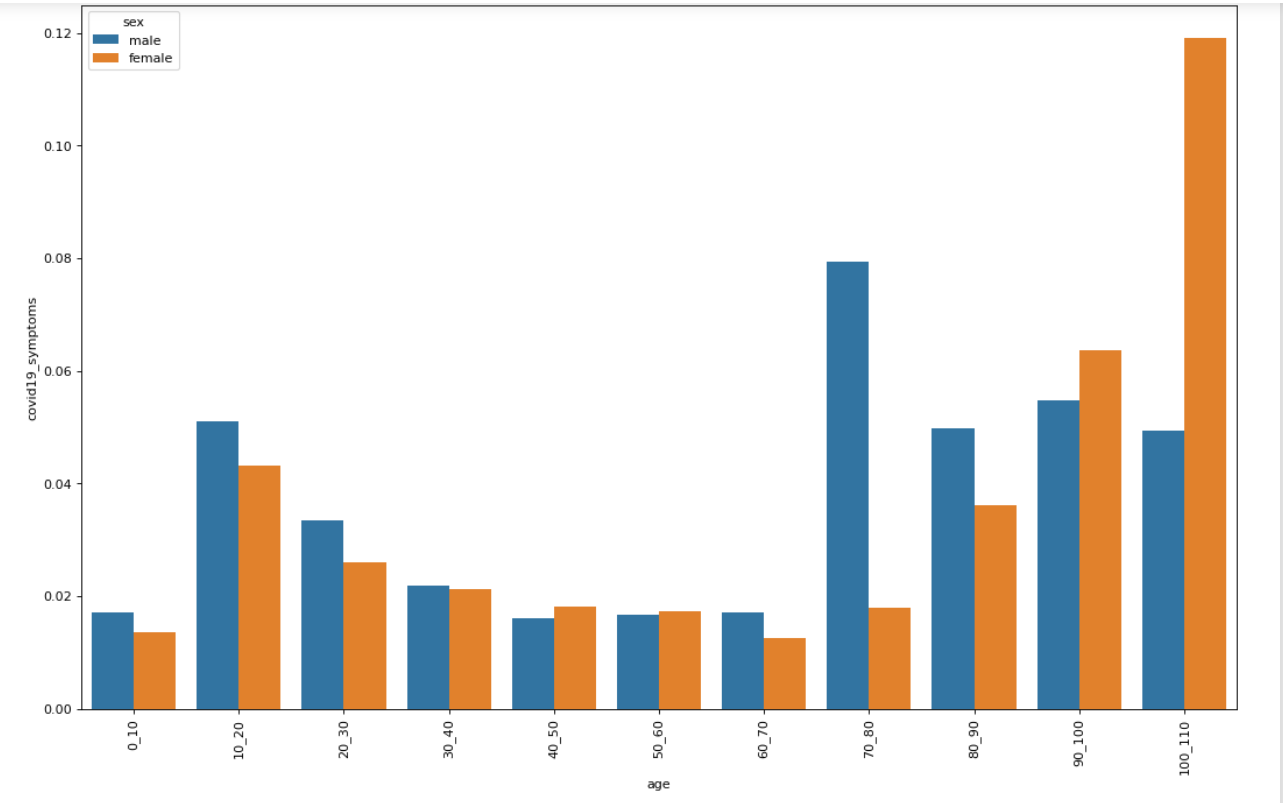
***Categorical columns:*** immigrant, heart\_disease, rate\_reducing\_risk\_house\_washing\_hands, rate\_government\_spend, age, rate\_reducing\_risk\_single\_sanitizer, covid19\_contact, rate\_reducing\_risk\_house, rate\_reducing\_mask, covid19\_positive, nursing\_home, asthma, region, income, other\_chronic, hypertension, worried, race, rate\_reducing\_risk\_single\_washing\_hands, diabetes, country, health\_worker, working, smoking, survey\_date, covid19\_symptoms, rate\_reducing\_risk\_single, kidney\_disease, rate\_reducing\_mask\_type, rate\_government\_control, lung\_disease, rate\_reducing\_risk\_house\_social\_distancing, blood\_type, compromised\_immune, liver\_disease, prescription\_medication, hiv\_positive, rate\_reducing\_risk\_single\_social\_distancing, sex, insurance, rate\_government\_action

***Univariate Analysis***



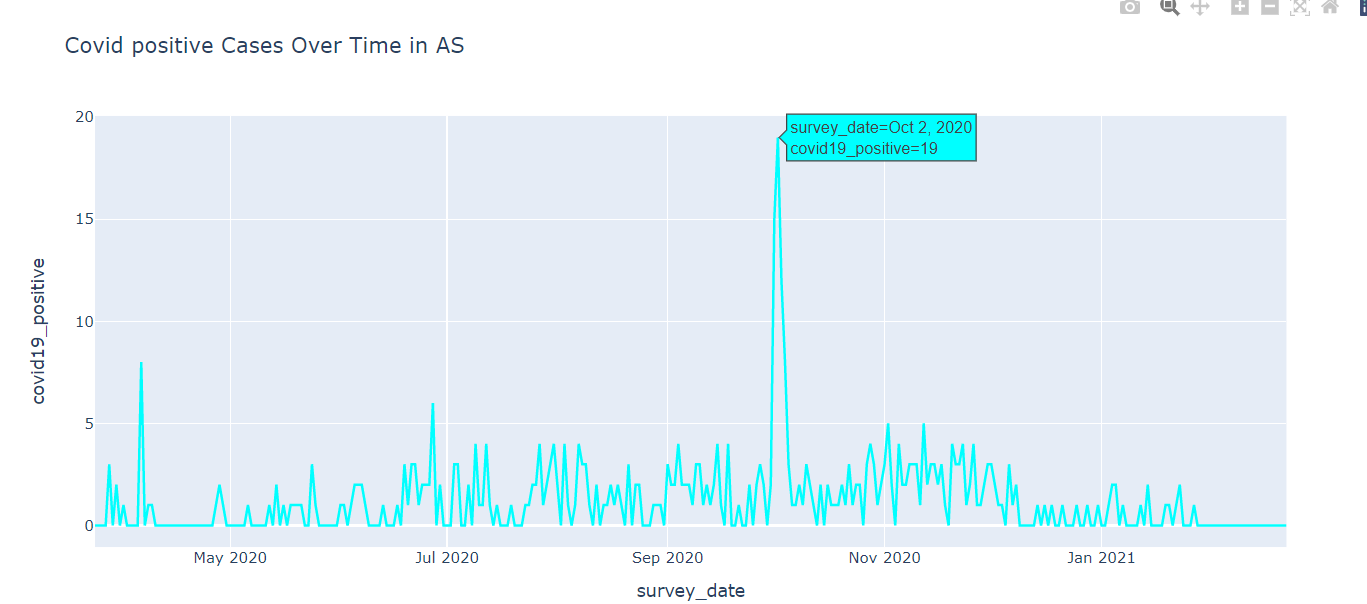
*** ***

***Bivariate Analysis***



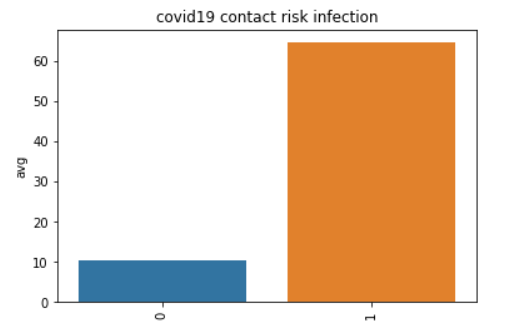
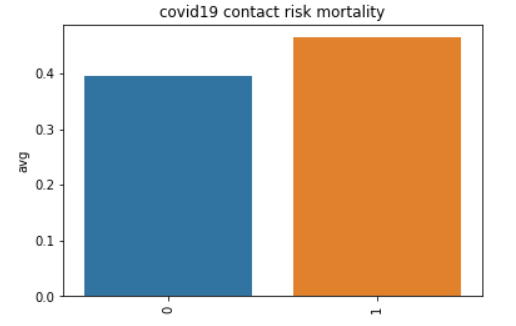
Inferences from the above plot:

Covid symptoms gradually increases after the age of 70 and more risk to females.



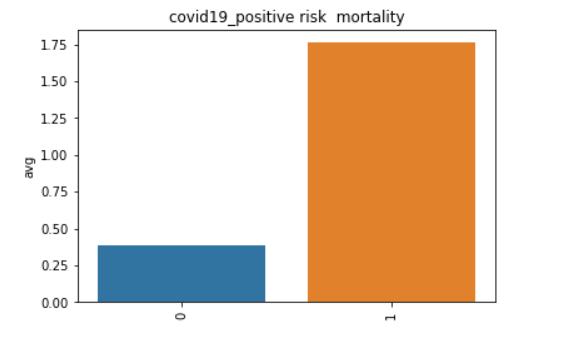
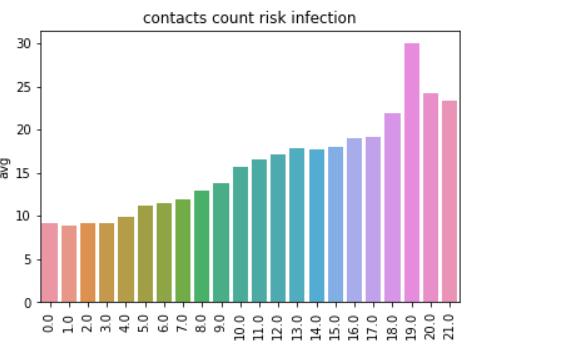
Inferences from the above plot:

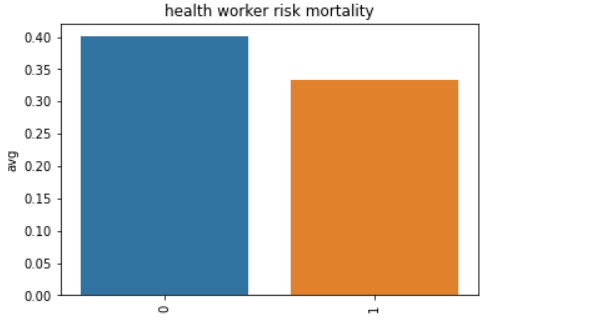
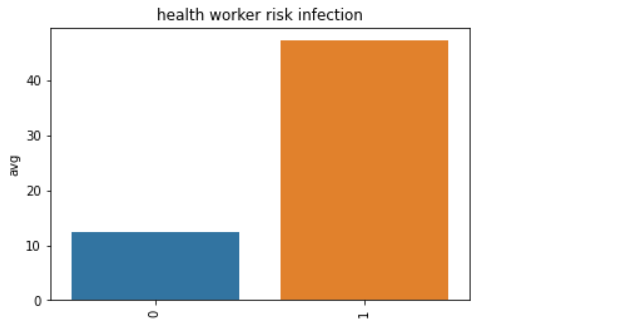
Covid19 positive rate with respect to survey date in Asia.

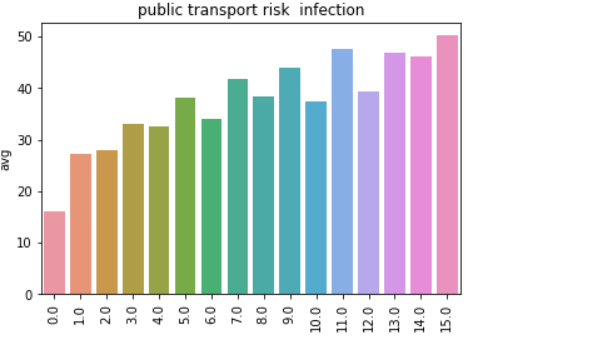
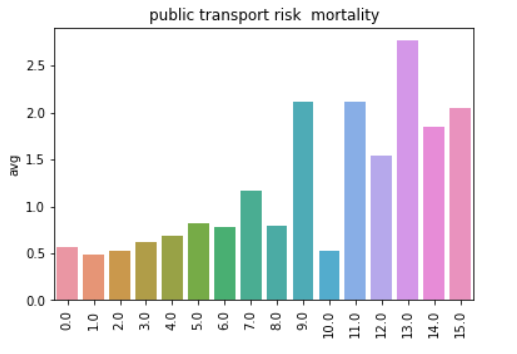


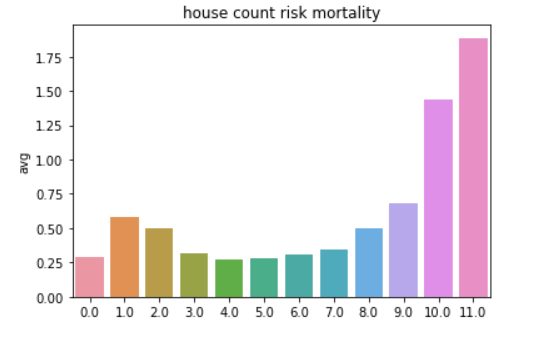
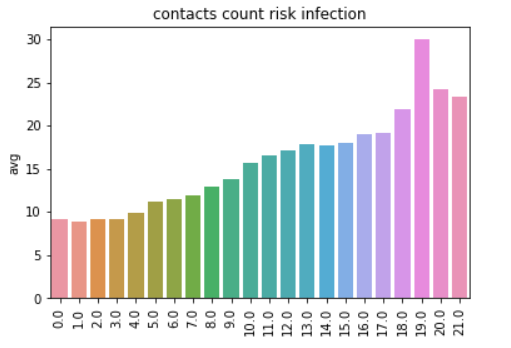
Inferences from the above plot:

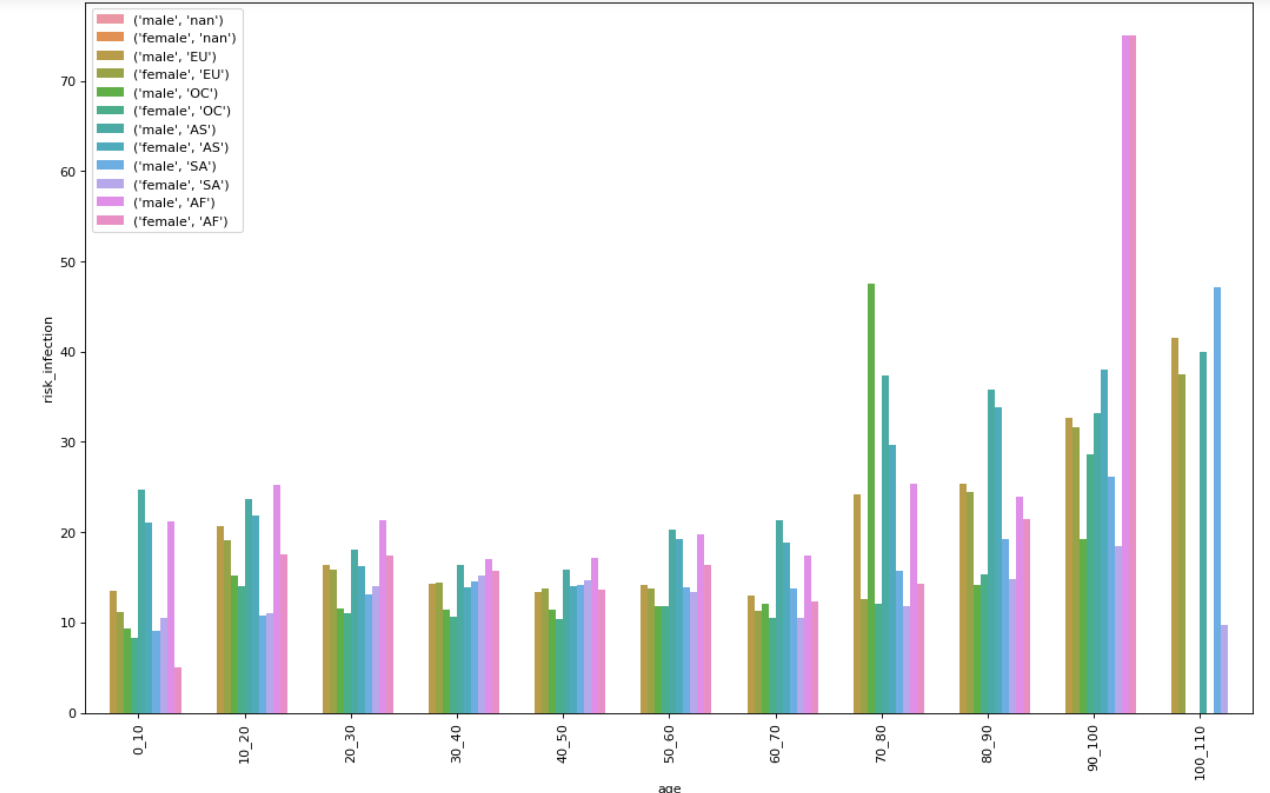
Covid19 contact risk infection has more than 60.



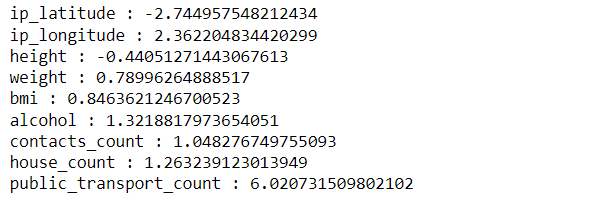
Risk infection with age, sex and region

**Data cleaning**

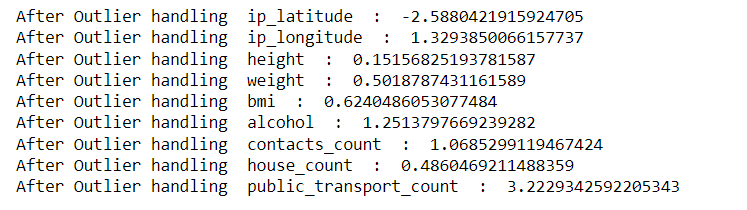
In data cleaning part, first check the unique column, if find any then drop that column as this column is not significant to our data. After that check the null values in dataset. There are 30 columns that having null values.

* Start by filling missing values of the ***country*** column, we are having missing values in region column too, but in given data we use latitude and longitude to extract the country. BY install geopy library of python, we will provide function of this library with latitude and longitude values and in turn this function will provide us with address of that IP and then we can extract country name from it, geopy is a Python client for several popular geocoding web services,geopy makes it easy for Python developers to locate the coordinates of addresses, cities, countries, and landmarks across the globe using third-party geocoders and other data sources.
* In ***region*** column which are more than 6lakh missing values, to get the region according to country, after analysing the data we got to know only North America is not there in region column other missing values which are not North America, that we can extract from other columns because they are mapped to at least one country.
* In ***age*** column, there are only 5 values missing in data of 10lakh+ we will use backward fill to fill these rows.
* In ***height, weight and bmi*** column there are only 4 missing values that filled them by median of these columns.
* In ***smoking,*** 1857 missing values, that are supposed to be non-smokers, so we fill these Nan values by 'never' using fillna method.
* In ***house\_count, contacts\_count, risk\_infection, risk\_mortality and rate\_reducing\_mask*** are filled with median values.
* In ***working*** column, there are more than 4000 values missing so we drop these rows.

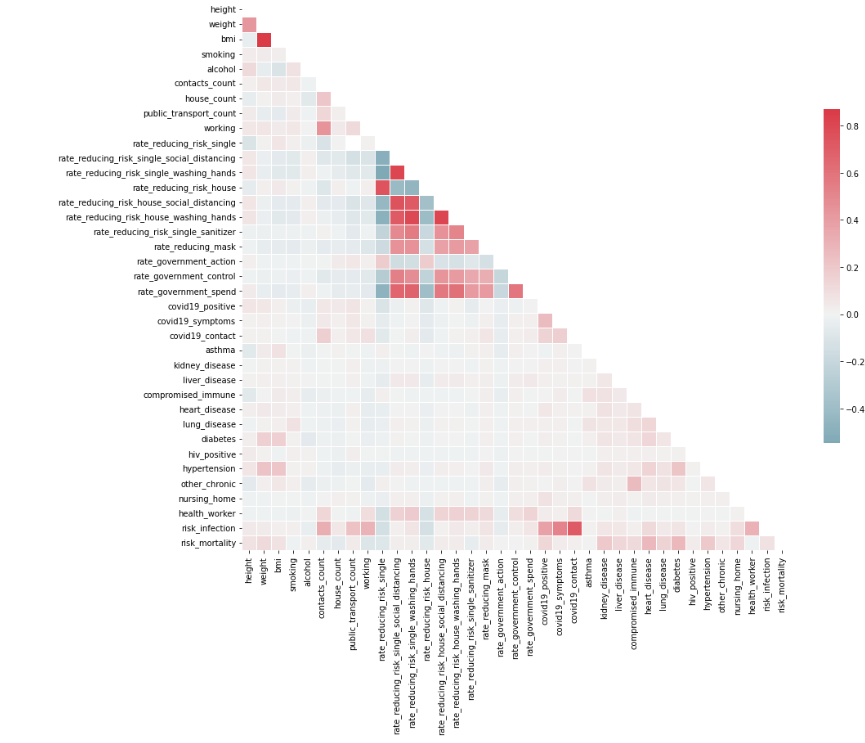
After cleaning the data drop the columns that are not having any significance.Check the ***skewness*** of continuous data, skewness tells us about data if it is in negative it means data if left skewed and if not, then our data is right skewed, mainly our data is right skewed.



**Handling outliers** using z-score and quantile capping method.



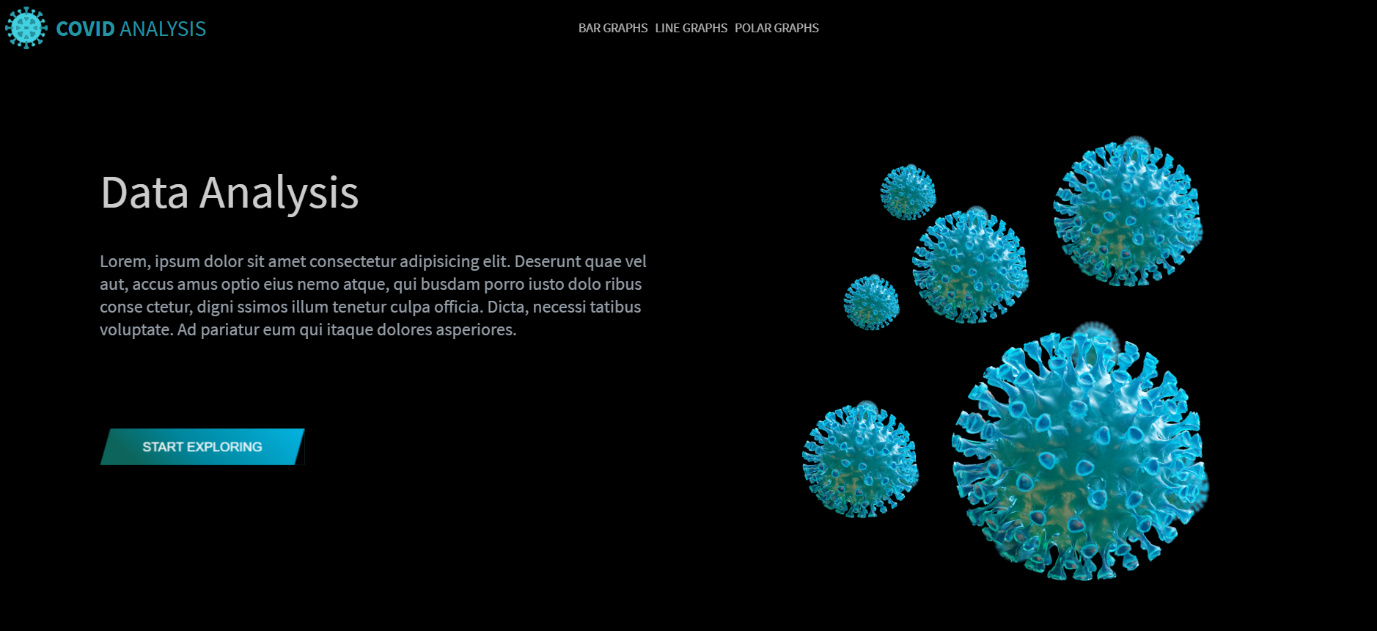
After handling outliers, find the correlation between different columns to know highly corelated data. As shown in fig, bmi is highly corelated with weight, risk\_infection is highly corelated with covid19\_positive, covid19\_symptoms, covid19\_contact.



1. **Implementation**

The front end of this project is done in React.js which is essentially JavaScript library to create single page web applications. A React.js project consists of multiple components that work together to create a complex User Interface. Instead of HTML, we have used Styled Components, which basically wraps the HTML tags in JavaScript constants and within these constants we can use CSS properties to style them. Instead of using multiple div’s in HTML, using styled components we can make our code more expressive, readable and elegant. The charts are displayed using a popular charts.js javascript library, which consists of various beautiful charts ready to be implemented. To make the UI more interactive we have used Animate on Scroll library to animate the components on scroll which looks more professional.

Front page:



Second page:

