

# **Project Report**

**Transforming COVID-19 testing data into actionable evidence for public health decision-making using epidemiological, spatiotemporal, and data-science methods**

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Project Description . . . . .	3
1.2	Objectives . . . . .	3
<b>2</b>	<b>Fetching and Preparing COVID-19 Testing Data</b>	<b>5</b>
2.1	Background . . . . .	5
2.2	Initial Process . . . . .	5
2.3	Our Approach . . . . .	6
<b>3</b>	<b>Cleaning and Exploring the Data Variables</b>	<b>18</b>
3.1	Looking at the variables . . . . .	18
3.2	Patient Category . . . . .	19
3.3	Clean test status positive . . . . .	21
3.4	Clean Age and Gender Variables . . . . .	27
3.5	Cleaning of the states and their districts: . . . . .	33
3.6	Cleaning the symptoms: . . . . .	39
3.7	Cleaning the Testing Kits Used . . . . .	44
<b>4</b>	<b>Organizing the Data for Analysis</b>	<b>49</b>
4.1	Convert the file format . . . . .	55
4.2	ICMR COVID-19 Downloaded Data Information . . . . .	58
<b>5</b>	<b>Results</b>	<b>59</b>
5.1	National-Level . . . . .	59
5.2	Symptoms among Positive Cases . . . . .	60
5.3	State Level . . . . .	62
5.4	By Age Group . . . . .	62
5.5	State-Wise COVID-19 Symptom Percentages by Wave in India . . . . .	105
5.6	Spatial Distribution . . . . .	123
<b>6</b>	<b>Conclusions</b>	<b>146</b>

# 1 Introduction

## 1.1 Project Description

In a low-middle-income country like India, public health managers at the central, state, district, and block-level are not only faced with vast challenges of managing the COVID-19 outbreak but also to be considerate of the precious resources and utilize them judiciously and efficiently. As policymakers and community leaders have worked to respond to the COVID-19 pandemic, it has become increasingly clear that statistics and data science can play a critical role in protecting public health and determining the best path forward. However, moving from theory to practice presents challenges for working with a patchwork of data from many different sources across public and private sectors. In addition, uncertainties at various levels such as the evolving nature and novelty of the disease, differences in the societal and community response to the pandemic, and disparities in the health system response contribute significantly to the challenge. Under these circumstances, only well-chosen data that undergoes thorough and meticulous analysis can guide the actions of the health system, which is possible by the field of data science. Furthermore, the data science approach to public health based on robust scientific methods may offer various solutions to the current challenges. These include innovative solutions such as enhancing data quality, integration and linkage of multiple data sources for a comprehensive understanding of disease patterns; engaging stakeholders at every level for a plausible explanation of the patterns; timely communication of epidemiological insights and their public health implications to the decision-makers. This study aims to demonstrate the applications of data science, epidemiology, and spatiotemporal methods in transforming routine COVID-19 testing data into actionable evidence for public health decision-making. The proposed study has many public health implications in strengthening disease surveillance, capacity building, and improving the overall resilience of the routine health information system.

## 1.2 Objectives

### 1.2.1 Primary Objectives

1. To develop a framework for generating evidence from data on COVID-19 testing to aid in public health decision-making using open-source tools.
2. To attempt data linkage, triangulation, and integration of various databases on epidemiological, socio-demographic, environmental, and genome sequencing data relating to COVID-19 in India.

### 1.2.2 Secondary Objectives

1. To identify, quantify data quality issues in COVID-19 testing data and address these issues using the data science approach and reproducible research.
2. To study the variations in symptomatology, cycle threshold (Ct) values, and test positivity and its associations with patient categories and dominant strains in space and time.
3. To explain and draw inferences about these variations using traditional epidemiology, infectious disease epidemiology, and spatio-temporal methods.
4. To communicate the findings of the study in the form of interactive and dynamic dashboards, which can potentially aid district-level program managers in public health decision-making in near-real-time.

The *Transforming COVID-19 testing data into actionable evidence for public health decision-making using epidemiological, spatiotemporal, and data-science methods* project aimed to address the challenges associated with handling large-scale pandemic-related data. While the project's objectives could not be fully realized due to unforeseen limitations, significant progress was made in several key areas. These efforts culminated in developing practical workflows for accessing, organizing, and converting data for analysis, showcasing the team's adaptability and problem-solving capabilities.

The following sections outline the methodologies and approaches employed in the project, detailing our work on data retrieval, combination, and optimization. These processes not only enhanced the efficiency of managing complex datasets but also contributed to the broader research community. The project's outcomes were well-received, earning recognition in the form of a poster presentation award at a prestigious conference.

## 2 Fetching and Preparing COVID-19 Testing Data

As part of the initial phase of the project, the first critical task was to fetch the raw COVID-19 testing data from the ICMR database. This data serves as the foundation for all further analyses, ensuring that reliable and up-to-date records are available for epidemiological modeling, spatiotemporal analysis, and data science techniques. The objective of this process was to download the COVID-19 testing data, store it in a local server, and prepare it for subsequent analysis.

The process of fetching the data was not straightforward due to several challenges, including the limitations on the number of records retrieved per request and the need for a continuous data-fetching mechanism. To overcome these obstacles, we developed a systematic approach using a combination of tools like POSTMAN and custom scripts in R.

The following sections detail the process of fetching the data, including the challenges faced and the methods used to automate and streamline this task.

### 2.1 Background

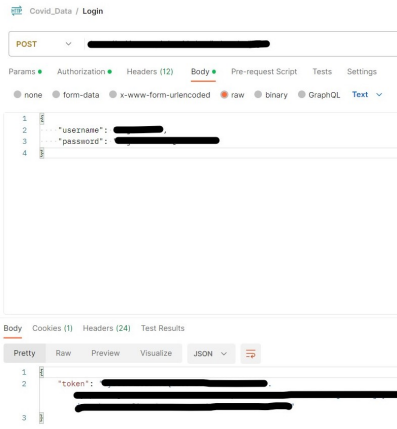
The data on COVID-19 Testing in India is provided to select researchers through a two-step authentication process by the ICMR. This involves generating an API Token using the credentials (username and password) provided by the BMI Division, ICMR. The access to the data is limited to 10,000 records at a time and if we want more records, we need to provide a separate API request by offsetting the initial 10,000 records. The API access for downloading the data was provided on 26th July 2022.

Objective: Download the COVID-19 testing data to the local server for further analysis

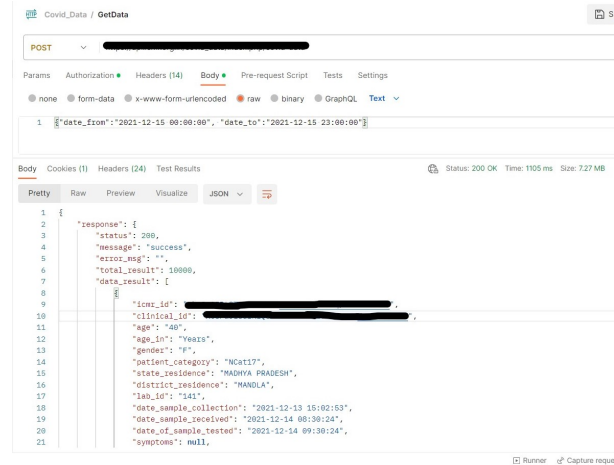
### 2.2 Initial Process

#### 2.2.1 Using POSTMAN

We have used the [POSTMAN](#) application for getting information about the number of records per day by giving input as credentials to get the token. The credentials are passed in the body of the POST request and the token is generated. This token is provided as the bearer token, and a second POST request is made along with the `date_from` and `date_to` arguments. The output is the first 10,000 records of the requested duration in `.json` format. Note that the size of the output (10,000 records) is close to 8MB.



(a) (A) Get Token



(b) (B) Get Data

Figure 2.1: Screenshots of the POSTMAN application

## 2.2.2 Challenges using POSTMAN

We faced a number of challenges when downloading the data using POSTMAN.

Some of them are:

1. The API Token needs to be generated and updated each time a POST request is made.
2. Each request provides a maximum of 10,000 records.
3. A fresh request needs to be made by updating the **offset** argument to get the next 10,000 records.
4. There was no way of knowing how many times we need to make this request as we do not have any idea how many records are there each day.
5. The output data was provided in a plain text `.json` format, which was bulky and needed to be converted to a more memory-efficient format.
6. All the above steps needed to be done manually.

## 2.3 Our Approach

To address the challenges faced with POSTMAN, we implemented a data science approach to streamline the download process and optimize the data format for analysis. We have implemented this approach in R, which is an open-source statistical programming language.

We have achieved this with the help of a few custom helper functions:

1. `get_token_fn`: Retrieves the authorization token
2. `get_data_fn`: Retrieves data for a specific day using the token
3. `check_data_fn`: Checks the status and total results of the fetched data
4. `update_offset_fn`: Updates the offset value based on the status code



```

)

# seq.Date(from = as.Date(x), to = Sys.Date(), by = 1)

body = paste(
  '{"date_from\":"',
  day,
  ' 00:00:00"',
  '\date_to\":"',
  day,
  ' 23:59:59"',
  '\offset\":"',
  offset_updated, '\"} ',
  sep = "")

res <- VERB("POST",
            url = "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
            body = body,
            add_headers(headers),
            config = httr::config(ssl_verifypeer = FALSE))

day <- day
json_data <- content(res, 'text')

json_data <- jsonify::from_json(json_data)
df <- json_data$response$data_result |> as_tibble()

df |>
write_csv(
  here("data", folder_path, paste0("df_", epitrix::clean_labels(day),
  "_", offset_zero, ".csv", sep = "")))
}

```

### 3. Verify the Data

```

# Function for verifying the data

check_data_fn <- function(){

```

```

StatusCode <- content(res, 'parsed') |>
pluck('response') |>
pluck('status')

TotalResults <- content(res, 'parsed') |>
pluck('response') |>
pluck('total_result')

return(
  cat(paste("Status Code: ", StatusCode, "\n
  Total Result: ", TotalResults, "\n ", "Day is: ", day, "\n "))
)
}

```

#### 4. Update the offset

```

# Function for updating the offset

update_offset_fn <- function(){
  if(StatusCode == 200) offset_zero <- offset_zero + 10000
  # StatusCode 200 means success
  if(StatusCode == 401) token <- get_token()
  # StatusCode 401 means Token expired

}

```

#### 5. Wrapper function to fetch the data by date

```

# Function for updating the offset

fetch_data_by_date_fn <- function(x, offset_zero = 0){
  options(scipen=999)

  folder_path <- paste(paste("df_", epitrix::clean_labels(x), sep = ""))
  fs::dir_create(here("data", folder_path))

  get_token_fn()
  get_data_fn()
}

```

```

check_data_fn()

while(StatusCode == 200 & TotalResults >= 10000){
  update_offset_fn()
  cat('This is offset number:', offset_zero , "\n")
  get_data_fn()
  check_data_fn()
}

}

```

## 6. Fetching All Records

In order to fetch all the records per day, we have faced the challenge and written a function to fetch the day-by dates. The other challenge was that the generated API token gets expired at a time and also we have written another function to check the status of the token and re-run the function to get the token. We have downloaded the data in the .rds format in the initial stage. The data is larger in size in .rds format and it was complex when taken for further analysis process. So we have decided to download the data in .csv format where the size is comparatively smaller than .rds format. We have downloaded data from March 2020 to April 2023.

```

# Script for Fetching ICMR COVID-19 DATA

# Load the Libraries
library(httr)
library(tidyverse)
library(here)
library(fs)
library(furrr)
library(future)

# Set-up Parallel processing
future::availableWorkers()
future::availableCores()

furrr::furrr_options(seed = TRUE)

future::plan(multisession, workers = 10)

```

```

# Set the initial parameters

# day_1 <- '2021-05-02'
# x <- '2021-05-02'

fetch_data_by_date_fn <- function(x, offset_zero = 0){
  options(scipen=999)
  # offset_zero = 870000
  folder_path <- paste(paste("df_", epitrix::clean_labels(x), sep = ""))
  fs::dir_create(here("data", folder_path))

  get_token_fn()
  get_data_fn()
  check_data_fn()

  while(StatusCode == 200 & TotalResults >= 10000){

    update_offset_fn()

    cat('This is offset number:', offset_zero , "\n")

    get_data_fn()

    check_data_fn()
  }
}

dates <- here("data") |>
  dir_info() |>
  slice(-1) |>
  slice(-31) |>
  pull(path) |>
  map_chr(fs::path_file) |>
  str_remove_all(., "df_") |>
  str_replace_all(., "_", "-") |>
  lubridate::ymd()

folder_path <- here("data") |>

```

```

dir_ls() |>
fs::path_dir() |>
unique()

folder_names <- here("data") |>
  dir_info() |>
  slice(-1) |>
  slice(-31) |>
  pull(path) |>
  str_remove(., folder_path)

file_prefix <- here("data") |>
  dir_info() |>
  slice(-1) |>
  slice(-31) |>
  pull(path) |>
  paste(folder_names, sep = "")

X <- here("data") |>
  dir_info() |>
  slice(-1) |>
  slice(-31) |>
  pull(path)

file_names <- X |>
  map(dir_ls)

check_last_offset <- function(x, y){
  x |>
  str_remove_all(paste(y, "_", sep = "")) |>
  str_remove_all(".csv") |>
  as.numeric() |>
  max()
}

dates_offset_list <- future_map2(file_names, file_prefix, check_last_offset) |>
  set_names(dates) |>
  enframe() |>
  unnest_wider(value) |>
  set_names(c("date", "offset")) |>
  mutate(offset = offset - 10000) |>
  deframe()

```

```

# possibly_fetch_data_by_date_fn = possibly(.f = fetch_data_by_date_fn,
#otherwise = "Error")
#
# # dates_offset_list <- dates_offset_list[c(1:2,11)]
# # dates_offset_list_old <- dates_offset_list
# dates_offset_list_old == dates_offset_list

# x <- furrr::future_map2(
#names(dates_offset_list),
#      dates_offset_list,
#      ~ possibly_fetch_data_by_date_fn(x = ..1,
#                                       offset_zero = ..2))

furrr::furrr_options(seed = NULL)
furrr::furrr_options(seed = TRUE)
future::plan(multisession, workers = length(dates_offset_list))

# availableCores()
# availableWorkers()

x <- furrr::future_map2(names(dates_offset_list),
                        dates_offset_list,
                        ~ possibly_fetch_data_by_date_fn(x = ..1,
                                                         offset_zero = ..2))

x |> enframe()
names(dates_offset_list)[30]
dates_offset_list[30]

possibly_fetch_data_by_date_fn(x = names(dates_offset_list)[1],
                              offset = dates_offset_list[30])

possibly_fetch_data_by_date_fn = possibly(.f = fetch_data_by_date_fn,
                                          otherwise = "Error")

selected_dates <- seq.Date(from = as.Date("2021-08-25"),
                          to = as.Date("2021-08-30"),
                          by = 1)
# selected_dates <- seq.Date(from = as.Date("2020-12-19"),

```

```

#to = as.Date("2020-12-20"),
#by = 1)
# fetch_data_by_date_fn("2021-05-17", offset_zero = 1450000)
# fetch_data_by_date_fn("2021-05-17", offset_zero = 0000000)
# x <- selected_dates |> map(possibly_fetch_data_by_date_fn)

furryr::furryr_options(seed = NULL)
furryr::furryr_options(seed = TRUE)
future::plan(multisession, workers = length(selected_dates))

x <- furryr::future_map(selected_dates,
                        possibly_fetch_data_by_date_fn,
                        offset_zero = 0)

here("data") |> fs::dir_tree()
here("data") |> map(fs::file_size)

library(tictoc)
tic()
x <- furryr::future_map(selected_dates,
                        possibly_fetch_data_by_date_fn,
                        offset_zero = 0)
toc()
x <- x |> set_names(selected_dates)
x
# selected_dates |> map(possibly_fetch_data_by_date_fn,
#offset_zero = 1000000)
# selected_dates |> map(possibly_fetch_data_by_date_fn,
#offset_zero = 2000000)
# fetch_data_by_date_fn("2021-05-21", offset_zero = 2140000)
# fetch_data_by_date_fn("2022-01-31", offset_zero = 1010000)
# fetch_data_by_date_fn("2022-01-24", offset_zero = 1720000)
# fetch_data_by_date_fn("2022-006-10", offset_zero = 460000)

dates <- here("data") |>
  dir_info() |>
  pull(path) |>
  map_chr(fs::path_file) |>
  str_remove_all(., "df_") |>
  str_replace_all(., "_", "-") |>
  lubridate::ymd()

```

```

folder_path <- here("data") |>
  dir_ls() |>
  fs::path_dir() |>
  unique()

folder_names <- here("data") |>
  dir_info() |>
  pull(path) |>
  str_remove(., folder_path)

file_prefix <- here("data") |>
  dir_info() |>
  pull(path) |>
  paste(folder_names, sep = "")

X <- here("data") |>
  dir_info() |>
  pull(path)

file_names <- X |>
  map(dir_ls)

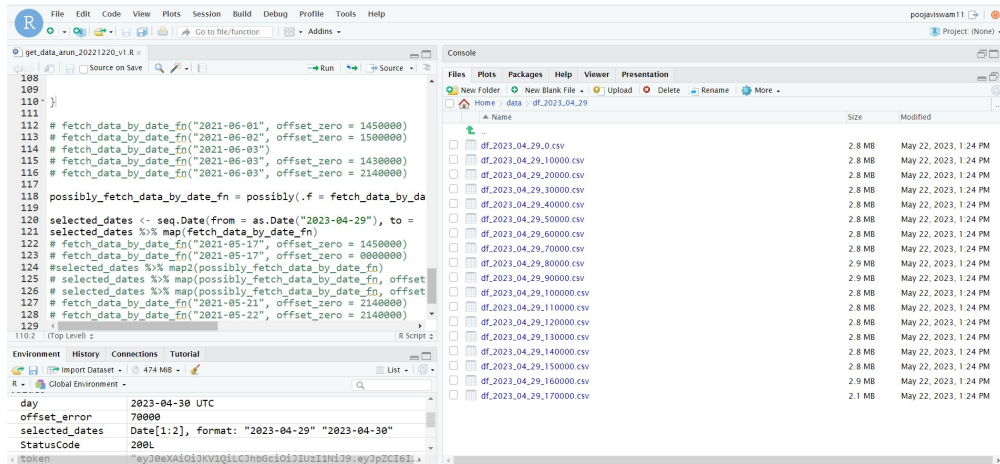
check_last_offset <- function(x, y){
  x |>
    str_remove_all(paste(y, "_", sep = "")) |>
    str_remove_all(".csv") |>
    as.numeric() |>
    max()
}

dates_offset_list <- future_map2(file_names, file_prefix, check_last_offset) |>
  set_names(dates) |>
  enframe() |>
  unnest_wider(value) |>
  set_names(c("date", "offset")) |>
  mutate(offset = offset - 10000) |>
  deframe()

# dates_offset_list <- dates_offset_list[c(11:28)]
# dates_offset_list_old <- dates_offset_list
# dates_offset_list_old == dates_offset_list

```





The advantages of this approach are:

1. **Automation and Efficiency:** The custom R functions automate the entire data-fetching process, including token generation, data retrieval, and offset management. This significantly reduces manual intervention, saving time and effort.
2. **Optimized Data Management:** The approach ensures that the data is fetched in a more memory-efficient .csv format, which is easier to handle than the bulkier .rds format. Additionally, the parallel processing allows the data to be fetched for multiple dates concurrently, speeding up the overall process.

## 3 Cleaning and Exploring the Data Variables

As part of the ongoing process of preparing the dataset for analysis, a key step involved cleaning and standardizing raw data. This phase focused on transforming the dataset into a more usable form by addressing inconsistencies, correcting errors, and categorizing variables into structured groups. The cleaning process involved handling missing values, identifying and removing outliers, and applying transformations to ensure uniformity across the dataset. These efforts are essential to ensuring that the data is both accurate and ready for further analytical procedures, enabling the generation of reliable insights for public health decision-making.

### 3.1 Looking at the variables

```
# Load Packages
library(tidyverse)
library(here)
library(fs)
library(arrow)

path <- paste("poojaviswam11","shared_folder", "data", "parquet_data",
             sep = "/")

# Create Folder Path
parquet_folder <- here() |>
  fs::path_dir() |>
  fs::path_dir() |>
  fs::path_dir() |>
  # fs::path_dir() |>
  paste(path, sep = '/')

# Load the dataset
df <- arrow::open_dataset(parquet_folder)

totalrows <- df |>
  nrow()
```

```
df |> glimpse()
```

```
> df %>% glimpse()
FileSystemDataset with 39 Parquet files
936,600,724 rows x 24 columns
$ icmr_id <string> "MpQ2q2j582SvqQHsvL19Pw==", "Y9LHwnYr2XwCO...
$ clinical_id <string> "t301aJPBqN+UVnE2iDOLUicn0/eUEOq/MhOXh5144...
$ age <double> 24, 73, 69, 77, 65, 70, 65, 44, 66, 63, 56...
$ age_in <string> "Years", "Years", "Years", "Years", "Years...
$ gender <string> "M", "F", "M", "F", "F", "F", "F", "F", "M...
$ patient_category <string> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
$ state_residence <string> "TELANGANA", "HARYANA", "HARYANA", "HARYAN...
$ district_residence <string> "HYDERABAD", "GURUGRAM", "GURUGRAM", "GURU...
$ lab_id <double> 7, 1, 1, 1, 1, 1, 1, 1, 1, 1, 112, 112, 11...
$ date_sample_collection <timestamp[us, tz=UTC]> 2020-03-01, 2020-03-03, 2020-03-03, 2020-0...
$ date_sample_received <timestamp[us, tz=UTC]> 2020-03-01, 2020-03-03, 2020-03-03, 2020-0...
$ date_of_sample_tested <timestamp[us, tz=UTC]> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
$ symptoms <string> "Fever, Sore Throat", "Asymptomatic at sam...
$ underlying_medical_condition <string> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
$ testing_kit_used <string> "ICMR-NIV Protocol", "ICMR-NIV Protocol", ...
$ egene_screening <string> "Positive", "Positive", "Positive", "Posit...
$ ct_value_screening <double> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, 28...
$ rdrp_confirmatory <string> "Positive", NA, NA, NA, NA, NA, NA, NA, NA, NA...
$ ct_value_rdrp <string> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, "3...
$ orfib_confirmatory <string> "Positive", NA, NA, NA, NA, NA, NA, NA, NA, NA...
$ ct_value_orfib <double> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
$ final_test_result <string> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, "P...
$ entry_date <timestamp[us, tz=UTC]> 2020-03-01, 2020-03-03, 2020-03-03, 2020-0...
$ updated_on <timestamp[us, tz=UTC]> 2020-10-19 07:20:33, 2020-10-19 07:20:31, ...
Call `print()` for full schema details
>
```

Figure 3.1: Screenshot of showing all the variables in the dataset in a glimpse

## 3.2 Patient Category

- According to the SRF Forms (7 versions)
  - [https://www.icmr.gov.in/pdf/covid/labs/archive/SRF\\_v3.pdf](https://www.icmr.gov.in/pdf/covid/labs/archive/SRF_v3.pdf) - 03/04/2020
  - [https://www.icmr.gov.in/pdf/covid/labs/archive/SRF\\_v9.pdf](https://www.icmr.gov.in/pdf/covid/labs/archive/SRF_v9.pdf) - 22/04/2020
  - [https://www.icmr.gov.in/pdf/covid/labs/archive/SRF\\_v11.pdf](https://www.icmr.gov.in/pdf/covid/labs/archive/SRF_v11.pdf) - 25/05/2020
  - [https://www.icmr.gov.in/pdf/covid/labs/archive/SRF\\_v12.pdf](https://www.icmr.gov.in/pdf/covid/labs/archive/SRF_v12.pdf) - 15/09/2020
  - [https://www.icmr.gov.in/pdf/covid/labs/archive/Revised\\_SRF\\_Form\\_16122020\\_1.pdf](https://www.icmr.gov.in/pdf/covid/labs/archive/Revised_SRF_Form_16122020_1.pdf) - 16/12/2020
  - [https://www.icmr.gov.in/pdf/covid/labs/archive/Revised\\_SRF\\_Form\\_22032021\\_1.pdf](https://www.icmr.gov.in/pdf/covid/labs/archive/Revised_SRF_Form_22032021_1.pdf) - 07/04/2021
  - [https://www.icmr.gov.in/pdf/covid/labs/archive/Revised\\_SRF\\_Form\\_26052021\\_1.pdf](https://www.icmr.gov.in/pdf/covid/labs/archive/Revised_SRF_Form_26052021_1.pdf) - 26/05/2021

### 3.2.1 In the Patient Category form, the category get differs in different versions

Cat 1: Symptomatic international traveller in last 14 days

Cat 2: Symptomatic contact of lab confirmed case

Cat 3: Symptomatic healthcare worker

Cat 4: Hospitalized SARI (Severe Acute Respiratory Illness) patient Cat 5a: Asymptomatic direct and high risk contact of confirmed case – family member

Cat 5b: Asymptomatic healthcare worker in contact with confirmed case without adequate protection

Cat 1: All symptomatic (ILI symptoms) cases

Cat 2: All asymptomatic high-risk individuals (Any individual who falls under Section B2)

Cat 3: All symptomatic (ILI symptoms) individuals with history of international travel in the last 14 days

Cat 4: All individuals who wish to get themselves tested

Cat 1: All patients of Severe Acute Respiratory Infection (SARI)

Cat 2: All symptomatic (ILI symptoms) patients presenting in a healthcare setting

Cat 3: Asymptomatic high-risk patients who are hospitalized or seeking immediate hospitalization

Cat 4: Asymptomatic patients undergoing surgical / non-surgical invasive procedures (not to be tested more than once a week during hospital stay).

Cat 5: All pregnant women in/near labour who are hospitalized for delivery

Cat 6: All symptomatic neonates presenting with acute respiratory / sepsis like illness

Cat 7: Patients presenting with atypical manifestations [stroke, encephalitis, pulmonary embolism, acute coronary symptoms, Guillain Barre syndrome, Multi-system Inflammatory Syndrome in Children (MIS-C), progressive gastrointestinal symptoms] based on the discretion of the treating physician

Cat 8: All individuals who wish to get themselves tested

### **3.2.2 Patient Category given by ICMR**

#### **For Community Settings**

Patient Category Code Patient Category

NCat1 All symptomatic (ILI symptoms) cases.

NCat3 All asymptomatic high-risk individuals.

NCat4 All symptomatic (ILI symptoms) individuals with history of international travel in the last 14 days.

NCat17 All individuals who wish to get themselves tested.

#### **For Hospital Settings**

Patient Category Code Patient Category

NCat9 All patients of Severe Acute Respiratory Infection (SARI).

NCat10 All symptomatic (ILI symptoms) patients presenting in a healthcare setting. NCat11 Asymptomatic high-risk patients who are hospitalized or seeking immediate hospitalization.

NCat12 Asymptomatic patients undergoing surgical / non-surgical invasive procedures (not to be tested more than once a week during hospital stay). NCat13 All pregnant women in/near labour who are hospitalized for delivery.

NCat14 All symptomatic neonates presenting with acute respiratory / sepsis like illness.

NCat15 Patients presenting with atypical manifestations.[stroke, encephalitis, pulmonary embolism, acute coronary symptoms, Guillain Barre syndrome, Multi-system Inflammatory Syndrome in Children (MIS-C), progressive gastrointestinal symptoms]

NCat18 All individuals who wish to get themselves tested.

### 3.2.3 Missing Data

- Patient Category: 0.27 %
- Symptoms: 97.8 %
- Underlying Medical Condition: 99.11 %
- Missing Symp & UMC: 96.99 %
- All Three: 0.26 %

#### 3.2.3.1 Missing Data among Positives

- Patient Category: 0.04 %
- Symptoms: 5.02 %
- Underlying Medical Condition: 5.35 %
- Missing Symp & UMC: 4.96 %
- All Three: 0.04 %

## 3.3 Clean test status positive

We have to clean the variable **final\_test\_result** and categorize as positive, negative and others. We have created a master list of each categories into a .csv file for cleaning the variable final\_test\_result. We have to load the parquet file of each month wise with the master list .csv file for cleaning data.

	A	B	C
1	positive	negative	NA
2	postive	nagative	none
3	posotive	negetive	under process
4	positive	neagative	sample rajected
5	positive	neative	sample rejected
6	posetive	negaative	sample rejcted
7	possitive	nigetive	sample not received
8	detected	negaitive	recollection
9	Antigen Positive	ngative	reject
10	Positive	negativr	sent to rtPCR
11	TrueNAT Screening Positive	nenative	indeterminate
12	ANTIGEN Positive	negattive	sample reject
13	POSITIVE	neagative	rtPCR
14	ANTIGEN POSITIVE	negitive	spillage
15	POSITIVER	negativ	repeat
16	Postive	negitive	am
17	Antigen POSITIVE	not detected	sample rected
18	Antigen POSSITIVE	nagative	result
19	POSETIVE	not not detected	pending
20	Antigen positive	Antigen Negative	invalid
21	antigen positive	Negative	immunoquick_ag
22	Antigen Screening Positive	TrueNAT Screening Negative	invalid
23	Positve	ANTIGEN Negative	select result
24	Anitgen Positive	NEGATIVE	spillageinconclusive_spillage_rejected
25	Antigen Positve	ANTIGEN NEGATIVE	not done
26	Posotive	NEGATIVE	inconclusive
27	POSITVE	Nagative	refer to covai
28	ANTIGENE POSITIVE	NEGATIVE	repeat sample
29	Rapid Positive	NEAGATIVE	sample rejected
30	pOSITIVE	Antigen negative	snr
31	positive	NEGATIV	rejected

Figure 3.2: Screenshot of Master List for categorizing testing data

The custom function for cleaning test status positive:

1. `test_status_positive_fn`: Function for cleaning the variable `final_test_result` data

### 3.3.0.1 Clean test status positive

```
# Function for cleaning final_test_result variable

test_status_positive_fn <- function(parquet_month){

  # Path
  # parquet_month <- "2020_03"

  path <- paste("poojaviswam11", "shared_folder","data", "parquet_data", sep = "/")

  # Create Folder Path
  parquet_folder <- here() |>
  fs::path_dir() |>
```

```

fs::path_dir() |>
fs::path_dir() |>
paste(path,parquet_month, sep = '/')

parquet_folder |>
  fs::dir_tree()

# Load the dataset
df <- arrow::open_dataset(parquet_folder)

# Load test_masterlist.csv
test_path <- paste("poojavismam11", "shared_folder",
                  "data", "master_csv","test_masterlist.csv", sep = "/")

# Load the master file and rename the categories as positive,
#negative and others
test_masterlist <- here() |>
  fs::path_dir() |>
  fs::path_dir() |>
  fs::path_dir() |>
  paste(test_path, sep = '/') |>
  read_csv("test_masterlist.csv") |>
  mutate(across(everything(), ~ str_replace_all(., '\\xa0' , " "))) |>
  set_names("positive", "negative", "others")

# Make the unique list of categories positive, negative and others
pos_sample <- test_masterlist |>
  select(positive) |>
  drop_na() |>
  pull(positive) |>
  unique()

neg_sample <- test_masterlist |>
  select(negative) |>
  drop_na() |>
  pull(negative) |>
  unique()

oth_sample <- test_masterlist |>
  select(others) |>
  drop_na() |>
  pull(others) |>

```

```

unique()

# Create a new variable in the dataset as final_test_status and listing
# out the matching list to the new variable.
df <- df |>
  mutate(final_test_status = NA_character_) |>
  mutate(final_test_status =
    case_when(
      (final_test_result %in% pos_sample) ~ "positive",
      (final_test_result %in% neg_sample) ~ "negative",
      (final_test_result %in% oth_sample) ~ "others"))

# Create output folder and output file and write the cleaned output
# as new parquet file.

out_folder <- parquet_folder |> str_replace("parquet_data", "cleaned_data")

out_folder |> fs::dir_create()

out_file <- paste(parquet_month, "_cleaned_test_status.parquet", sep = "")

df |>
  dplyr::select(c(icmr_id, final_test_status)) |>
  write_parquet(here(out_folder, out_file))
}

```

```

# Test Status Clean

# Load Packages
library(tidyverse)
library(here)
library(fs)
library(arrow)
library(lubridate)

# Functions

# Test Status Function
test_status_positive_fn <- function(parquet_month){

```

```

# Path
# parquet_month <- "2020_03"

path <- paste("poojavismam11", "shared_folder","data", "parquet_data", sep = "/")

# Create Folder Path
parquet_folder <- here() |>
  fs::path_dir() |>
  fs::path_dir() |>
  fs::path_dir() |>
  paste(path,parquet_month, sep = '/')

parquet_folder |>
  fs::dir_tree()

# Load the dataset
df <- arrow::open_dataset(parquet_folder)

# Load test_masterlist.csv
test_path <- paste("poojavismam11", "shared_folder",
                  "data", "master_csv","test_masterlist.csv", sep = "/")

# Load the master file and rename the categories as positive,
#negative and others
test_masterlist <- here() |>
  fs::path_dir() |>
  fs::path_dir() |>
  fs::path_dir() |>
  paste(test_path, sep = '/') |>
  read_csv("test_masterlist.csv") |>
  mutate(across(everything(), ~ str_replace_all(., '\\xa0' , " "))) |>
  set_names("positive", "negative", "others")

# Make the unique list of categories positive, negative and others
pos_sample <- test_masterlist |>
  select(positive) |>
  drop_na() |>
  pull(positive) |>
  unique()

neg_sample <- test_masterlist |>
  select(negative) |>

```

```

drop_na() |>
pull(negative) |>
unique()

oth_sample <- test_masterlist |>
  select(others) |>
  drop_na() |>
  pull(others) |>
  unique()

# Create a new variable in the dataset as final_test_status and listing
#out the matching list to the new variable.
df <- df |>
  mutate(final_test_status = NA_character_) |>
  mutate(final_test_status =
    case_when(
      (final_test_result %in% pos_sample) ~ "positive",
      (final_test_result %in% neg_sample) ~ "negative",
      (final_test_result %in% oth_sample) ~ "others"))

# Create output folder and output file and write the cleaned output
#as new parquet file.
out_folder <- parquet_folder |> str_replace("parquet_data", "cleaned_data")

out_folder |> fs::dir_create()

out_file <- paste(parquet_month, "_cleaned_test_status.parquet", sep = "")

df |>
  dplyr::select(c(icmr_id, final_test_status)) |>
  write_parquet(here(out_folder, out_file))
}

# Map the function for all the monthly folders
folder_names <- parquet_folder |> fs::path_dir() |> dir() |> unique()

folder_names |> map(test_status_positive_fn)

```

The output will be in parquet file with cleaned data and having variables icmr\_id and final\_test\_status.

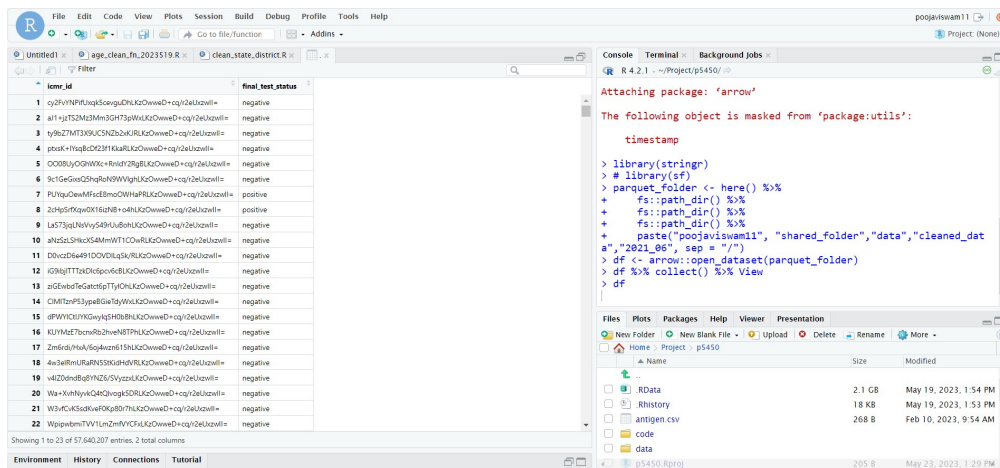


Figure 3.3: Output of cleaning final\_test\_result

### 3.4 Clean Age and Gender Variables

We have to clean the variables age and gender. Here also we are taking input as the monthly parquet data.

The custom function for cleaning age and gender:

1. age\_clean\_fn: Combined Function for cleaning the variables age and gender

#### 3.4.1 Clean the age variable

```
#Function for cleaning age variable

age_clean_fn <- function(parquet_month){

  # Path
  # parquet_month <- "2022_10"

  path <- paste("poojaviswam11", "shared_folder","data", "parquet_data", sep = "/")

  # Create Folder Path
  parquet_folder <- here() |>
  fs::path_dir() |>
  fs::path_dir() |>
  fs::path_dir() |>
  paste(path,parquet_month, sep = '/')

  parquet_folder |>
```

```

fs::dir_tree()

# Load the dataset
df <- arrow::open_dataset(parquet_folder)

# Cleaning age_in variable has categorized year to Years
df <- df |> mutate(age_in = ifelse(age_in == 'year', "Years", age_in))

# We need to convert the age variable with age_in variable.
# The age_in variable has Years, Month and Days categories.
df <- df |>
  mutate(age_in_years = case_when(age_in == "Years" ~ age,
                                  age_in == "Months" ~ age/12,
                                  age_in == "Days" ~ age/365))

# We need to clean the age_in_years variable taking less than and equal
# to age 126 and discarding above 126 age
df <- df |>
  mutate(age_in_years_cleaned = ifelse(age_in_years < 126, T, F))

# Age Groups are categorized from age 1 to 80 and above with levels

df <- df |>
  mutate(age_groups =
    case_when(
      (age_in_years_cleaned == T) & (age_in_years < 1) ~ "Infant",
      (age_in_years_cleaned == T) & (age_in_years < 5) ~ "Under-5",
      (age_in_years_cleaned == T) & (age_in_years < 10) ~ "5-10",
      (age_in_years_cleaned == T) & (age_in_years < 20) ~ "10-20",
      (age_in_years_cleaned == T) & (age_in_years < 30) ~ "20-30",
      (age_in_years_cleaned == T) & (age_in_years < 40) ~ "30-40",
      (age_in_years_cleaned == T) & (age_in_years < 50) ~ "40-50",
      (age_in_years_cleaned == T) & (age_in_years < 60) ~ "50-60",
      (age_in_years_cleaned == T) & (age_in_years < 70) ~ "60-70",
      (age_in_years_cleaned == T) & (age_in_years < 80) ~ "70-80",
      (age_in_years_cleaned == T) & (age_in_years >= 80) ~ "80+",
      TRUE ~ NA_character_
    ))

```

```

df <- df |> collect() |>
  mutate(age_groups = factor(age_groups, levels = c("Infant",
                                                  "Under-5",
                                                  "5-10",
                                                  "10-20",
                                                  "20-30",
                                                  "30-40",
                                                  "40-50",
                                                  "50-60",
                                                  "60-70",
                                                  "70-80",
                                                  "80+")))

df |>
count(age_groups) |>
summarize(age_groups, n, prop = round((n/sum(n))*100, 2))
}

```

### 3.4.2 Clean the gender variable

```

# Code for cleaning gender variable
df <- df |>
mutate(gender_cleaned = case_when(gender %in% c("F", "M", "T") ~ gender ,
                                TRUE ~ NA_character_))

# Categorize the gender variable to Female, Male and Transgender
df <- df |>
mutate(gender_cleaned = case_when(
  (gender_cleaned == "F") ~ "Female" ,
  (gender_cleaned == "M") ~ "Male",
  (gender_cleaned == "T") ~ "Transgender",
  TRUE ~ NA_character_))

df |>
count(age_groups, gender_cleaned )

```

```

# Load Packages
library(tidyverse)
library(here)
library(fs)
library(arrow)

```

```

library(lubridate)

# Create the function class

age_clean_fn <- function(parquet_month){

  # Path
  # parquet_month <- "2022_10"

  path <- paste("poojavismam11", "shared_folder","data", "parquet_data", sep = "/")

  # Create Folder Path
  parquet_folder <- here() |>
    fs::path_dir() |>
    fs::path_dir() |>
    fs::path_dir() |>
    paste(path,parquet_month, sep = '/')

  parquet_folder |>
    fs::dir_tree()

  # Load the dataset
  df <- arrow::open_dataset(parquet_folder)

  # Cleaning age_in variable has categorized year to Years
  df <- df |> mutate(age_in = ifelse(age_in == 'year', "Years", age_in))

  # We need to convert the age variable with age_in variable.
  # The age_in variable has Years, Month and Days categories.
  df <- df |>
    mutate(age_in_years = case_when(age_in == "Years" ~ age,
                                     age_in == "Months" ~ age/12,
                                     age_in == "Days" ~ age/365))

  # We need to clean the age_in_years variable taking less than and equal
  # to age 126 and discarding above 126 age
  df <- df |>
    mutate(age_in_years_cleaned = ifelse(age_in_years < 126, T, F))

```

```

# Age Groups are categorized from age 1 to 80 and above with levels

df <- df |>
  mutate(age_groups =
    case_when(
      (age_in_years_cleaned == T) & (age_in_years < 1) ~ "Infant",
      (age_in_years_cleaned == T) & (age_in_years < 5) ~ "Under-5",
      (age_in_years_cleaned == T) & (age_in_years < 10) ~ "5-10",
      (age_in_years_cleaned == T) & (age_in_years < 20) ~ "10-20",
      (age_in_years_cleaned == T) & (age_in_years < 30) ~ "20-30",
      (age_in_years_cleaned == T) & (age_in_years < 40) ~ "30-40",
      (age_in_years_cleaned == T) & (age_in_years < 50) ~ "40-50",
      (age_in_years_cleaned == T) & (age_in_years < 60) ~ "50-60",
      (age_in_years_cleaned == T) & (age_in_years < 70) ~ "60-70",
      (age_in_years_cleaned == T) & (age_in_years < 80) ~ "70-80",
      (age_in_years_cleaned == T) & (age_in_years >= 80) ~ "80+",
      TRUE ~ NA_character_
    )
  )

df <- df |> collect() |>
  mutate(age_groups = factor(age_groups, levels = c("Infant",
    "Under-5",
    "5-10",
    "10-20",
    "20-30",
    "30-40",
    "40-50",
    "50-60",
    "60-70",
    "70-80",
    "80+")))

df |>
  count(age_groups) |>
  summarize(age_groups, n, prop = round((n/sum(n))*100, 2))

# Clean the gender
df <- df |>
  mutate(gender_cleaned = case_when(gender %in% c("F", "M", "T") ~ gender,
    TRUE ~ NA_character_))

```

```

# Categorize the gender variable to Female, Male and Transgender
df <- df |>
  mutate(gender_cleaned =
    case_when(
      (gender_cleaned == "F") ~ "Female" ,
      (gender_cleaned == "M") ~ "Male",
      (gender_cleaned == "T") ~ "Transgender",
      TRUE ~ NA_character_))

#df_1 |> names()

df |>
  count(age_groups, gender_cleaned)

# Create output folder and output file and write the cleaned output as
# new parquet file.
cleaned_path <- parquet_folder |>
  str_replace("parquet_data","parquet_age_gender_clean_data" )

cleaned_path |> fs::dir_create()

cleaned_parquet_filename <- paste(parquet_month, "_cleaned.parquet",
                                  sep = "")

df_1 |>
  dplyr::select(c(icmr_id, age_in_years,age_in_years_cleaned,
                 age_groups,gender_cleaned)) |>
  write_parquet(here(cleaned_path, cleaned_parquet_filename))
}

#x <- '2021_10'
#age_clean_fn(x)

# Map the function for all the monthly folders
folder_names <- parquet_folder |> fs::path_dir() |> dir() |> unique()

folder_names |> map(age_clean_fn)

```

The output will be in parquet file with cleaned data and having variables `icmr_id` `age_in_years`, `age_in_years_cleaned` and `gender_cleaned`.

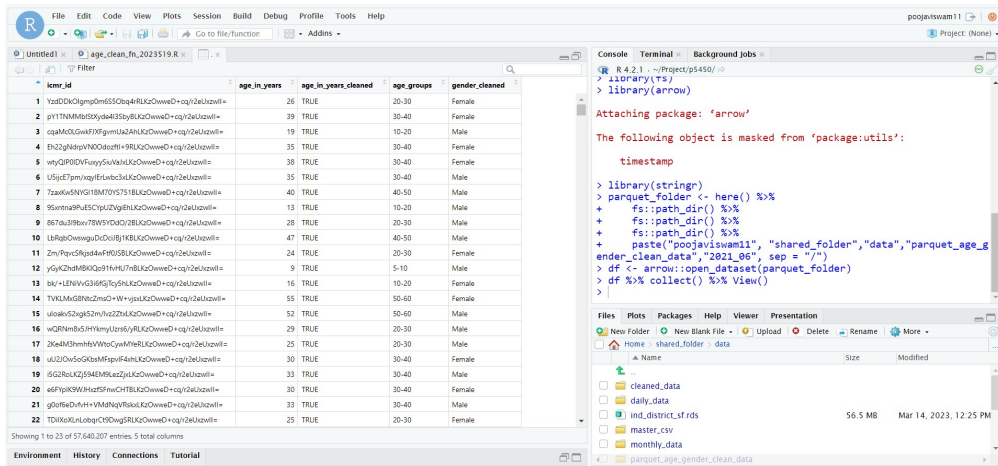


Figure 3.4: Output of cleaning age and gender

### 3.5 Cleaning of the states and their districts:

We have to clean the variables `state_residence` and `district_residence`. Here also we are taking input as the whole parquet data files.

```
#Code for cleaning the variables state_residence and district_residence in
#COVID19 Dataset
```

```
df <- df |>
  mutate(district = str_trim(str_to_lower(district_residence)))
```

```
df <- df |>
  mutate(statename = str_trim(str_to_lower(state_residence)))
```

```
df <- df |>
  mutate(district = (str_replace_all(district, " ", "_")))
```

```
df <- df |>
  mutate(statename = (str_replace_all(statename, " ", "_")))
```

```
# Clean the states names
```

```
df <- df |>
  mutate(statename =
    case_when(
      statename == "jammu_and_kashmir" ~ "jammu_kashmir",
      statename == "andaman_and_nicobar_islands" ~ "andaman_nicobar",
      statename == "dadra_and_nagar_haveli" ~ "dadra_nagar_have",
```

```

statename == "daman_and_diu" ~ "daman_diu",
TRUE ~ statename))

# We need to create a new variable as district_name where we can combine
#district and state names as district_statename

df <- df |> mutate(district_name = paste(district, statename, sep = "_"))

df1 <- df |>
select(statename, district, district_name) |>
distinct() |>
collect()

df1 |>
write_csv(here("data", "state_dist_covid.csv"))

# Create a dataframe for covid19 dataset for cleaning the districts
dfcovid <- read_csv(here("data", "state_dist_covid.csv"))

#Load the shape file

ind_district_sf <- read_rds(here("shared_folder",
"data",
"ind_district_sf.rds"))

# ind_district_sf |>
ggplot() +
geom_sf()

#Clean the shape file

ind_district_df <- ind_district_sf |>
st_drop_geometry() |>
as_tibble()

ind_district_df <- ind_district_df |>
rename(state = statename)

ind_district_df <- ind_district_df |>
mutate(statename = epitrix::clean_labels(state))

ind_district_df <- ind_district_df |>
rename(district_name = district)

```

```

ind_district_df <- ind_district_df |>
rename(district = distname)

ind_district_df <- ind_district_df |>
mutate(distname = epitrix::clean_labels(district))

```

```

##Find the common district names between both datasets

```

```

dfcovid |>
nrow()
ind_district_df |>
nrow()

dfcovid$district_name[dfcovid$district_name %in% ind_district_df$district_name]
ind_district_df$district_name[ind_district_df$district_name %in% dfcovid$district_name]

# Find difference between both dataframes

dfcovid$district_name[!dfcovid$district_name %in% ind_district_df$district_name]
ind_district_df$district_name[!ind_district_df$district_name %in% dfcovid$district_name]

# Joining Districts of COVID19 with shape file
df_joined <- dfcovid |>
left_join(ind_district_df |>
select(district_name) |>
st_drop_geometry(), by = "district_name")

df_joined |>
View()

```

```

# Load Parquet Files and explore data

```

```

# Load Packages
library(tidyverse)
library(here)
library(fs)
library(arrow)
library(stringr)
library(sf)

```

```

# Create Folder Path
parquet_folder <- here() |>

```

```

fs::path_dir() |>
fs::path_dir() |>
fs::path_dir() |>
paste("poojavismam11", "shared_folder","data", "parquet_data", sep = "/")

df <- arrow::open_dataset(parquet_folder)

#Clean the variables state_residence and district_residence in COVID19 Dataset

df <- df |>
  mutate(district = str_trim(str_to_lower(district_residence)))

df <- df |>
  mutate(statename = str_trim(str_to_lower(state_residence)))

df <- df |>
  mutate(district = (str_replace_all(district," ", "_")))

df <- df |>
  mutate(statename = (str_replace_all(statename," ", "_")))

# Clean the states names
# Clean the states names
df <- df |>
  mutate(statename =
    case_when(
      statename == "jammu_and_kashmir" ~ "jammu_kashmir",
      statename == "andaman_and_nicobar_islands" ~ "andaman_nicobar",
      statename == "dadra_and_nagar_haveli" ~ "dadra_nagar_have",
      statename == "daman_and_diu" ~ "daman_diu",
      TRUE ~ statename))

df <- df |>
  mutate(district_name = paste(district, statename, sep = "_"))

df1 <- df |>
  select(statename, district, district_name) |>
  distinct() |>
  collect()

df1 |>

```

```

write_csv(here("data", "state_dist_covid.csv"))

dfcovid <- read_csv(here("data", "state_dist_covid.csv"))

#Load the shape file

ind_district_sf <- read_rds(here("shared_folder","data","ind_district_sf.rds"))

# ind_district_sf |> ggplot() + geom_sf()

#Clean the shape file

ind_district_df <- ind_district_sf |>
  st_drop_geometry() |>
  as_tibble()
ind_district_df <- ind_district_df |>
  rename(state = statename)
ind_district_df <- ind_district_df |>
  mutate(statename = epitrix::clean_labels(state))
ind_district_df <- ind_district_df |>
  rename(district_name = district)
ind_district_df <- ind_district_df |>
  rename(district = distname)
ind_district_df <- ind_district_df |>
  mutate(distname = epitrix::clean_labels(district))

##Find the common district names and difference between both dataframes

#(dfcovid$district_name |> unique()) %in% (ind_district_df$district_name
# |> unique())
#intersect(dfcovid$district_name,ind_district_df$district_name)

com_dist <- intersect(
  unique(dfcovid$district_name),
  unique(ind_district_df$district_name))

com_dist <- as.data.frame(com_dist)

diff_dist <- as.data.frame(
  setdiff(
    unique(ind_district_df$district_name) ,
    unique(dfcovid$district_name)))

```

```

setdiff(
  unique(ind_district_df$district_name) ,
  unique(dfcovid$district_name))
## Create variable names

names(diff_dist) <- c("mismatch")
names(com_dist) <- c("match")

mdf <- bind_rows(com_dist, diff_dist)

mdf |>
  write_csv(here('data', 'com_mis_dist.csv'))

dfcovid |> nrow()
ind_district_df |> nrow()

# Joining Districts of COVID19 with shape file
df_joined <- dfcovid |>
  left_join(ind_district_df |>
    select(district_name) |>
    st_drop_geometry(), by = "district_name")

df_joined |>
  View()

```

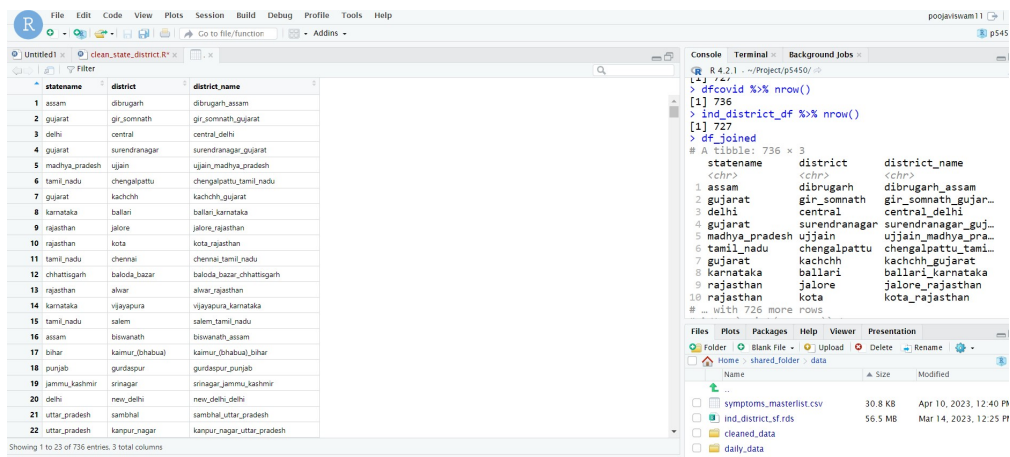


Figure 3.5: Screenshot of output of COVID19 Districts

### 3.6 Cleaning the symptoms:

We have to clean the variable symptoms. The challenges we faced while cleaning the symptoms is that they are distributed in other variables underlying\_medical\_conditions also. So we have write functions for cleaning variables symptoms, underlying\_medical\_conditions.

Here we are taking input as the monthly parquet data file. We have created a master list of each categories of symptoms into a .csv file for cleaning the variable.

The screenshot shows a large table with columns labeled A through W. Each row contains a list of related medical symptoms and conditions, such as 'breathless myalgia fever', 'headache', 'abd\_pain', 'cough', 'sore\_throat', 'vomiting', 'cold', 'malaise', 'diarrhoea', 'loss\_of\_smell', 'loss\_of\_taste', 'shivering', 'asymptomatic', 'contact', 'anc\_pnc', 'others', 'repeat\_test', 'influenza', 'checkup', 'hypertensi', 'asthma', 'drt', etc. The table is a comprehensive list of symptoms used for data cleaning.

Figure 3.6: Screenshot of Master list of symptoms categories

symp\_clean\_fn: Function for cleaning the symptoms

```
# Function for Cleaning Symptoms

symp_clean_fn <- function(x, y){
  # x = symp_cat
  # y = 'fever'
  y_new = paste0(y, '_1')
  y_versions <- symptoms_masterlist |>
  dplyr::select(all_of(y)) |>
  drop_na() |>
  pull(1)
  x <- x |>
  mutate({{y_new}} := map_lgl(x$symp_cat_new_chr, ~str_detect(.,
  paste(y_versions, collapse = '|'))))
  x
}
```

```
# Load Parquet Files and explore data

# Load Packages
```

```

library(tidyverse)
library(here)
library(fs)
library(arrow)

# Path
path <- paste("poojavismam11", "shared_folder","data", "parquet_data",
              "2021_12", sep = "/")

# Create Folder Path
parquet_folder <- here() |>
  fs::path_dir() |>
  # fs::path_dir() |>
  # fs::path_dir() |>
  paste(path, sep = '/')

here() |>
  fs::path_dir() |>
  paste(path, sep = '/') |>
  fs::dir_ls()

# Load the dataset
df <- arrow::open_dataset(parquet_folder)

# Load Symptoms Masterlist
symptoms_masterlist <- read_csv(here("shared_folder","data", "master_csv",
                                     "symptoms_masterlist.csv"))
symptoms_masterlist <- symptoms_masterlist |>
  mutate(across(everything(), ~ str_replace_all(., '\\xa0' , " ")))

# Explore the dataset
# df |>
#   count(symptoms) |>
#   arrange(-n) |>
#   collect() |>
#   slice(1:50) |>
#   View()

# Clean the data
df |> names()

```

```

symp_df <- df |>
  dplyr::select(symptoms) |>
  filter(!is.na(symptoms))

symp_df <- symp_df |>
  mutate(symptoms_new = str_trim(str_to_lower(symptoms)))

symp_cat <- symp_df |>
  count(symptoms_new) |>
  arrange(-n) |>
  collect()

symp_cat <- symp_cat |>
  mutate(commas = str_detect(symptoms_new, ","))

symp_cat <- symp_cat |>
  mutate(symp_cat_new = str_split(symptoms_new, ","))

symp_cat <- symp_cat |>
  mutate(symp_cat_new_chr = map(symp_cat_new, paste, collapse = " "))

symp_clean_fn <- function(x, y){
  # x = symp_cat
  # y = 'fever'
  y_new = paste0(y, '_1')
  y_versions <- symptoms_masterlist |>
    dplyr::select(all_of(y)) |>
    drop_na() |>
    pull(1)
  x <- x |>
    mutate({{y_new}} := map_lgl(x$symp_cat_new_chr,
                                ~str_detect(.,
                                              paste(y_versions, collapse = '|'))))
  x
}

symp_cat <- symp_cat |>
  symp_clean_fn('fever') |>
  symp_clean_fn('breathlessness') |>
  symp_clean_fn('myalgia') |>
  symp_clean_fn('headache') |>

```

```
symp_clean_fn('abd_pain') |>
symp_clean_fn('cold') |>
symp_clean_fn('malaise') |>
symp_clean_fn('diarrhoea') |>
symp_clean_fn('sore_throat') |>
symp_clean_fn('vomiting') |>
symp_clean_fn('cough') |>
symp_clean_fn('shivering_chills')|>
symp_clean_fn('asymptomatic')|>
symp_clean_fn('contact')|>
symp_clean_fn('anc_pnc') |>
symp_clean_fn('loss_of_smell') |>
symp_clean_fn('loss_of_taste') |>
symp_clean_fn('others') |>
symp_clean_fn('repeat_test') |>
symp_clean_fn('influenza_like_illness') |>
symp_clean_fn('checkup') |>
symp_clean_fn('hypertension') |>
symp_clean_fn('asthma') |>
symp_clean_fn('diabetes') |>
symp_clean_fn('travel') |>
symp_clean_fn('flw') |>
symp_clean_fn('obesity') |>
symp_clean_fn('heart_disease') |>
symp_clean_fn('chronic_kidney_disease') |>
symp_clean_fn('malignancy') |>
symp_clean_fn('stroke')
```

```
symp_cat |>
  filter(fever_1 == F,
         breathlessness_1 == F,
         myalgia_1 == F,
         headache_1 == F,
         abd_pain_1 == F,
         cold_1 == F,
         malaise_1 == F,
         diarrhoea_1 == F,
         sore_throat_1 == F,
         vomiting_1 == F,
         cough_1 == F,
         shivering_chills_1 == F,
```

```

asymptomatic_1 == F,
contact_1 == F,
anc_pnc_1 == F,
loss_of_smell_1 == F,
loss_of_taste_1 == F,
others_1 == F,
repeat_test_1 == F,
influenza_like_illness_1 == F,
checkup_1 == F,
hypertension_1 == F,
asthma_1 == F,
diabetes_1 == F,
travel_1 == F,
flw_1 == F,
obesity_1 == F,
heart_disease_1 == F,
chronic_kidney_disease_1 == F,
malignancy_1 == F,
stroke_1 == F) |>
dplyr::select(-symp_cat_new) |>
View()

```

```

symp_cat |>
  filter(
    fever_1 == F,
    breathlessness_1 == F,
    myalgia_1 == F,
    headache_1 == F,
    abd_pain_1 == F,
    cold_1 == F,
    malaise_1 == F,
    diarrhoea_1 == F,
    sore_throat_1 == F,
    vomiting_1 == F,
    cough_1 == F,
    shivering_chills_1 == F,
    asymptomatic_1 == F,
    contact_1 == F,
    anc_pnc_1 == F,
    loss_of_smell_1 == F,
    loss_of_taste_1 == F,
    others_1 == F,
    repeat_test_1 == F,

```

```

influenza_like_illness_1 == F,
checkup_1 == F,
hypertension_1 == F,
asthma_1 == F,
diabetes_1 == F,
travel_1 == F,
flw_1 == F,
obesity_1 == F,
heart_disease_1 == F,
chronic_kidney_disease_1 == F,
malignancy_1 == F,
stroke_1 == F) |>
select(symptoms_new) |>
mutate(symptoms_new = as.character(symptoms_new)) |>
View()

```

### 3.7 Cleaning the Testing Kits Used

We have to clean the variable `testing_kit_used`. Here we are taking input as the monthly parquet data file. We have created a master list of each categories of testing kits as `rtPCR`, `antigen` and other into a `.csv` file for cleaning the variable.

	A	B	C
1	icmr-niv protocol	ag-sd_biosensor_standard_q_covid-19_ag_detection_kit	triviron health care pvt.ltd
2	abi_taqpath	sd biosensor standard q covid-19 ag detection kit	zybio_sars_kit
3	labgun-tm_exofast	meril_ag	angstrom biotech pvt.ltd
4	triviron_covidsure	alpine_antigen_kit	euroimmun
5	meril_rtprc	labcare_diagnostics_antigen_lateral	genepath_covidx
6	true_pcr_kelpest	biocard_pro_ag_triviron	perkinelmer_coronavirus_detection_kit
7	genes2me_viraltdect	sd_biosensor	maccura_sars-cov-2
8	gb_sars_rtprc_kit	angcard_ag_angstorm	gcc biotech diagsure nvoc-19 detection assay
9	covipath_rtprc_thermofisher	mylab	perkinelmer new coronavirus nucleic acid detection kit
10	huwel_lifesciences_quantiplus	biocard pro covid-19 rapid ag test kit (triviron healthcare pvt. ltd)	innodx_innodetect_kit
11	genetix_covisure	labcare diagnostics ltd - covid-19 antigen lateral test device	id_now_abbott
12	labgun	oscar_ag	sansure_biotech
13	seegene	alpine covid-19 antigen rapid test kit (alpine biomedical pvt. ltd., india)	aptima_hologic
14	genes2me viral detect-ii	cosara_diagnostics_saragene	pcl covid-19 rapid fia (pcl inc.)
15	truenat	oscar corona rapid test kit (oscar medicare pvt. ltd.)	intron_biotechnology_llif
16	gcc_biotech_diagsure	bag_viroq	genepath kit
17	qline_mer_rtprc_pict_ltd	immunoquick covid-19 antigen rapid card test kit (immunoscience india pvt. ltd.)	id now covid-19
18	truenat_rdrp_confirmatory	angcard covid-19 rapid antigen test kit (angstrom biotech pvt. ltd.)	bhat_biotech_geneasy
19	quantiplus_version2	immunoquick_ag	med_achievers_genome
20	poct_q-line_rtprc_kit	sensit_ag_ubio_biotech	cancer_rop_q_sens_2019-ncov
21	abi_taqman	agq_n_ag_agappe	llc_art
22	truenat_duplex_molbio	bag_diagnostics-viroq_rt-prc_kit	jitm_skills_covid-19
23	altona	meriscreen covid-19 antigen detection test (meril diagnostics)	mbi_mebright_sars-cov-2
24	triviron_covidsure_pro	angcard covid-19 rapid antigen test kit	perkinelmer-new coronavirus nucleic acid detection kit
25	genestore_detection_expert	karwa_sars-cov-2	coviqwik_rtlamp_acrannolife_genomics
26	biomerieux_argene	abbott_binaxnow_ag_card	coviwok_wockhardt
27	neodx_covidx_mplex-4r	covid-19 antigen lateral test device	
28	genetix covisure covid-19 rt prc kit	covirat_ag_zephyr	
29	bgj_genomics	covid-19 ag respi strip	
30	cosara_diagnostics_saragene	xamin covid-19 ag rapid test device (diagnocure)	
31	hi_pcr_himedia	perkinelmer_covid19_antigen_kit	
32	covipath covid 19 rt prc kit	one step ag sidak	

Figure 3.7: Screenshot of Master list of Testing Kits Used

The custom function for cleaning testing kits used:

1. `testkit_fn`: Clean the testing kits used variable

### 3.7.1 Clean Testing Kits Used

```
# Function for cleaning the testing kits used

testkit_fn <- function(parquet_month){

  # Path
  # parquet_month <- "2020_03"

path <- paste("poojaviswam11", "shared_folder","data", "parquet_data",
sep = "/")

# Create Folder Path
parquet_folder <- here() |>
  fs::path_dir() |>
  fs::path_dir() |>
  fs::path_dir() |>
  paste(path, parquet_month, sep = "/")

df <- arrow::open_dataset(parquet_folder)

test_path <- paste("poojaviswam11", "shared_folder",
                  "data", "master_csv","test_kitlist.csv", sep = "/")

test_kitlist <- here() |>
  fs::path_dir() |>
  fs::path_dir() |>
  fs::path_dir() |>
  paste(test_path, sep = '/') |>
  read_csv("test_kitlist.csv") |>
  mutate(across(everything(), ~ str_replace_all(., '\\xa0' , " "))) |>
  set_names("rtPCR", "antigen", "others")

rtPCR_sample <- test_kitlist |>
  select(rtPCR) |>
  drop_na() |>
  pull(rtPCR) |>
  unique()

antigen_sample <- test_kitlist |>
  select(antigen) |>
```

```

drop_na() |>
pull(antigen) |>
unique()

oth_sample <- test_kitlist |>
  select(others) |>
  drop_na() |>
  pull(others) |>
  unique()

df_1 <-< as_tibble(df |>
  dplyr::select(c(icmr_id, testing_kit_used)))
df_1 <- df_1 |>
  mutate(testing_kit_used = str_trim(str_to_lower(testing_kit_used)))

df_1 <- df_1 |>
  mutate(testing_kit_status = NA_character_) |>
  mutate(testing_kit_status =
  case_when(
    (testing_kit_used %in% rtpcr_sample) ~ "rtpcr",
    (testing_kit_used %in% antigen_sample) ~ "antigen",
    (testing_kit_used %in% oth_sample) ~ "others"))

out_folder <- parquet_folder |>
str_replace("parquet_data", "testkit_cleaned_parquetdata")

out_folder |> fs::dir_create()

out_file <- paste(parquet_month, "_cleaned_testkit.parquet", sep = "")

df_1 |>
  #dplyr::select(c(icmr_id, final_test_status)) |>
  write_parquet(here(out_folder, out_file))
}

```

```
# Load Parquet Files and explore data
```

```
# Load Packages
library(tidyverse)
library(here)
```

```

library(fs)
library(arrow)
library(stringr)

testkit_fn <- function(parquet_month){

  # Path
  # parquet_month <- "2020_03"

  path <- paste("poojaviswam11", "shared_folder","data", "parquet_data",
               sep = "/")

  # Create Folder Path
  parquet_folder <- here() |>
  fs::path_dir() |>
  fs::path_dir() |>
  fs::path_dir() |>
  paste(path, parquet_month, sep = "/")

  df <- arrow::open_dataset(parquet_folder)

  test_path <- paste("poojaviswam11", "shared_folder",
                    "data", "master_csv","test_kitlist.csv", sep = "/")

  test_kitlist <- here() |>
  fs::path_dir() |>
  fs::path_dir() |>
  fs::path_dir() |>
  paste(test_path, sep = '/') |>
  read_csv("test_kitlist.csv") |>
  mutate(across(everything(), ~ str_replace_all(., '\\xa0' , " "))) |>
  set_names("rtPCR", "antigen", "others")

  rtPCR_sample <- test_kitlist |>
  select(rtPCR) |>
  drop_na() |>
  pull(rtPCR) |>
  unique()

  antigen_sample <- test_kitlist |>
  select(antigen) |>

```

```

drop_na() |>
pull(antigen) |>
unique()

oth_sample <- test_kitlist |>
  select(others) |>
  drop_na() |>
  pull(others) |>
  unique()

df_1 <-< as_tibble(df |>
  dplyr::select(c(icmr_id, testing_kit_used)))
df_1 <- df_1 |>
  mutate(testing_kit_used = str_trim(str_to_lower(testing_kit_used)))

df_1 <- df_1 |>
  mutate(testing_kit_status = NA_character_) |>
  mutate(testing_kit_status =
    case_when(
      (testing_kit_used %in% rtpcr_sample) ~ "rtpcr",
      (testing_kit_used %in% antigen_sample) ~ "antigen",
      (testing_kit_used %in% oth_sample) ~ "others"))

out_folder <- parquet_folder |>
  str_replace("parquet_data", "testkit_cleaned_parquetdata")

out_folder |> fs::dir_create()

out_file <- paste(parquet_month, "_cleaned_testkit.parquet", sep = "")

df_1 |>
  #dplyr::select(c(icmr_id, final_test_status)) |>
  write_parquet(here(out_folder, out_file))

}

folder_names <- parquet_folder |> fs::path_dir() |> dir() |> unique()

folder_names |> map(testkit_fn)

```

## 4 Organizing the Data for Analysis

Following the completion of the data cleaning and standardization processes, the next critical step in preparing the dataset for analysis involved organizing the data into manageable formats. This process aimed to combine raw, date-wise records into comprehensive daily and monthly datasets, ensuring the data could be efficiently analyzed.

To achieve this, custom R functions were employed to combine individual records from each day into a unified daily file, and subsequently, to consolidate these daily records into monthly files. The functions, `combine_daily_fn` and `combine_monthly_fn`, automated the process, significantly improving the efficiency of handling large volumes of data. Additionally, to facilitate the transition from raw `.csv` files to a more optimized format for analysis, the data was converted into the `.parquet` format using the `csv_to_parquet_fn` function. This conversion reduced the file size and improved data accessibility, enabling faster processing.

The following sections detail the methods used to combine and convert the data, as well as the tools and techniques leveraged for handling large datasets, including `Apache Spark`, `disk.frame`, and `Apache Arrow`.

### Combine the Daily and Monthly Data

We have already downloaded the whole offsets data date wise in each folders in `.csv` format from the ICMR Server([Fetch-ICMR-COVID19-Data](#)).The downloaded data is in 10,000 record per each offsets in a day.So we have written custom functions to combine the records for each day in `.csv` format and other function is to combine the daily data files to the month-wise files.

We have achieved this with the help of a few custom helper functions:

1. `combine_daily_fn`: Combines all the records of specific day to as daily data csv file
2. `combine_monthly_fn`: Combine all the daily data to a specific monthly data as a csv file

#### 1. Combine Daily Data

```
# Function for combining all the records of each day to daily data file

combine_daily_fn <- function(x){
  # x <- "/home/poojaviswam11/data/df_2022_11_13"
  # x |> dir_ls()
```

```

x_combined_daily <- x |>
  dir_ls() |>
  map_dfr(read_csv,
          col_types = cols(date_sample_collection = col_character(),
                           date_sample_received = col_character(),
                           date_of_sample_tested = col_character(),
                           ct_value_screening = col_character(),
                           ct_value_rdrp = col_character(),
                           ct_value_orf1b = col_character(),
                           entry_date = col_character(),
                           updated_on = col_character()))

file_name <- x |>
  basename() |>
  paste0('.csv')

dest_path <- x |>
  str_replace(., "data", "daily_data") |>
  dirname() |>
  paste(., file_name, sep = "/")
dest_path |>
  dirname() |>
  fs::dir_create()
x_combined_daily |> write_csv(dest_path)
x_combined_daily |> rm()
}

```

## 2. Combine the Monthly Data

```

# Function for combining all the Daily data to Monthly data

combine_monthly_fn <- function(x){

  x1 <- here("shared_folder", "daily_data", "raw_data", x) |> dir_ls()

  x_combined_monthly <- x1 |>
    future_map_dfr(read_csv,
                  col_types = cols(date_sample_collection = col_character(),
                                   date_sample_received = col_character(),
                                   date_of_sample_tested = col_character(),

```

```

        ct_value_screening = col_character(),
        ct_value_rdrp = col_character(),
        ct_value_orf1b = col_character(),
        entry_date = col_character(),
        updated_on = col_character())

dest_path <- here("monthly_data", x) |> paste0('.csv')
dest_path |> dirname() |> fs::dir_create()
x_combined_monthly |> write_csv(dest_path)
x_combined_monthly |> rm()
}

# Combine Data Functions

# Load Packages
library(tidyverse)
library(here)
library(fs)
library(furrr)
library(tictoc)

# gc()
plan(multisession, workers = 5)

# Check file paths
# x <- fs::dir_ls(here(here()) |> dirname(), 'poojavismam11', "data", "jan_2021"))
# x <- "jan_2021"

# Create Daily Function
combine_daily_fn <- function(x){
  # x <- "/home/poojavismam11/data/df_2022_11_13"
  # x |> dir_ls()

  x_combined_daily <- x |>
    dir_ls() |>
    map_dfr(read_csv,
            col_types = cols(date_sample_collection = col_character(),
                             date_sample_received = col_character(),
                             date_of_sample_tested = col_character(),
                             ct_value_screening = col_character(),
                             ct_value_rdrp = col_character(),
                             ct_value_orf1b = col_character(),
                             entry_date = col_character(),

```

```

        updated_on = col_character()))

file_name <- x |>
  basename() |>
  paste0('.csv')

dest_path <- x |>
  str_replace(., "data", "daily_data") |>
  dirname() |>
  paste(., file_name, sep = "/")
dest_path |>
  dirname() |>
  fs::dir_create()
x_combined_daily |> write_csv(dest_path)
x_combined_daily |> rm()

}

combine_monthly_fn <- function(x){

  x1 <- here("shared_folder", "daily_data", "raw_data", x) |> dir_ls()

  x_combined_monthly <- x1 |>
    future_map_dfr(read_csv,
                  col_types = cols(date_sample_collection = col_character(),
                                   date_sample_received = col_character(),
                                   date_of_sample_tested = col_character(),
                                   ct_value_screening = col_character(),
                                   ct_value_rdrp = col_character(),
                                   ct_value_orf1b = col_character(),
                                   entry_date = col_character(),
                                   updated_on = col_character()))

  dest_path <- here("monthly_data", x) |> paste0('.csv')
  dest_path |> dirname() |> fs::dir_create()
  x_combined_monthly |> write_csv(dest_path)
  x_combined_monthly |> rm()

}

```

```

# Running the functions

#### For Daily

here("data", "2023_03") |>
  dir_ls() |>
  future_map(combine_daily_fn)

here("data", "feb_2021") |>
  dir_ls() |>
  future_map(combine_daily_fn)
tic()
here("shared_folder", "data", "raw_data", "2023_04") |>
  dir_ls() |>
  future_map(combine_daily_fn)
toc()

#### For Monthly

tic()
combine_monthly_fn("jan_2021")
toc()

tic()
combine_monthly_fn("feb_2021")
toc()

tic()
combine_monthly_fn("2023_04")
toc()

```

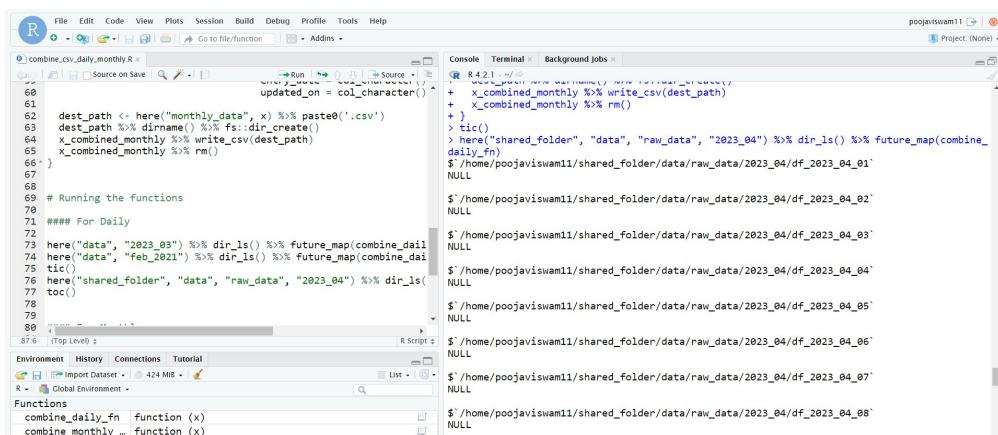


Figure 4.1: Screenshot of Running the function for Combining the Daily Data in RStudio

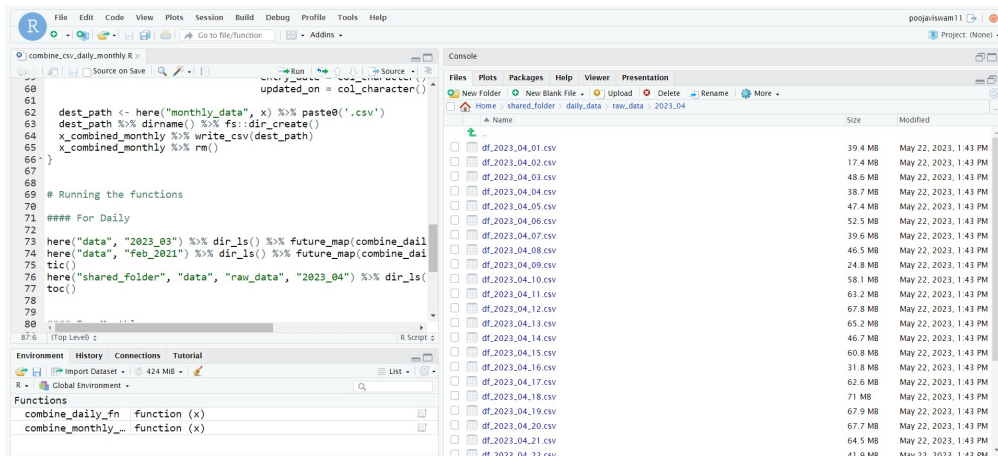


Figure 4.2: Screenshot of the output showing the Combined Daily Data Files

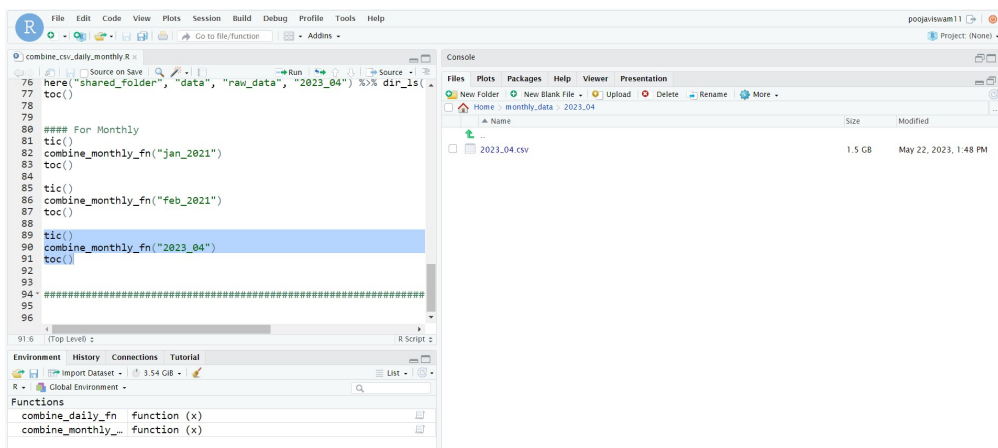


Figure 4.3: Screenshot of the output showing the Combined Monthly Data File

## Sparklyr

To analyze the big data we have started using [sparklyr](#). Apache Spark as a leading tool that is democratizing our ability to process large datasets. It is an open source cluster computing platform. sparklyr is an R package that lets you write R code to work with data in a Spark cluster. It has a dplyr interface, which means that you can write (more or less) the same dplyr-style R code, whether you are working with data on your machine or on a Spark cluster.

## Disk.frame

We have switched to use [disk.frame](#) due to the memory allocation issue. Here we have converted to disk.frame i.e. .fst format. Disk.frame supports parallel data operations using infrastructures provided by the excellent future package to take advantage of multi-core CPUs. Further, disk.frame uses state-of-the-art data storage techniques such as fast data compression, and random access to rows and columns provided by the fst package to provide superior data manipulation speeds. Here also we have faced the same memory allocation issue.

## Apache Arrow

We have started using [Apache Arrow](#) package solution for handling this large set data pre-processing. Apache Arrow is a software development platform for building high performance applications that process and transport large data sets. It is designed to improve the performance of data analysis methods, and to increase the efficiency of moving data from one system or programming language to another.

The arrow R package exposes an interface to the Arrow C++ library, enabling access to many of its features in R. It provides low-level access to the Arrow C++ library API and higher-level access through a dplyr backend and familiar R functions. The arrow package provides functionality for a wide range of data analysis tasks. Read and write Parquet files, an efficient and widely used columnar format.

## Parquet file format

Parquet is an open source file format built to handle flat columnar storage data formats. Parquet operates well with complex data in large volumes. It is known for its both performant data compression and its ability to handle a wide variety of encoding types.

Apache Arrow package uses parquet format and we have converted .csv to .parquet format which is comparatively smaller in size. We have written a function to convert csv files to .parquet format.

### 4.1 Convert the file format

We have achieved this with the help of custom helper functions:

1. `csv_to_parquet_fn`: Converts .csv to .parquet file format
2. `bind_csv_fn`: Binds the rows of the csv files

```
# Function for converting the .csv format to .parquet format

csv_to_parquet_fn <- function(x){

  # x <- "2021_02"

  folder_paths <- root_dir |>
  paste(x, sep = "/") #|>
  #fs::dir_ls()

# Function for binding rows of csv files

bind_csv_fn <- function(x){
  # x <- folder_paths[[3]]
```

```

x |>
  str_replace_all("monthly_data", "parquet_data") |>
  path_dir() |>
  unique() |>
  dir_create()

dest_file <- x |>
  path_dir() |>
  unique() |>
  str_replace_all("monthly_data", "parquet_data") |>
  paste(x |> fs::path_file(), sep = "/") |>
  paste(., ".parquet", sep = "")

X <- x |>
  dir_ls() |>
  read_csv() |>
  bind_rows()

X |> arrow::write_parquet(sink = dest_file)
}

a <- folder_paths |>
  map(bind_csv_fn)

}

```

```

# Script for Converting .csv files to .parquet format

# Load Packages
library("data.table")
library("arrow")
library("tidyverse")
library("here")
library("fs")
# Set the root directory

root_dir <- here() |>
  fs::path_dir() |>
# fs::path_dir() |>

```

```

# fs::path_dir() |>
# paste("poojavism11", "shared_folder","data", "raw_data", sep = "/")
paste("poojavism11", "shared_folder","data", "monthly_data", sep = "/")

csv_to_parquet_fn <- function(x){

  # x <- "2021_02"

  folder_paths <- root_dir |>
    paste(x, sep = "/") #|>
    #fs::dir_ls()

  bind_csv_fn <- function(x){
    # x <- folder_paths[[3]]

    x |>
      str_replace_all("monthly_data", "parquet_data") |>
      path_dir() |>
      unique() |>
      dir_create()

    dest_file <- x |>
      path_dir() |>
      unique() |>
      str_replace_all("monthly_data", "parquet_data") |>
      paste(x |> fs::path_file(), sep = "/") |>
      paste(., ".parquet", sep = "")

    X <- x |>
      dir_ls() |>
      read_csv() |>
      bind_rows()

    X |> arrow::write_parquet(sink = dest_file)
  }

  a <- folder_paths |>
    map(bind_csv_fn)
#

```

```

}

x <- "2021_01"

library(furrr)
plan(multisession, workers = 4)
furrr_options(seed = TRUE)

root_dir |>
  fs::dir_ls() |>
  fs::path_file() |>
  as_tibble() |>
  slice(7:7) |>
  pull(value) |>
  map(csv_to_parquet_fn)

```

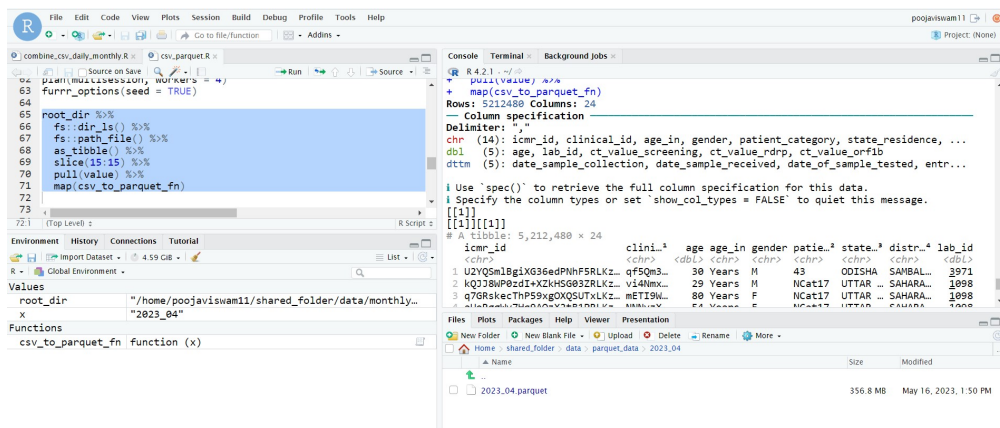


Figure 4.4: Screenshot of converting the .csv to .parquet file format

## 4.2 ICMR COVID-19 Downloaded Data Information

	Format	Size
<b>Raw Data</b>	.csv	254 GB
<b>Parquet Data</b>	.parquet	63 GB

# 5 Results

## 5.1 National-Level

The testing data analyzed spans from **March 2020** to **May 2023**, encompassing results from districts nationwide. As of the last date of data retrieval from the ICMR data portal (01-05-2023), a total of 901,303,947 COVID-19 tests have been conducted nationwide. The distribution of test results is presented in **Table 1**.

The table above shows that:

- **Positive Cases:** 48,602,943 tests (5.39%) were positive.
- **Negative Cases:** 848,334,910 tests (94.12%) were negative.
- **No Result:** 4,366,094 tests (0.48%) had no result.

The overall test positivity rate is **5.39%**.

### 5.1.1 Age Distribution

The following table summarizes positive cases nationally by age group:

### 5.1.2 Monthly Average Tests

As seen in the figure below, the timeline of positive cases across months from March 2020 to May 2023 illustrates three distinct epidemic peaks: the first in September 2020, the second and most devastating peak in April and May of 2021, and the third in January 2022.

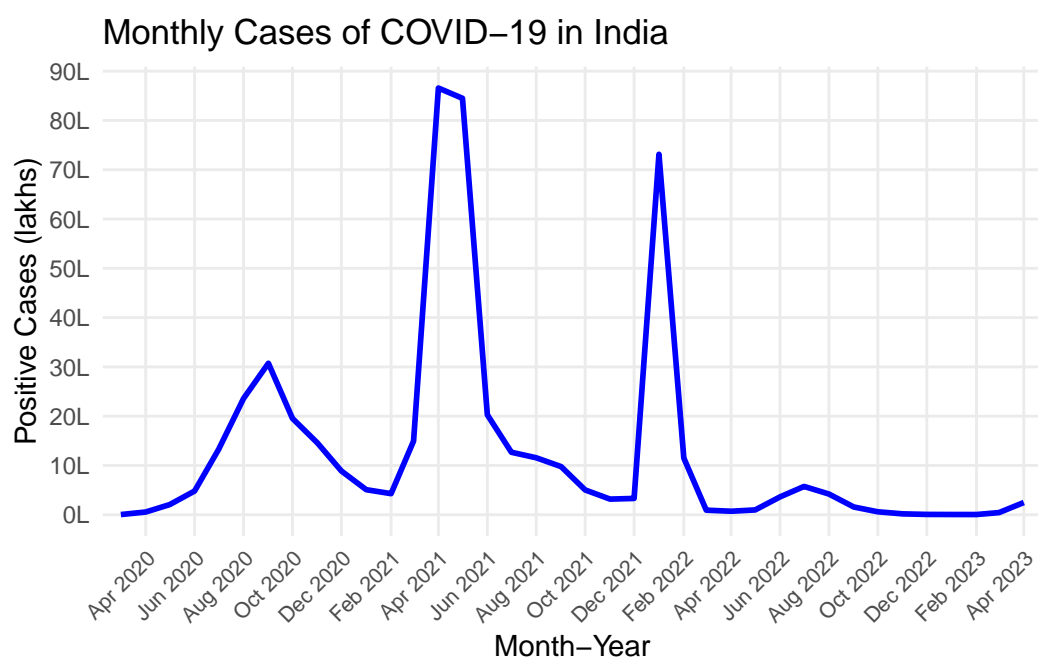
Table 1: Distribution of COVID-19 Test Results  
National-Level Summary

Test Result	Number of Tests	Percentage (%)
Positive	48,602,943	5.39
Negative	848,334,910	94.12
No Result	4,366,094	0.48
<b>Overall</b>	<b>901,303,947</b>	<b>100.00</b>

Table 2: Positive Cases by Age Group  
National-Level Summary

Age Group	Positive Cases	Percentage (%)
Infant	61,928	0.13
Under-5	553,599	1.14
5-10	963,444	1.99
10-20	4,057,009	8.36
20-30	9,851,941	20.30
30-40	10,586,775	21.82
40-50	8,204,968	16.91
50-60	6,701,402	13.81
60-70	4,579,604	9.44
70-80	2,191,490	4.52
80+	776,675	1.60
<b>Overall</b>	<b>48,602,705</b>	<b>100.00</b>

Note: 5354 positive cases with missing age group information were excluded.



## 5.2 Symptoms among Positive Cases

The analysis of symptoms among positive cases reveals interesting patterns. A total of 6,407,840 individuals reported at least one symptom. Notably, fever was the most frequently reported symptom, present in 2,366,817 individuals, which accounts for 487% of total cases and 36.94% of all reported symptoms.

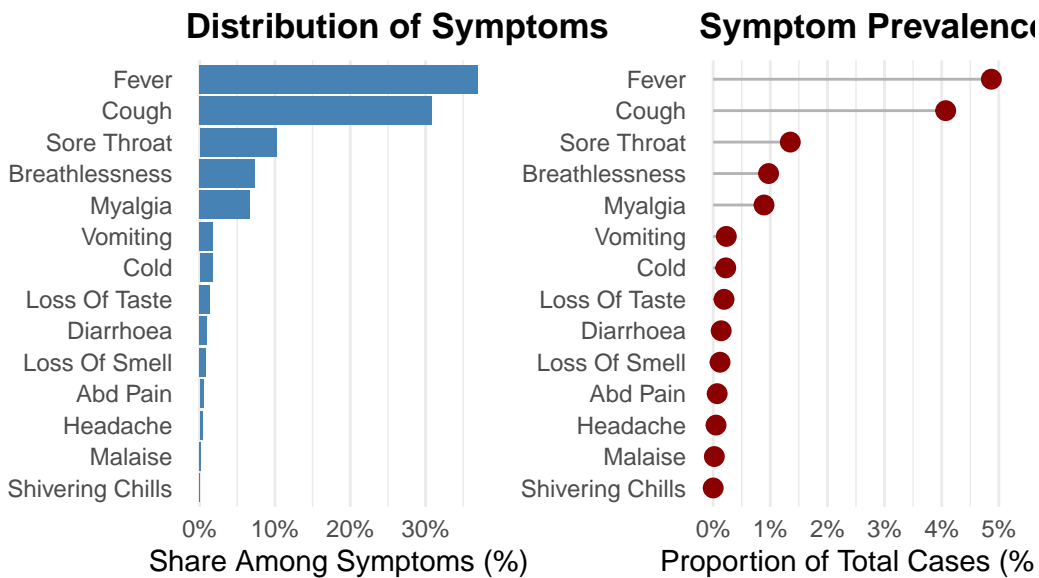
Other prominent symptoms included:

- **Cough:** Reported by 1,976,200 individuals (407% of total cases).
- **Sore Throat:** Observed in 654,954 cases (135%).
- **Breathlessness:** Affected 472,303 individuals, making up 97% of positive cases.

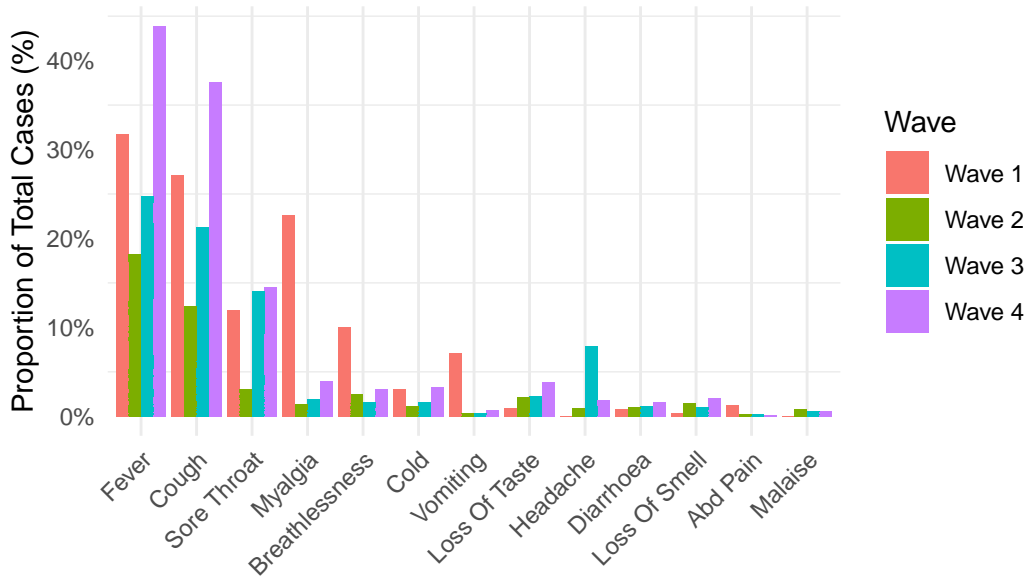
Table 3: Distribution of Symptoms among Positive Cases  
National-Level Summary

Symptoms	Count	Proportion of Total Cases (%)	Share Among Symptoms (%)
Fever	2,366,817	4.87	36.94
Breathlessness	472,303	0.97	7.37
Myalgia	431,582	0.89	6.74
Headache	25,358	0.05	0.40
Abd Pain	33,785	0.07	0.53
Cold	109,211	0.22	1.70
Malaise	12,043	0.02	0.19
Diarrhoea	66,270	0.14	1.03
Sore Throat	654,954	1.35	10.22
Vomiting	112,080	0.23	1.75
Cough	1,976,200	4.07	30.84
Loss Of Smell	56,489	0.12	0.88
Loss Of Taste	90,270	0.19	1.41
Shivering Chills	478	0.00	0.01

### Symptom Distribution and Prevalence among Positive Cas



**Percentage of Symptoms Among COVID-19 Positive Cases During Different Waves in India**



### 5.3 State Level

### 5.4 By Age Group

Table: Total COVID-19 Tests by State  
Cumulative data as of May 2023

State Name	Total Tests	Proportion (%)	Positive Cases	Test Positivity
Uttar Pradesh	139,059,432	15.43	2706090	1.95
Bihar	97,238,584	10.79	970035	1.00
Maharashtra	84,621,051	9.39	8778216	10.37
Karnataka	70,976,569	7.87	4400774	6.20
Tamil Nadu	61,586,389	6.83	3334116	5.41
Kerala	49,133,392	5.45	7190273	14.63
Delhi	40,900,640	4.54	2233071	5.46
Gujarat	39,717,559	4.41	1454576	3.66
Odisha	34,264,695	3.80	1497685	4.37
Andhra Pradesh	33,389,588	3.70	2774484	8.31
Telangana	28,485,540	3.16	170992	0.60
West Bengal	26,632,036	2.95	2240916	8.41
Madhya Pradesh	25,956,027	2.88	1350262	5.20
Punjab	21,261,449	2.36	928339	4.37
Jammu & Kashmir	19,805,103	2.20	494185	2.50
Jharkhand	19,156,353	2.13	550976	2.88
Rajasthan	18,849,115	2.09	1567340	8.32
Haryana	18,769,893	2.08	1401774	7.47
Chhattisgarh	17,890,649	1.98	1234020	6.90
Assam	16,896,330	1.87	815063	4.82
Uttarakhand	10,251,691	1.14	465245	4.54
NA	6,491,717	0.72	392575	6.05
Himachal Pradesh	4,331,451	0.48	316464	7.31
Tripura	2,611,162	0.29	99623	3.82
Goa	2,161,012	0.24	276883	12.81
Mizoram	1,698,288	0.19	224764	13.23
Puducherry	1,662,245	0.18	193694	11.65
Manipur	1,603,535	0.18	149828	9.34
Chandigarh	1,392,081	0.15	121167	8.70
Meghalaya	1,335,311	0.15	100727	7.54
Arunachal Pradesh	764,925	0.08	56074	7.33
Andaman & Nicobar	530,767	0.06	12483	2.35
Ladakh	441,826	0.05	29197	6.61
Dadra & Nagar Have	414,426	0.05	6820	1.65
Lakshadweep	307,904	0.03	8370	2.72
Nagaland	303,896	0.03	28751	9.46
Daman & Diu	217,179	0.02	4343	2.00
Sikkim	194,137	0.02	22510	11.59
<b>Total</b>	<b>901,303,947</b>	<b>100.00</b>	<b>48602705</b>	<b>5.39</b>

Note: Data includes only non-missing values. 6,491,717 rows with missing state details were excluded.

Table: Positive Cases by State and Age Group  
 COVID-19 data categorized by age groups and states

State Name	Infant	Under-5	10-20	20-30	30-40	40-50	5-10	50-60	60-70	70-80	80+	Total Cases
Andaman And Nicobar Islands	8	161	958	2,798	3,054	2,228	407	1,679	791	289	104	12,477
Andhra Pradesh	186	19,925	254,775	606,210	604,418	491,976	54,666	379,879	236,536	102,184	21,954	2,772,709
Assam	184	843	15,660	187,017	13,971	7,856	1,679	4,656	1,612	597	265	56,070
Arunachal Pradesh	3,180	9,964	73,044	183,241	139,709	139,709	16,599	107,937	58,852	26,241	9,019	814,803
Bihar	201	7,590	92,851	238,507	216,559	159,858	15,640	119,771	78,433	30,065	8,803	968,278
Chhattisgarh	368	1,131	8,803	27,158	28,085	19,471	2,058	16,700	9,658	5,219	2,425	121,076
Chhatisgarh	1,576	14,380	124,658	290,287	296,489	209,888	27,538	155,021	85,706	30,965	9,494	1,246,002
Dadra And Nagar Haveli	10	47	533	1,797	1,858	1,300	97	717	298	129	32	6,818
Daman And Diu	8	51	301	1,052	1,080	810	64	542	271	126	37	4,342
Delhi	4,693	17,509	145,929	468,543	521,301	369,519	32,405	323,810	209,861	100,098	38,324	2,231,992
Goa	906	4,390	23,599	59,977	57,346	44,655	6,809	36,841	24,530	12,724	4,101	275,878
Gujarat	1,839	7,594	92,435	284,828	314,371	248,868	16,105	223,335	163,351	74,558	26,095	1,453,379
Haryana	2,327	10,759	99,851	308,114	359,375	227,642	21,935	170,928	122,076	58,798	19,394	1,401,199
Himachal Pradesh	755	3,052	32,430	66,126	68,249	54,041	5,922	39,971	25,968	13,921	5,990	316,425
India	71	271	2,534	10,105	9,199	5,533	419	4,255	1,390	1,390	410	37,207
Jammu And Kashmir	531	3,334	35,688	112,212	123,406	81,911	6,687	61,243	39,033	17,601	6,483	488,129
Jharkhand	609	3,577	45,467	124,709	130,922	97,568	7,196	76,096	41,996	17,453	5,186	550,779
Karnataka	7,348	43,054	432,303	978,909	1,006,831	724,118	85,585	519,790	368,691	172,832	59,092	4,398,553
Kerala	1,785	179,878	764,319	1,275,968	1,291,791	1,164,801	245,936	992,982	714,199	367,660	162,774	7,162,093
Ladakh	39	304	3,453	8,489	7,120	4,422	567	2,504	1,276	649	368	29,191
Lakshadweep	NA	71	1,289	1,754	1,849	1,351	258	904	563	260	70	8,369
Madhya Pradesh	1,776	10,194	100,115	298,312	301,852	228,711	19,655	197,298	123,977	50,729	17,417	1,350,036
Maharashtra	11,013	87,219	643,654	1,613,340	1,976,134	1,532,174	162,479	1,234,350	885,020	455,948	160,016	8,761,347
Manipur	555	3,040	17,242	27,818	32,162	22,872	5,829	17,726	12,654	7,003	2,887	149,788
Meghalaya	539	2,299	15,589	25,279	22,110	13,588	3,908	9,455	4,745	2,159	931	100,602
Mizoram	3,327	13,462	48,483	44,671	42,470	28,669	19,557	19,360	13,121	7,244	3,352	243,716
Nagaland	46	260	2,505	7,921	8,626	4,515	406	2,569	1,105	519	276	28,748
Odisha	3,135	13,465	136,070	328,352	333,019	260,579	29,745	205,197	122,228	49,079	16,756	1,497,625
Puducherry	464	2,553	18,072	39,280	40,031	35,158	4,250	27,856	16,477	7,141	2,378	193,660
Punjab	1,145	5,038	58,902	184,548	209,656	155,993	9,766	137,347	100,276	49,576	15,937	928,184
Rajasthan	1,847	10,842	137,404	365,295	357,740	254,819	23,288	201,858	136,777	58,372	18,611	1,566,853
Sikkim	94	252	2,328	5,911	5,502	3,404	429	2,330	1,276	683	254	22,463
Tamil Nadu	4,357	28,927	247,831	708,658	755,038	633,861	52,924	569,113	395,635	193,281	64,567	3,654,192
Telangana	278	1,918	9,036	33,282	41,940	30,136	2,977	25,966	15,941	7,155	2,016	170,645
Tripura	162	835	9,510	21,586	24,626	17,067	1,774	12,432	7,192	3,084	1,269	99,537
Uttar Pradesh	1,439	21,227	196,723	599,844	632,129	453,489	41,002	370,886	244,023	106,313	33,851	2,700,926
Uttarakhand	729	2,776	38,878	106,533	109,385	79,083	5,688	59,934	37,509	18,237	6,028	464,780
West Bengal	4,398	21,407	130,700	371,091	453,840	393,325	31,195	368,164	274,927	141,208	49,709	2,239,964

Note: Data includes only non-missing age groups.

## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

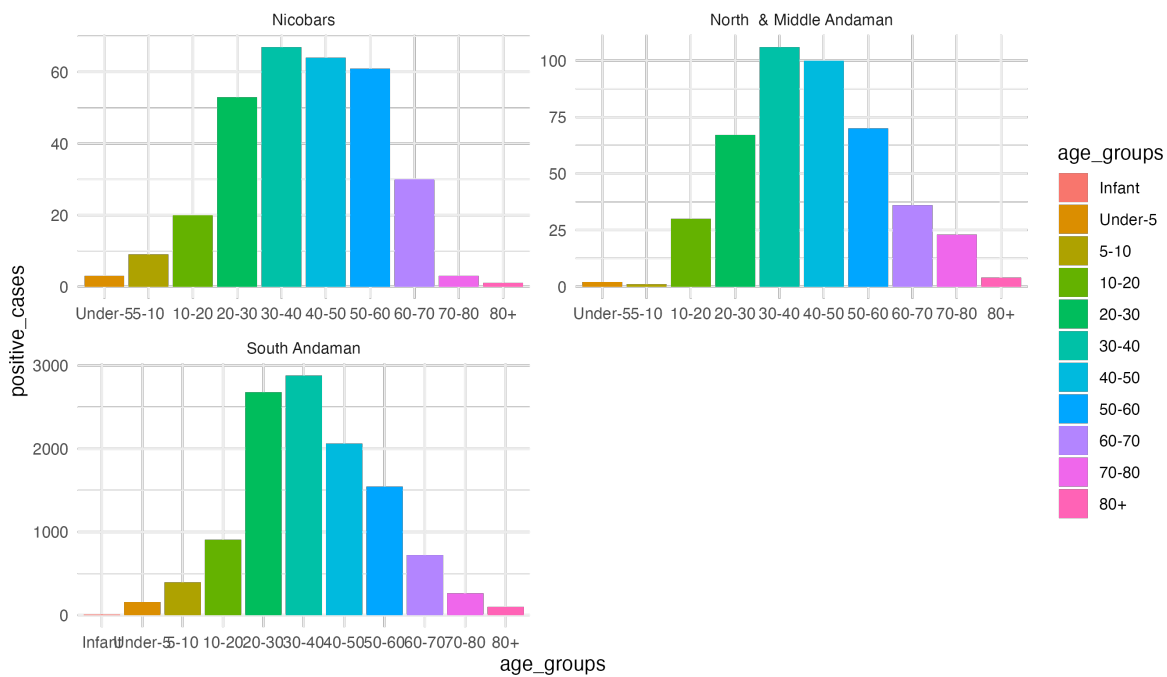
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Nicobars	NA	3	9	20	53	67	64	61	30	3	1	311
North & Middle Andaman	NA	2	1	30	67	106	100	70	36	23	4	439
South Andaman	8	156	397	908	2,678	2,881	2,064	1,548	725	263	99	11,727

Note: Data includes only non-missing age groups.

### 5.4.1 Andaman And Nicobar Islands

#### 5.4.1.1 District Wise Summary Table

#### 5.4.1.2 District wise Age Groups



## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

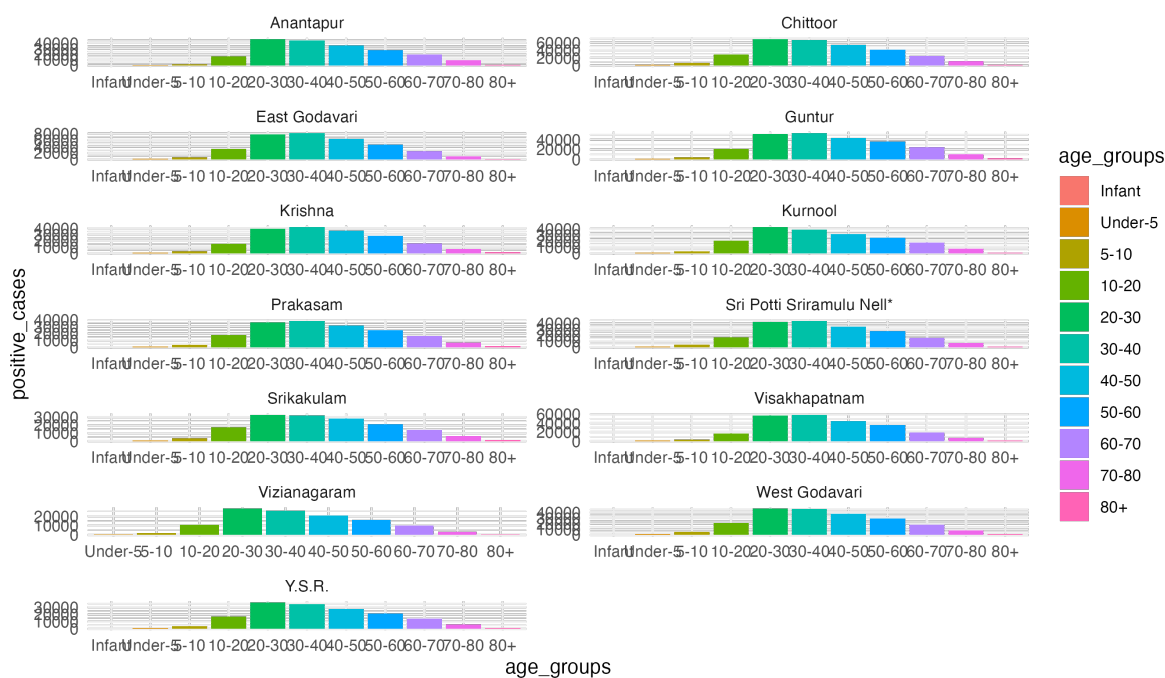
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Anantapur	32	792	2,644	16,795	46,503	44,324	35,974	27,422	19,520	9,591	1,841	205,438
Chittoor	48	2,801	7,209	29,207	68,150	66,163	54,344	41,425	24,906	11,199	2,766	308,218
East Godavari	16	2,557	6,738	32,392	78,061	81,532	64,577	46,355	25,644	10,307	1,856	350,035
Guntur	11	1,924	4,645	21,433	51,146	52,834	43,278	36,023	24,576	10,692	2,575	249,137
Krishna	12	1,451	3,332	15,419	39,177	42,025	36,457	28,099	16,266	6,833	1,886	190,957
Kurnool	15	1,489	3,521	20,986	42,758	38,128	31,539	25,277	17,539	7,724	1,571	190,547
Prakasam	4	1,177	3,868	17,705	36,262	38,221	31,976	24,718	16,662	6,871	1,500	178,964
Sri Potti Sriramulu Nelli*	23	1,431	3,925	16,992	42,642	44,158	34,768	26,944	16,278	7,098	1,574	195,833
Srikakulam	2	1,027	3,622	17,500	33,058	32,360	28,569	21,362	14,027	6,236	1,176	158,939
Visakhapatnam	9	1,356	3,436	17,560	57,199	58,385	44,257	35,441	18,917	7,695	1,619	245,874
Vizianagaram	NA	673	2,155	10,783	27,973	25,992	20,850	15,874	9,765	3,612	611	118,288
West Godavari	5	1,845	5,616	20,979	46,643	46,137	38,039	29,374	17,998	8,383	1,776	216,795
Y.S.R.	9	1,402	3,955	17,024	36,638	34,159	27,348	21,565	14,438	5,943	1,203	163,684

Note: Data includes only non-missing age groups.

### 5.4.2 Andhra Pradesh

#### 5.4.2.1 District Wise Summary Table

#### 5.4.2.2 District wise Age Groups



## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Anjaw	NA	20	37	191	321	218	125	43	11	3	1	970
Changlang	20	62	127	669	1,218	1,062	676	450	146	62	23	4,515
East Kameng	4	30	44	333	480	293	197	101	30	4	1	1,517
East Siang	14	65	86	567	983	1,004	592	433	154	61	24	3,983
Kamle	1	4	11	34	50	47	24	13	3	3	NA	190
Kra Daadi	NA	1	2	18	46	33	18	3	NA	1	1	123
Kurung Kumey	NA	6	17	46	50	58	47	23	20	3	NA	270
Lepa Rada	14	46	58	149	270	286	160	98	53	17	4	1,155
Lohit	4	31	106	465	683	707	390	270	84	33	22	2,795
Longding	NA	1	5	70	143	114	53	31	22	4	1	444
Lower Dibang Valley	6	41	123	536	685	705	502	318	98	46	26	3,086
Lower Siang	3	9	14	104	371	404	204	81	19	8	5	1,222
Lower Subansiri	9	61	143	553	725	722	441	262	75	33	9	3,033
Namsai	8	16	78	526	960	761	520	301	109	60	16	3,355
Pakke Kessang	NA	6	20	81	140	118	73	32	18	3	3	494
Papum Pare	73	241	413	2,384	4,930	3,991	1,851	1,085	387	112	66	15,533
Shi Yomi	NA	NA	1	29	31	37	23	21	4	4	3	153
Siang	4	7	12	57	112	129	91	53	18	10	4	497
Tawang	2	54	92	435	736	699	373	253	84	33	22	2,783
Tirap	2	7	24	210	334	365	208	141	29	11	2	1,333
Upper Dibang Valley	2	12	29	121	118	120	74	33	14	5	NA	528
Upper Siang	NA	18	20	112	239	272	177	106	41	11	6	1,002
Upper Subansiri	6	18	47	229	361	238	167	72	37	10	2	1,187
West Kameng	7	56	124	527	1,042	1,037	579	261	98	36	18	3,785
West Siang	5	31	46	301	632	551	291	172	58	24	6	2,117

Note: Data includes only non-missing age groups.

### 5.4.3 Arunachal Pradesh

#### 5.4.3.1 District Wise Summary Table

#### 5.4.3.2 District wise Age Groups



Positive Cases by District and Age Group  
COVID-19 data categorized by age groups and district

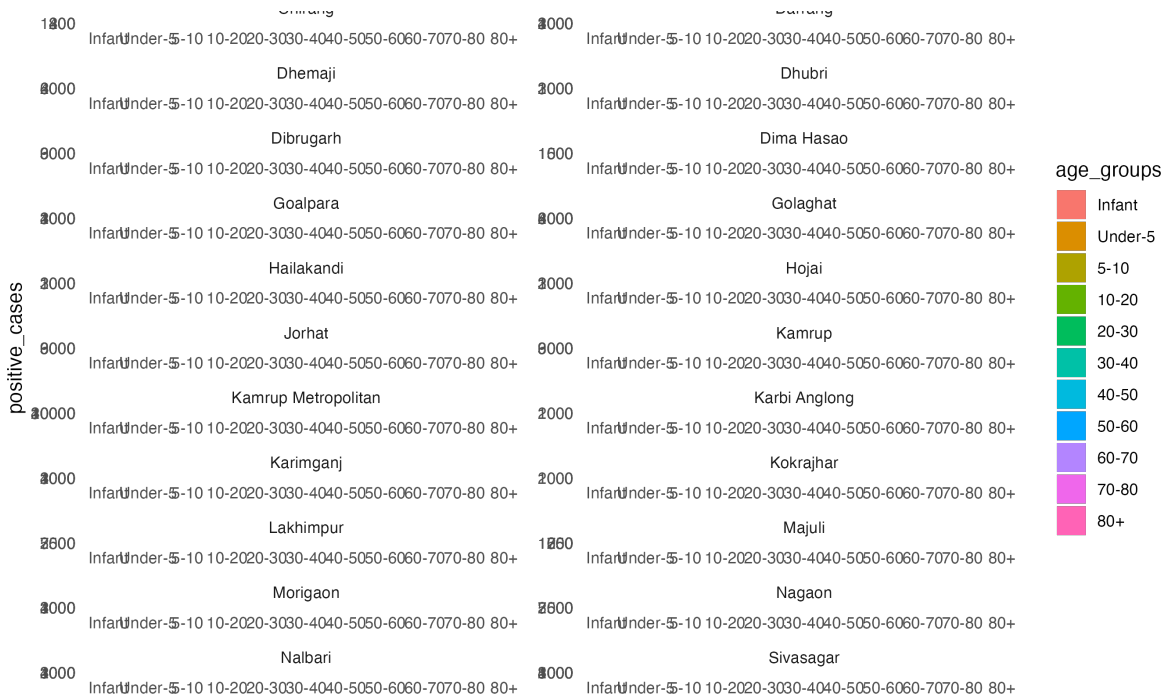
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Baksa	66	139	329	1,260	2,448	2,348	1,931	1,451	631	245	90	10,938
Barpeta	108	245	479	2,118	4,903	4,605	3,828	3,046	1,736	781	269	22,118
Biswanath	121	308	588	2,047	4,031	3,732	2,953	2,066	1,230	545	159	17,780
Bongaigaon	40	103	159	1,058	2,630	2,596	1,993	1,862	873	397	140	11,851
Cachar	137	444	747	3,341	9,985	9,704	7,715	5,700	3,451	1,612	492	43,328
Charaideo	69	105	197	885	2,395	2,177	1,786	1,420	540	217	77	9,868
Chirang	6	42	86	494	1,238	1,258	901	704	231	98	19	5,077
Darrang	77	301	467	2,031	4,018	3,720	3,064	2,279	1,333	548	181	18,019
Dhemaji	85	277	588	2,890	6,506	5,835	4,350	2,793	1,133	371	200	25,028
Dhubri	26	103	185	1,519	3,338	3,109	2,547	1,977	932	338	101	14,175
Dibrugarh	194	629	963	3,622	10,756	11,285	8,427	6,213	3,400	1,533	495	47,517
Dima Hasao	12	67	117	773	1,743	1,564	1,087	760	276	129	35	6,563
Goalpara	86	256	458	2,029	4,090	3,333	2,690	2,016	1,078	435	172	16,643
Golaghat	108	398	783	3,183	7,849	7,413	5,957	4,130	2,015	893	291	33,020
Hailakandi	37	122	194	1,082	3,107	2,418	1,994	1,240	615	209	76	11,094
Hojai	24	136	196	1,152	3,042	2,851	2,310	1,807	848	382	132	12,880
Jorhat	169	464	790	3,588	10,177	10,979	8,346	6,781	4,020	2,127	841	48,282
Kamrup	134	560	849	3,588	10,694	9,496	7,377	5,140	2,759	1,161	358	42,116
Kamrup Metropolitan	504	2,056	3,324	13,817	38,985	42,299	30,390	25,835	15,632	7,495	2,523	182,860
Karbi Anglong	19	75	149	935	2,735	2,588	1,703	1,309	488	193	61	10,255
Karimganj	99	133	242	1,459	4,003	3,369	2,391	2,046	1,008	480	168	15,398
Kokrajhar	21	116	153	988	2,635	2,574	1,775	1,340	537	219	51	10,409
Lakhimpur	292	602	989	3,898	8,645	8,006	6,433	4,765	2,392	1,036	391	37,449
Majuli	31	70	86	545	1,229	1,028	773	671	234	102	37	4,806
Morigaon	66	306	536	2,077	4,025	3,562	2,874	1,969	1,114	395	131	17,055
Nagaon	82	480	867	3,995	8,690	7,962	6,510	4,990	2,844	1,069	374	37,863
Nalbari	162	344	464	1,746	4,132	4,511	3,664	3,167	1,835	801	301	21,127
Sivasagar	164	353	472	1,852	4,506	4,789	3,761	3,054	1,733	789	275	21,748
Sonitpur	148	462	725	2,477	6,300	6,535	4,686	3,276	2,106	900	312	27,927
South Salmara Mancachar	5	21	32	300	756	655	536	290	112	45	7	2,759
Tinsukia	68	183	258	1,468	5,671	5,248	3,870	3,026	1,390	575	209	21,966
Udalguri	18	55	101	632	1,297	1,367	888	667	271	101	38	5,435
West Karbi Anglong	2	9	26	195	458	325	199	147	55	20	13	1,449

Note: Data includes only non-missing age groups.

## 5.4.4 Assam

### 5.4.4.1 District Wise Summary Table

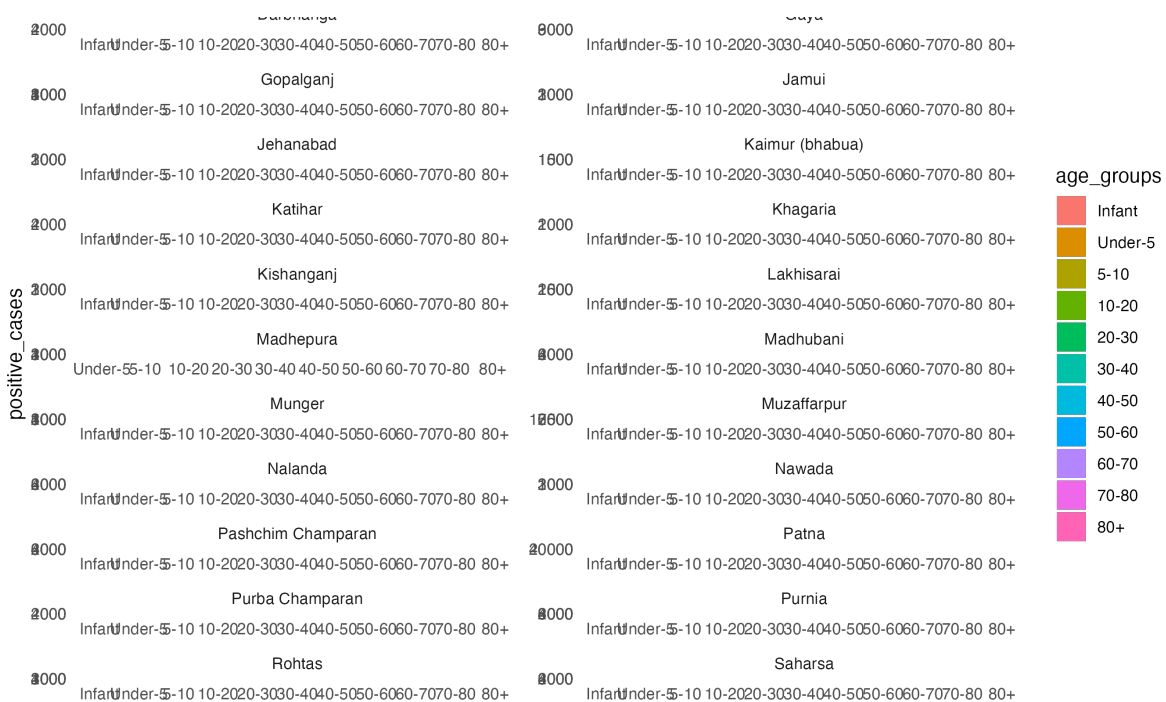
#### 5.4.4.2 District wise Age Groups



## 5.4.5 Bihar

### 5.4.5.1 District Wise Summary Table

#### 5.4.5.2 District wise Age Groups



Positive Cases by District and Age Group  
COVID-19 data categorized by age groups and district

District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Araria	2	132	273	1,790	4,639	3,826	2,553	1,777	1,149	338	91	16,570
Arwal	1	70	198	1,221	2,087	1,856	1,344	919	652	208	66	8,622
Aurangabad	3	154	325	2,375	6,367	5,221	3,680	2,396	1,378	526	161	22,586
Banka	NA	64	144	1,157	2,561	2,143	1,527	1,015	702	216	55	9,584
Begusarai	4	300	710	4,129	9,020	7,638	5,820	4,273	2,381	926	207	35,408
Bhagalpur	9	344	659	3,109	8,680	8,414	6,546	4,903	3,087	1,164	330	37,245
Bhojpur	6	78	178	1,276	3,200	2,670	1,974	1,478	902	407	99	12,268
Buxar	5	134	250	1,291	2,689	2,420	1,686	1,235	880	348	83	11,021
Darbhanga	14	154	295	1,853	5,434	5,326	3,787	3,028	2,302	858	233	23,284
Gaya	7	244	604	4,205	10,942	8,789	6,095	4,158	2,538	945	286	38,813
Gopalganj	3	133	341	2,185	4,860	4,241	3,021	1,976	1,257	418	104	18,539
Jamui	3	102	192	1,286	3,052	2,816	1,994	1,222	748	263	66	11,744
Jehanabad	6	135	238	1,630	3,360	2,671	1,990	1,482	966	378	110	12,966
Kaimur (bhabua)	4	47	127	796	1,740	1,439	948	699	584	206	57	6,647
Katihar	4	264	444	2,173	5,448	4,967	3,550	2,507	1,535	542	149	21,583
Khagaria	1	134	282	1,336	2,814	2,357	1,837	1,259	874	252	59	11,205
Kishanganj	5	114	223	1,260	3,574	3,033	1,977	1,360	781	235	67	12,629
Lakhisarai	2	65	149	878	2,261	2,151	1,593	1,023	755	220	55	9,152
Madhepura	NA	219	433	2,289	4,752	3,881	2,641	1,717	1,273	370	121	17,696
Madhubani	9	166	382	2,377	6,137	5,383	3,634	2,460	1,899	604	191	23,242
Munger	8	148	367	2,026	4,815	4,070	3,099	2,427	1,528	597	163	19,248
Muzaffarpur	6	274	568	3,281	10,184	9,452	6,566	5,114	3,123	1,229	331	40,128
Nalanda	2	232	413	2,593	7,194	6,201	4,538	3,328	2,052	720	194	27,467
Nawada	1	78	189	1,119	2,980	2,351	1,778	1,176	810	271	91	10,844
Pashchim Champaran	1	134	386	2,443	6,408	5,800	3,936	2,645	1,563	465	124	23,905
Patna	54	1,802	3,191	17,097	54,635	55,684	43,667	38,470	25,262	11,617	3,700	255,179
Purba Champaran	3	148	334	2,492	5,609	4,912	3,722	2,469	1,690	602	172	22,153
Purnia	2	441	777	3,350	7,960	7,109	5,399	3,658	2,456	743	226	32,121
Rohtas	5	131	286	1,689	4,378	4,074	2,922	2,186	1,464	478	164	17,777
Saharsa	1	145	380	2,604	5,718	4,855	3,473	2,124	1,414	446	136	21,296
Samastipur	4	253	588	3,517	6,563	5,719	4,211	3,049	1,998	621	172	26,695
Saran	7	169	359	2,675	6,833	6,114	4,578	3,072	2,054	813	223	26,897
Sheikhpura	2	51	138	1,022	2,156	1,869	1,346	848	672	209	60	8,373
Sheohar	1	53	132	740	1,068	979	761	480	353	124	27	4,718
Sitamarhi	3	97	187	1,276	2,823	2,567	1,862	1,367	1,048	305	64	11,599
Siwan	6	114	213	1,724	4,818	4,413	2,995	1,924	1,365	440	110	18,122
Supaul	1	158	369	2,337	5,139	3,987	3,026	2,027	1,342	390	109	18,885
Vaishali	6	109	316	2,250	5,609	5,161	3,782	2,520	1,596	571	147	22,067

Note: Data includes only non-missing age groups.

## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

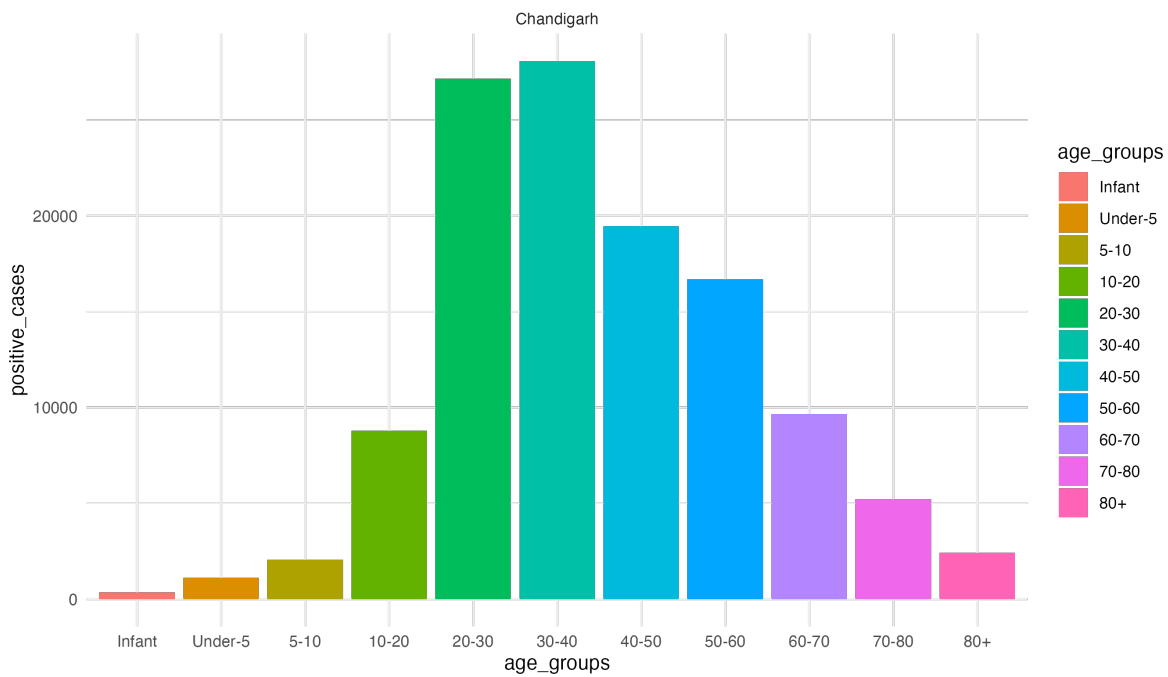
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Chandigarh	368	1,131	2,058	8,803	27,158	28,085	19,471	16,700	9,658	5,219	2,425	121,076

Note: Data includes only non-missing age groups.

### 5.4.6 Chandigarh

#### 5.4.6.1 District Wise Summary Table

#### 5.4.6.2 District wise Age Groups



## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

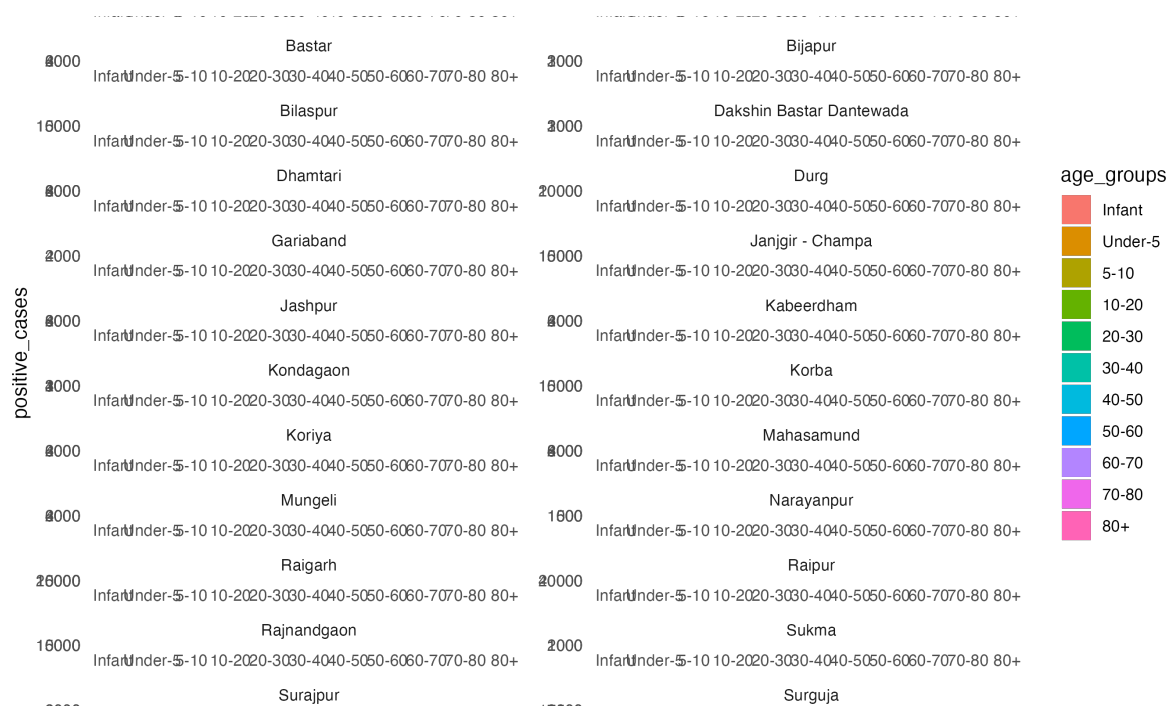
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Balod	72	366	743	3,126	7,242	6,927	5,559	4,165	2,269	825	224	31,518
Baloda Bazar	73	659	1,187	5,370	11,626	11,190	8,527	5,460	3,188	1,104	328	48,712
Balrampur	65	566	868	3,287	6,032	5,576	3,416	1,950	1,114	337	121	23,332
Bametara	25	307	641	2,860	5,348	5,149	3,943	2,731	1,606	568	174	23,352
Bastar	57	359	581	2,867	7,270	6,788	4,149	2,806	1,406	446	109	26,838
Bijapur	64	155	270	1,431	3,059	3,012	1,433	733	243	56	15	10,471
Bilaspur	101	925	1,645	6,562	17,823	18,731	13,708	10,962	6,482	2,523	827	80,289
Dakshin Bastar Dantewada	31	169	311	1,309	3,413	3,605	2,232	1,236	521	170	47	13,044
Dhamtari	50	424	770	3,962	7,927	7,999	5,951	4,367	2,715	1,061	302	35,528
Durg	126	1,044	2,076	9,589	26,413	27,928	23,632	22,824	10,964	4,828	1,711	131,135
Gariaband	33	310	623	2,458	5,061	5,319	3,533	2,293	1,193	359	76	21,258
Janjgir - Champa	47	714	1,520	7,174	15,016	15,959	10,947	7,544	4,614	1,577	452	65,564
Jashpur	44	456	915	4,299	7,690	7,925	5,288	3,448	2,242	644	175	33,126
Kabeerdham	42	410	822	3,491	6,540	6,032	4,075	2,553	1,406	400	124	25,895
Kondagaon	3	110	284	2,169	4,374	4,328	2,674	1,621	872	259	73	16,767
Korba	60	895	1,745	6,664	14,923	16,668	10,688	8,551	3,948	1,250	384	65,776
Koriya	24	345	755	3,818	7,536	7,251	5,066	3,675	1,793	513	168	30,944
Mahasamund	21	378	787	3,546	8,050	8,234	6,120	4,301	2,577	907	229	35,150
Mungeli	15	197	575	3,929	7,424	6,974	4,768	2,952	2,034	628	197	29,693
Narayanpur	3	65	117	885	1,586	1,347	640	324	188	45	7	5,207
Raigarh	81	977	1,763	7,721	17,838	19,089	13,418	8,945	4,974	1,875	512	77,193
Raipur	273	2,118	3,692	15,599	50,393	54,097	38,114	29,994	17,323	6,816	2,076	220,495
Rajnandgaon	49	736	1,463	6,771	16,034	15,639	11,658	8,482	4,823	1,539	475	67,669
Sukma	4	96	211	1,194	2,755	2,642	1,257	672	248	62	13	9,154
Surajpur	76	432	916	4,392	8,423	7,773	5,368	3,536	1,732	506	163	33,317
Surguja	39	523	998	4,561	10,083	9,530	6,621	4,360	2,713	884	260	40,572
Uttar Bastar Kanker	78	506	867	3,781	6,903	7,258	4,725	3,240	1,706	528	171	29,763
NA	20	138	393	1,843	3,505	3,519	2,378	1,296	812	255	81	14,240

Note: Data includes only non-missing age groups.

### 5.4.7 Chhattisgarh

#### 5.4.7.1 District Wise Summary Table

#### 5.4.7.2 District wise Age Groups



Positive Cases by District and Age Group  
 COVID-19 data categorized by age groups and district

District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Dadra & Nagar Haveli	10	47	97	533	1,797	1,858	1,300	717	298	129	32	6,818

Note: Data includes only non-missing age groups.

Positive Cases by District and Age Group  
 COVID-19 data categorized by age groups and district

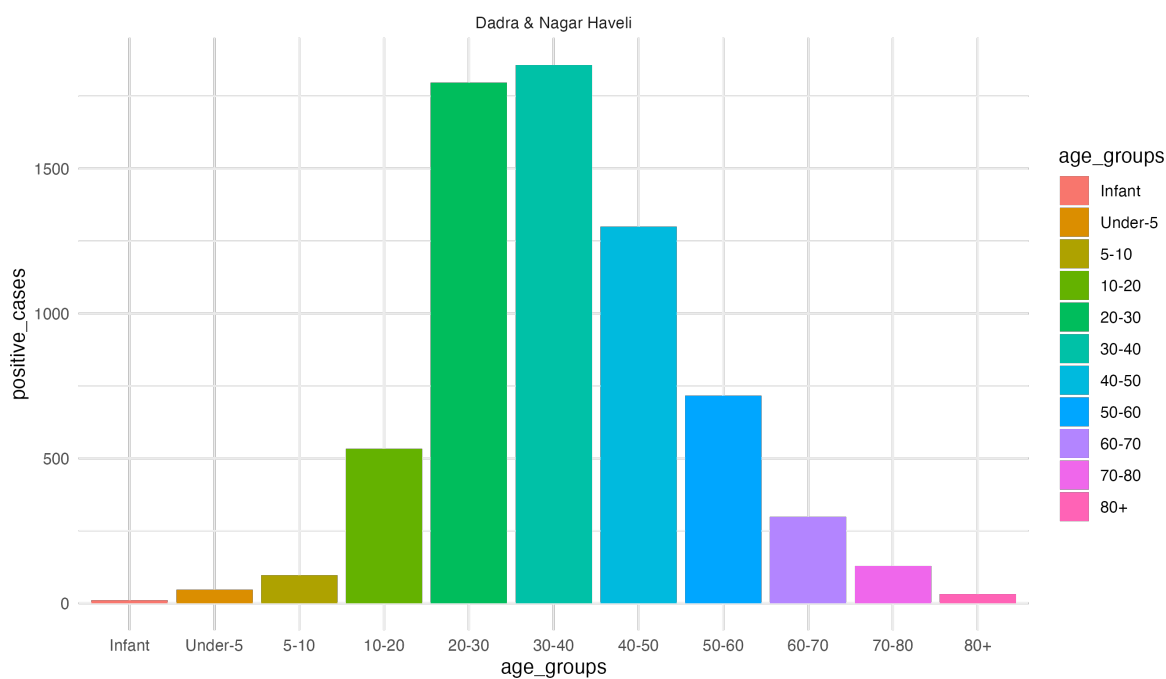
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Daman	6	46	52	262	903	895	670	421	201	97	27	3,580
Diu	2	5	12	39	149	185	140	121	70	29	10	762

Note: Data includes only non-missing age groups.

## 5.4.8 Dadra And Nagar Haveli

### 5.4.8.1 District Wise Summary Table

### 5.4.8.2 District wise Age Groups



## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

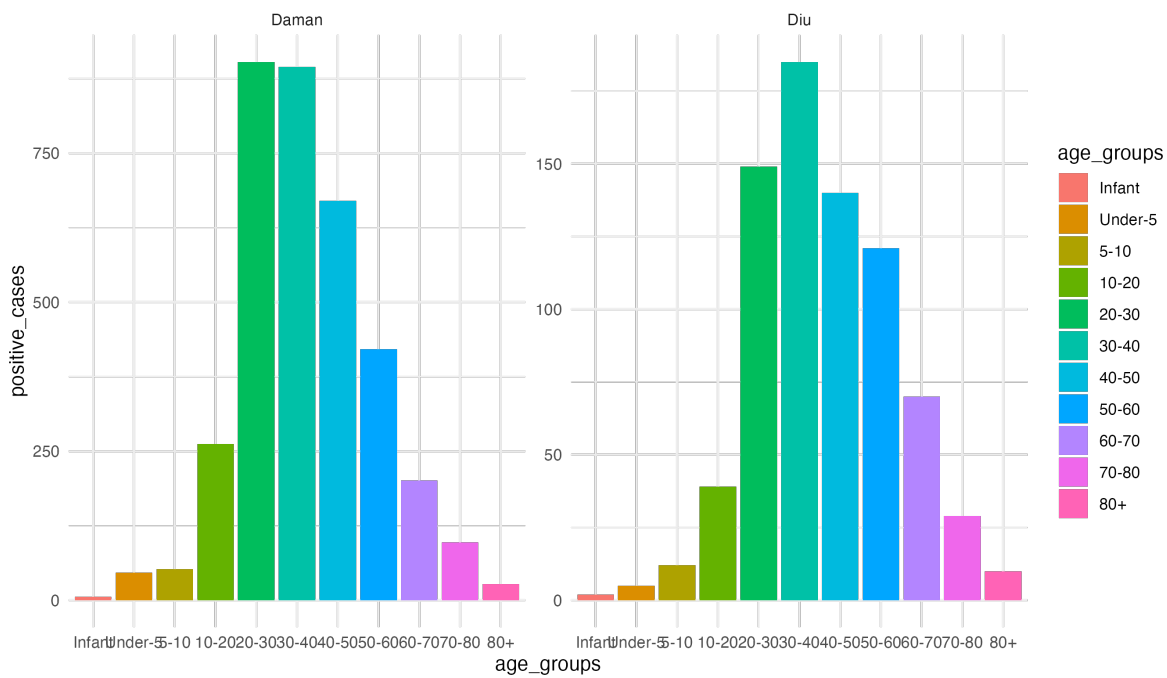
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Central	364	939	1,772	8,329	28,041	28,688	21,224	19,764	12,713	6,171	2,446	130,451
East	686	1,770	2,836	10,928	36,710	41,463	29,616	27,861	18,741	8,676	3,345	182,632
New Delhi	542	2,128	4,207	17,950	65,860	72,368	48,252	43,168	23,868	11,786	5,172	295,301
North	184	1,017	1,971	10,167	29,429	30,903	22,105	18,793	11,816	5,058	1,604	133,047
North East	153	597	1,097	5,545	16,335	16,003	11,090	8,678	5,270	1,967	567	67,302
North West	418	2,019	3,754	18,242	55,778	62,123	47,180	44,223	30,289	13,699	4,609	282,334
Shahdara	188	708	1,315	7,128	23,397	23,679	17,168	14,615	8,423	3,400	955	100,976
South	636	2,213	4,105	18,702	63,693	69,371	47,806	39,582	25,831	13,863	6,462	292,264
South East	230	928	2,202	10,199	34,965	37,494	27,652	23,091	14,214	7,086	2,936	160,997
South West	507	2,356	4,188	17,319	50,237	64,036	43,475	34,895	22,285	10,765	3,690	253,753
West	785	2,834	4,958	21,420	64,098	75,173	53,951	49,140	36,411	17,627	6,538	332,935

Note: Data includes only non-missing age groups.

### 5.4.9 Daman And Diu

#### 5.4.9.1 District Wise Summary Table

#### 5.4.9.2 District wise Age Groups



## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

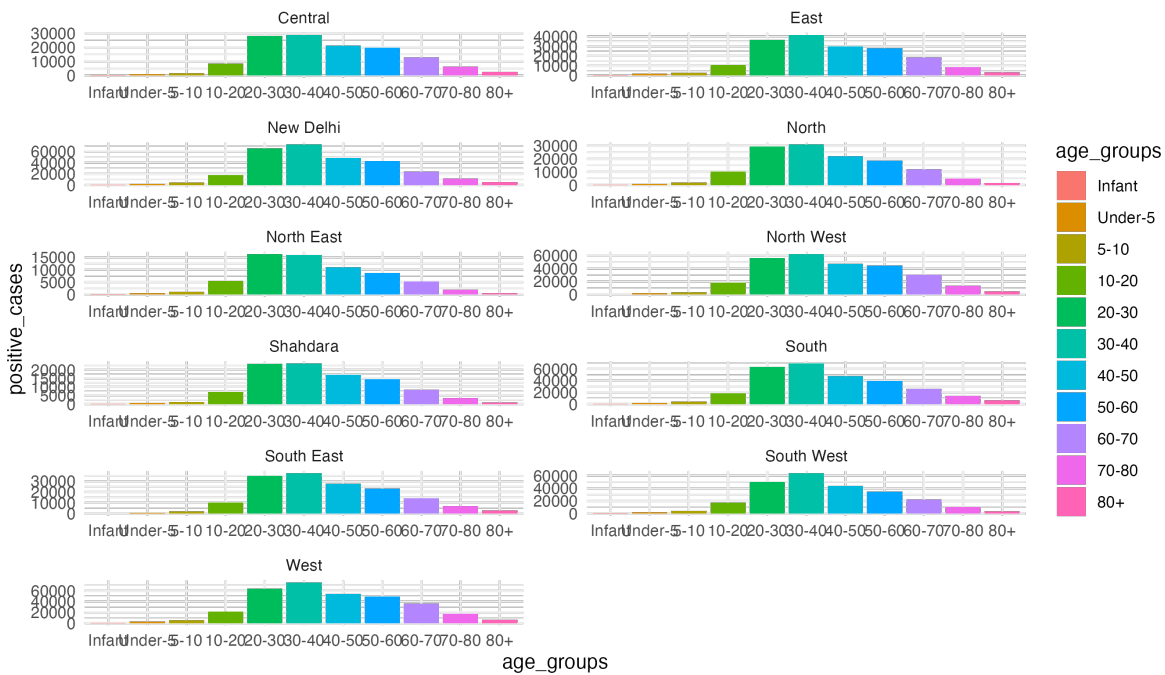
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
North Goa	387	1,843	2,963	10,663	28,740	28,408	21,786	17,380	11,957	6,226	1,995	132,348
South Goa	519	2,547	3,846	12,936	31,237	28,938	22,869	19,461	12,573	6,498	2,106	143,530

Note: Data includes only non-missing age groups.

### 5.4.10 Delhi

#### 5.4.10.1 District Wise Summary Table

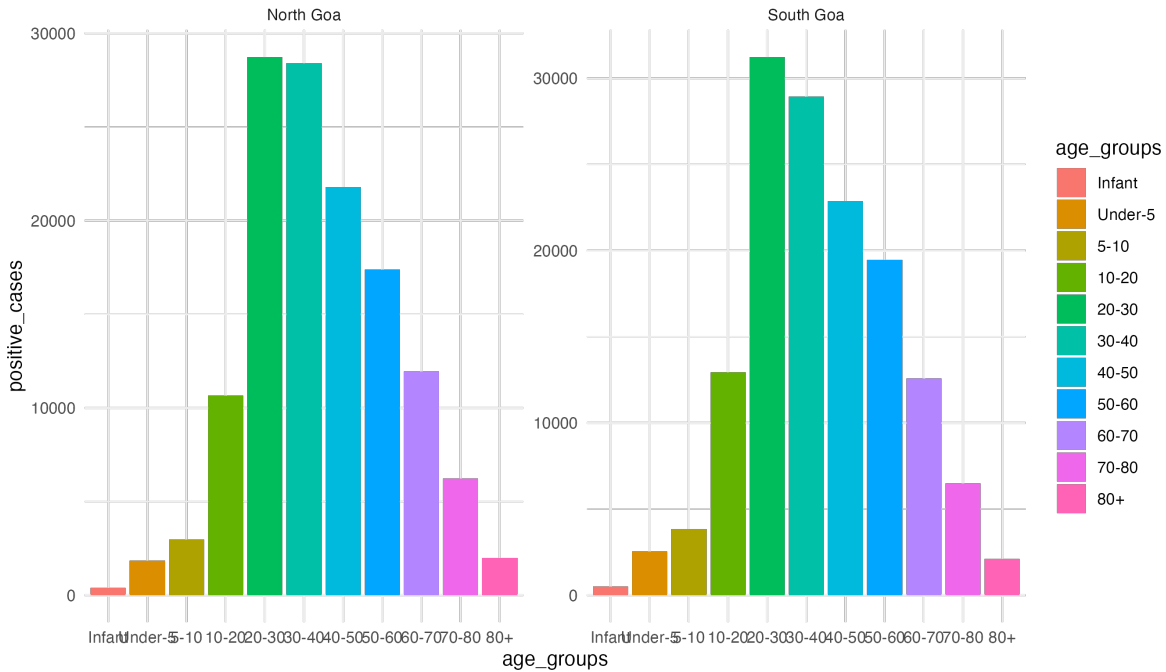
#### 5.4.10.2 District wise Age Groups



## 5.4.11 Goa

### 5.4.11.1 District Wise Summary Table

### 5.4.11.2 District wise Age Groups



## 5.4.12 Gujarat

### 5.4.12.1 District Wise Summary Table

### 5.4.12.2 District wise Age Groups



Positive Cases by District and Age Group  
COVID-19 data categorized by age groups and district

District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Ahmadabad	851	2,870	5,683	25,325	79,710	90,884	68,078	62,735	47,956	22,460	8,357	414,909
Amreli	12	54	131	1,176	3,606	4,245	4,401	3,828	3,067	1,567	548	22,635
Anand	27	97	208	1,165	4,064	4,488	3,499	3,229	2,528	1,342	466	21,113
Aravalli	4	28	65	523	1,769	1,865	1,599	1,420	1,013	379	122	8,787
Banas Kantha	15	55	155	1,532	5,033	4,843	3,445	2,756	1,861	736	197	20,628
Bharuch	8	78	181	1,256	5,998	6,562	4,871	4,499	2,679	1,118	362	27,612
Bhavnagar	55	146	325	2,125	6,610	6,897	6,221	6,444	5,274	2,388	911	37,396
Botad	3	5	19	96	348	406	322	477	406	190	63	2,335
Chota Udaipur	4	17	29	317	835	940	681	620	383	137	41	4,004
Devbhumi Dwarka	7	25	54	369	1,278	1,472	1,555	1,592	1,231	545	201	8,329
Dohad	16	80	167	1,214	3,951	4,040	2,888	2,323	1,625	638	195	17,137
Gandhinagar	85	265	580	3,443	11,418	12,436	8,842	7,824	5,389	2,336	767	53,385
Gir Somnath	8	47	103	809	2,782	2,644	2,040	1,539	934	367	114	11,387
Jamnagar	38	146	329	2,065	6,969	8,060	6,788	6,597	4,814	2,221	864	38,891
Junagadh	6	68	169	1,388	4,455	4,789	4,556	3,790	2,832	1,083	396	23,532
Kachchh	29	99	230	1,829	6,321	6,338	4,572	3,742	2,595	1,166	441	27,362
Kheda	31	98	182	1,013	3,313	3,849	3,309	3,322	2,580	1,197	342	19,236
Mahesana	29	113	322	2,117	7,213	7,290	5,939	5,537	3,997	1,891	534	34,982
Mahisagar	5	17	47	608	1,856	1,964	1,884	1,751	1,364	539	112	10,147
Morbi	10	75	159	757	2,453	2,504	1,809	1,864	1,472	633	231	11,967
Narmada	5	47	132	778	2,481	2,524	1,915	1,491	984	313	98	10,768
Navsari	11	75	185	1,332	3,505	3,951	3,268	2,582	1,773	724	205	17,611
Panch Mahals	11	42	102	781	2,858	2,935	2,284	2,180	1,489	574	160	13,416
Patan	13	67	144	1,181	3,857	3,803	3,050	2,877	2,291	1,083	321	18,687
Porbandar	4	25	47	388	1,139	1,115	1,070	863	589	306	76	5,622
Rajkot	121	584	1,226	7,034	20,809	23,676	19,938	18,586	14,520	6,956	2,510	115,960
Sabar Kantha	15	39	97	701	2,842	2,965	2,456	2,224	1,558	651	157	13,705
Surat	187	1,270	2,677	17,316	43,305	46,067	37,691	30,711	19,997	8,984	3,004	211,209
Surendranagar	4	28	70	768	3,120	3,098	2,394	2,341	1,838	772	251	14,684
Tapi	21	54	112	864	2,435	2,870	2,456	2,174	1,409	521	119	13,035
The Dangs	1	9	16	280	346	320	202	187	152	40	11	1,564
Vadodara	188	869	1,968	10,052	32,444	38,262	29,616	27,170	20,287	9,635	3,585	174,076
Valsad	15	102	191	1,833	5,705	6,269	5,229	4,060	2,464	1,066	334	27,268

Note: Data includes only non-missing age groups.

## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Ambala	122	506	1,063	4,893	12,935	15,446	11,262	9,396	7,404	3,604	1,101	67,732
Bhiwani	52	226	446	2,600	7,730	7,795	5,255	3,995	2,844	1,320	355	32,618
Charki Dadri	7	78	140	970	2,713	2,703	1,763	1,230	918	377	116	11,015
Faridabad	452	1,569	2,886	11,587	34,562	45,055	31,611	24,627	16,236	7,865	2,635	179,085
Fatehabad	26	150	321	2,234	6,170	5,873	3,781	3,144	2,256	847	251	25,053
Gurugram	554	3,307	7,784	25,858	87,255	116,722	64,352	41,393	31,093	16,446	5,948	400,712
Hisar	93	479	900	5,541	17,684	18,302	12,474	10,326	6,843	3,083	869	76,594
Jhajjar	62	235	420	2,127	7,815	8,257	4,840	3,475	2,345	1,171	410	31,157
Jind	26	152	306	2,158	6,964	6,738	4,449	3,597	2,553	1,036	278	28,257
Kaithal	35	121	240	1,333	4,437	4,048	2,805	2,268	1,826	840	220	18,173
Karnal	127	534	890	4,830	12,554	13,571	9,460	7,663	5,381	2,452	696	58,158
Kurukshetra	44	217	459	3,056	6,257	7,244	5,455	4,553	3,462	1,594	473	32,814
Mahendragarh	47	267	552	3,033	6,174	5,929	4,298	2,901	2,286	1,122	425	27,034
Nuh	57	59	104	534	2,228	1,730	981	670	462	200	73	7,098
Palwal	39	170	284	1,687	6,013	5,762	3,268	2,307	1,631	614	201	21,976
Panchkula	92	576	1,261	6,107	16,256	19,295	13,404	10,807	7,629	4,413	1,630	81,470
Panipat	49	303	615	3,072	9,540	10,338	6,803	5,809	3,652	1,466	457	42,104
Rewari	41	229	449	2,651	9,910	10,721	6,332	4,272	2,885	1,393	391	39,274
Rohtak	122	485	876	4,810	17,359	18,075	11,150	9,164	6,490	2,845	910	72,286
Sirsa	56	262	459	2,692	8,079	8,998	6,579	5,232	3,623	1,526	522	38,028
Sonipat	154	576	984	5,291	17,648	18,239	11,044	8,416	5,644	2,603	803	71,402
Yamunanagar	70	258	496	2,787	7,831	8,534	6,276	5,683	4,613	1,981	630	39,159

Note: Data includes only non-missing age groups.

### 5.4.13 Haryana

#### 5.4.13.1 District Wise Summary Table

#### 5.4.13.2 District wise Age Groups



Positive Cases by District and Age Group  
COVID-19 data categorized by age groups and district

District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Bilaspur	99	361	700	2,845	3,983	4,241	3,572	2,697	1,758	997	503	21,756
Chamba	26	229	487	2,660	3,990	4,051	3,069	2,151	1,365	718	270	19,016
Hamirpur	53	287	667	3,062	4,864	5,197	4,588	3,442	2,559	1,545	743	27,007
Kangra	182	803	1,410	6,354	11,089	12,518	10,270	7,857	5,954	3,406	1,376	61,219
Kinnaur	2	2	6	524	1,429	1,448	997	656	283	157	72	5,576
Kullu	48	86	137	1,223	2,762	2,796	2,228	1,568	1,069	482	189	12,588
Lahul & Spiti	8	21	46	406	1,040	873	650	401	187	73	42	3,747
Mandi	124	372	793	4,837	9,098	9,455	8,335	6,135	3,991	2,186	1,022	46,348
Shimla	43	199	384	2,887	8,323	8,374	6,644	5,549	3,112	1,607	670	37,792
Sirmaur	60	199	327	2,065	5,308	5,137	3,709	2,517	1,388	634	246	21,590
Solan	77	271	543	3,177	10,437	9,491	6,396	4,253	2,433	1,129	420	38,627
Una	33	222	422	2,390	3,803	4,668	3,583	2,745	1,869	987	437	21,159

Note: Data includes only non-missing age groups.

Positive Cases by District and Age Group  
COVID-19 data categorized by age groups and district

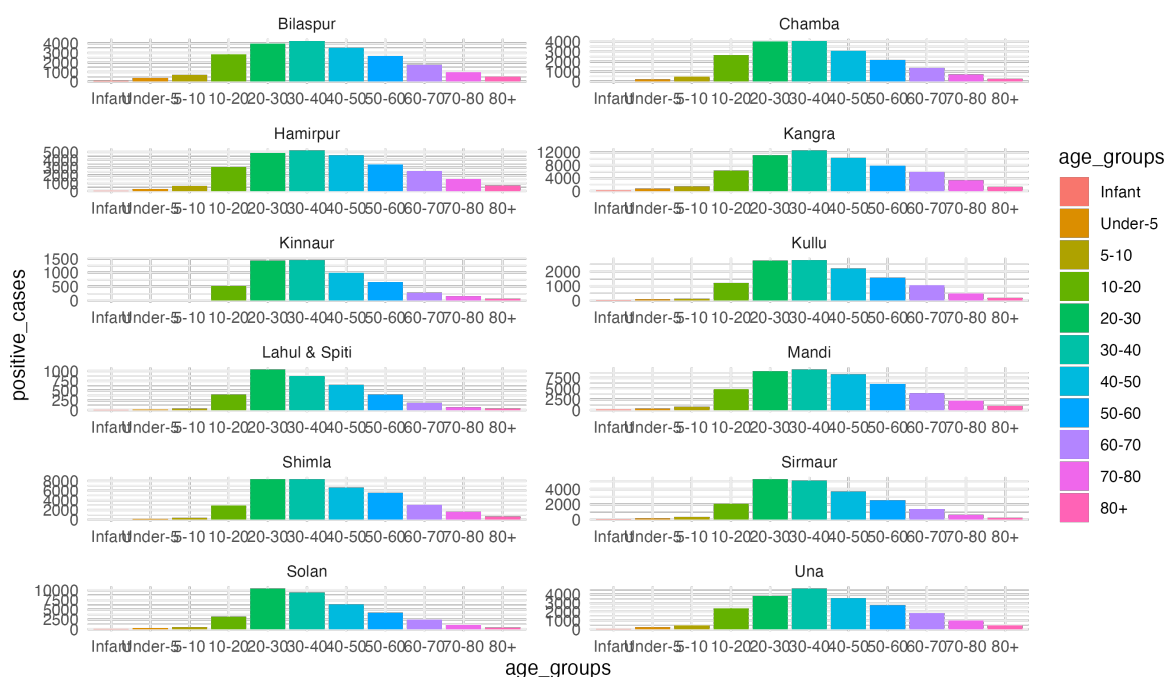
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
NA	71	271	419	2,534	10,105	9,199	5,533	4,255	3,020	1,390	410	37,207

Note: Data includes only non-missing age groups.

### 5.4.14 Himachal Pradesh

#### 5.4.14.1 District Wise Summary Table

#### 5.4.14.2 District wise Age Groups



Positive Cases by District and Age Group  
COVID-19 data categorized by age groups and district

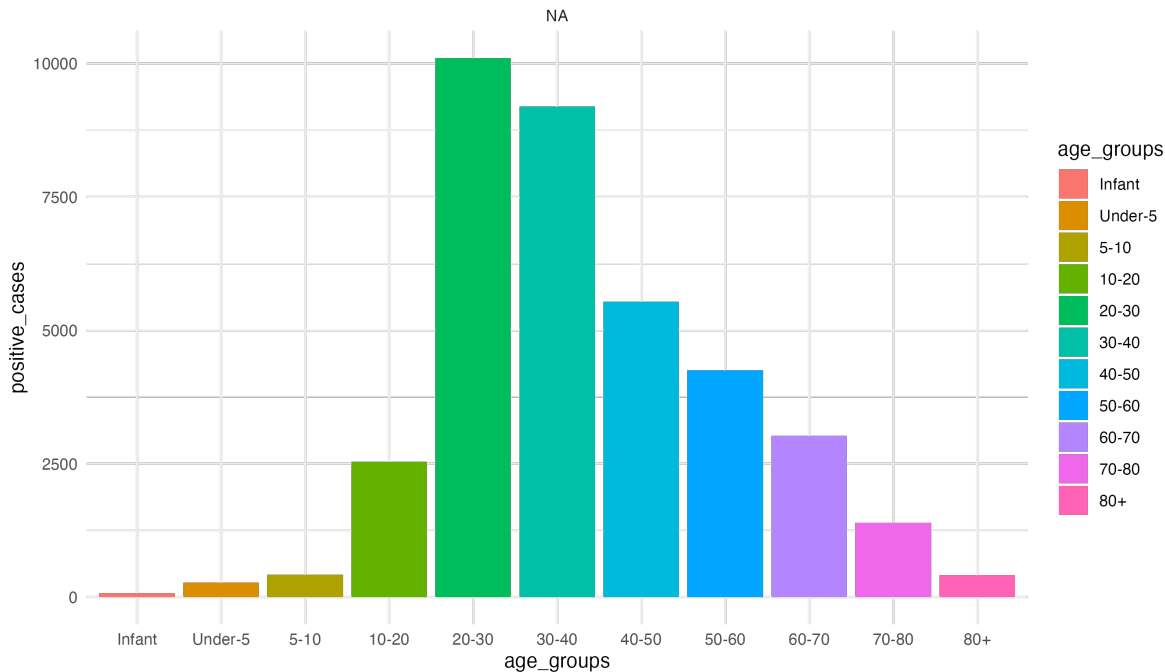
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Anantnag	14	96	201	1,264	4,648	5,101	3,174	2,134	1,483	584	156	18,855
Badgam	41	343	748	3,514	9,613	10,735	6,442	4,050	2,769	1,202	305	39,762
Bandipore	8	71	162	1,080	4,008	3,445	1,900	1,318	841	368	148	13,349
Baramula	12	121	222	1,421	5,867	4,682	2,952	2,018	1,418	673	254	19,640
Doda	8	36	59	736	2,421	1,934	1,161	815	455	258	108	7,991
Ganderbal	27	124	276	1,490	4,363	4,480	2,703	1,790	1,125	572	167	17,117
Jammu	130	718	1,402	6,765	22,024	27,290	19,389	16,890	10,978	4,812	2,201	112,599
Kathua	10	68	106	808	2,688	2,758	2,066	1,534	781	371	171	11,361
Kishtwar	21	35	51	360	1,117	1,181	809	594	330	167	83	4,748
Kulgam	10	53	171	1,038	3,327	3,789	2,300	1,452	889	369	83	13,481
Kupwara	50	112	224	2,004	5,198	5,198	3,056	2,111	1,285	630	179	20,047
Pulwama	16	150	410	2,032	4,997	6,107	3,736	2,542	1,777	767	238	22,772
Punch	5	31	43	584	1,812	1,722	1,158	840	386	146	63	6,790
Rajouri	8	41	102	879	2,506	2,499	1,729	1,233	777	289	154	10,217
Ramban	26	48	84	626	2,591	2,107	1,483	872	497	195	77	8,606
Reasi	2	37	64	458	1,857	1,909	1,382	1,088	785	467	187	8,236
Samba	13	60	114	681	2,286	2,678	1,871	1,261	476	185	85	9,710
Shupiyan	4	38	60	570	1,529	1,526	981	592	422	210	53	5,985
Srinagar	100	1,030	1,961	7,770	23,086	28,687	19,803	15,662	10,246	4,762	1,554	114,661
Udhampur	26	122	227	1,608	6,274	5,578	3,816	2,447	1,313	574	217	22,202

Note: Data includes only non-missing age groups.

## 5.4.15 India

### 5.4.15.1 District Wise Summary Table

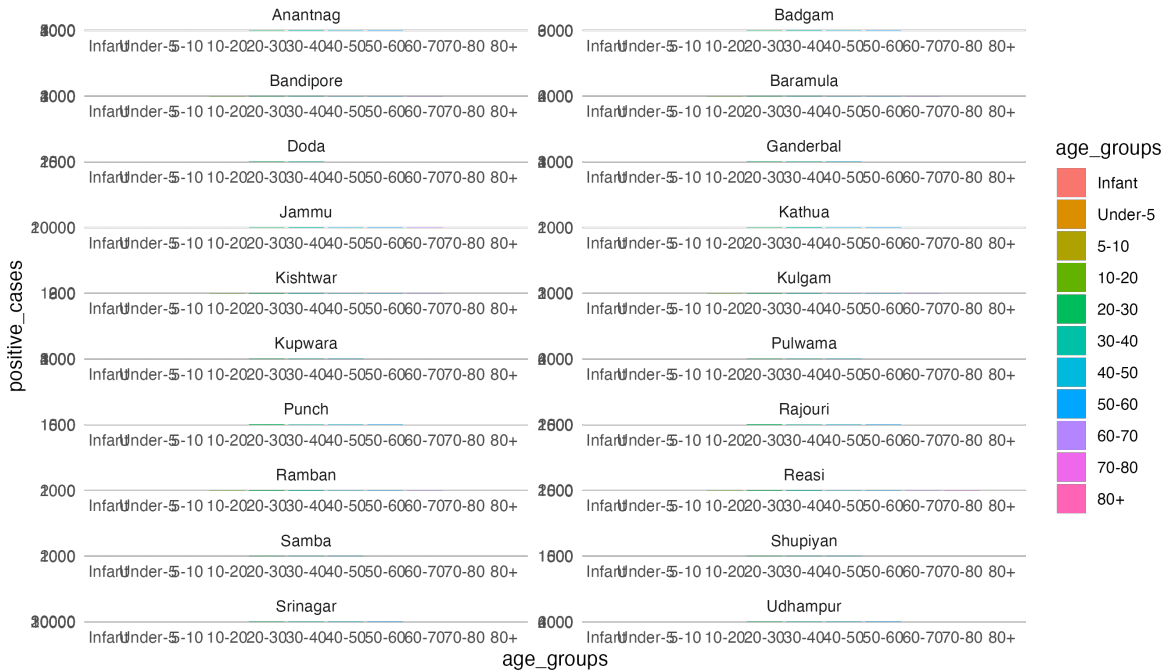
### 5.4.15.2 District wise Age Groups



## 5.4.16 Jammu And Kashmir

### 5.4.16.1 District Wise Summary Table

### 5.4.16.2 District wise Age Groups



## 5.4.17 Jharkhand

### 5.4.17.1 District Wise Summary Table

### 5.4.17.2 District wise Age Groups



## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

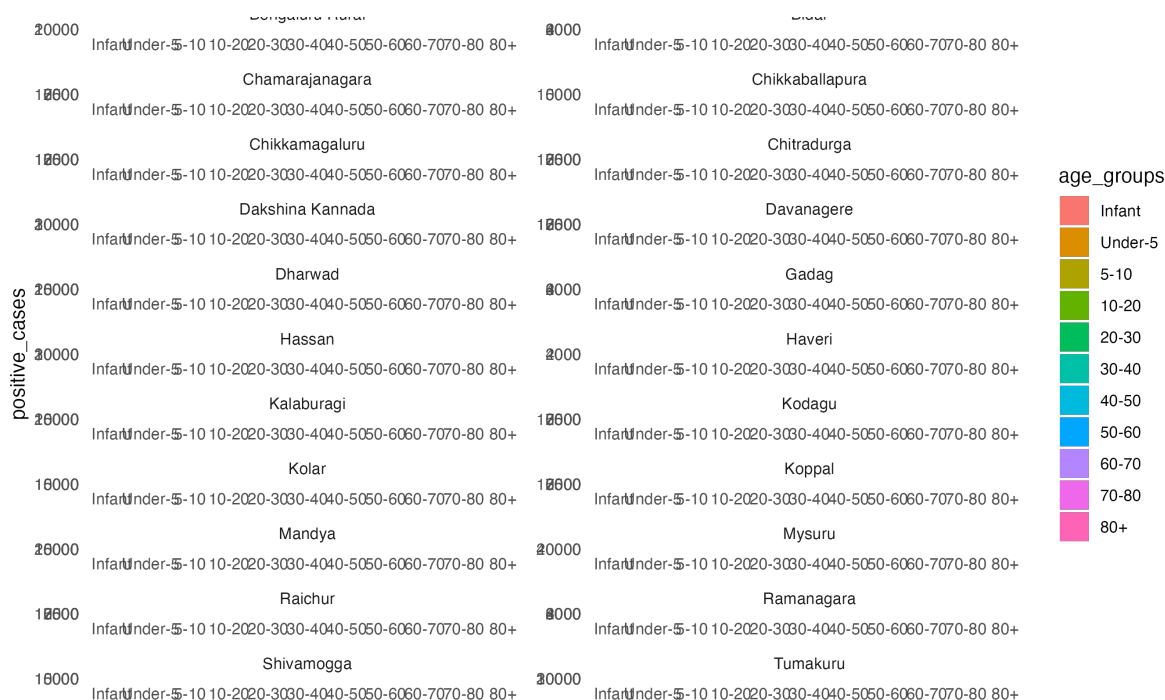
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Bokaro	13	147	338	1,800	5,936	7,082	5,700	4,512	2,394	1,219	200	29,341
Chatra	3	33	93	824	2,254	2,217	1,522	979	397	132	54	8,508
Deoghar	19	107	253	1,414	4,029	4,133	3,445	2,491	1,215	361	112	17,579
Dhanbad	25	131	275	1,977	6,808	7,016	5,230	4,899	2,443	1,008	254	30,066
Dumka	10	73	158	1,112	2,301	2,515	1,825	1,297	645	223	48	10,207
Garhwa	5	59	156	1,124	2,536	2,635	1,715	1,025	654	172	71	10,152
Giridih	10	69	146	851	2,421	2,867	2,644	1,823	1,036	273	65	12,205
Godda	4	43	126	899	2,735	2,504	1,880	1,291	683	253	49	10,467
Gumla	12	139	245	1,396	3,155	2,973	2,299	1,478	811	257	65	12,830
Hazaribagh	12	162	368	2,513	5,933	6,623	4,496	3,464	1,933	736	216	26,456
Jamtara	9	45	79	638	1,762	1,819	1,461	978	555	207	51	7,604
Khunti	5	57	162	982	2,605	2,656	1,661	977	544	145	52	9,846
Kodarma	5	115	211	1,616	3,981	4,043	2,724	1,809	1,048	324	100	15,976
Latehar	10	80	141	930	2,451	2,763	1,634	1,017	420	149	46	9,641
Lohardaga	6	61	133	934	2,268	1,874	1,398	993	476	164	51	8,358
Pakur	3	16	38	447	976	983	782	479	198	47	18	3,987
Palamu	13	106	273	1,637	4,253	4,410	3,064	1,936	1,111	433	135	17,371
Pashchimi Singhbhum	15	100	237	1,844	4,352	4,742	3,415	2,538	1,124	389	122	18,878
Purbi Singhbhum	143	676	1,156	6,174	18,752	20,224	16,336	14,123	8,076	4,250	1,434	91,344
Ramgarh	22	103	214	1,573	3,851	4,103	3,122	2,316	843	314	92	16,553
Ranchi	245	1,048	1,877	11,080	32,721	34,633	25,634	22,030	13,607	5,842	1,814	150,531
Sahibganj	7	70	237	987	2,104	1,972	1,426	994	507	150	35	8,489
Saraikela-kharsawan	7	47	111	895	2,716	2,749	1,945	1,217	561	193	46	10,487
Simdega	6	90	169	1,820	3,809	3,386	2,210	1,430	715	212	56	13,903

Note: Data includes only non-missing age groups.

### 5.4.18 Karnataka

#### 5.4.18.1 District Wise Summary Table

#### 5.4.18.2 District wise Age Groups



## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

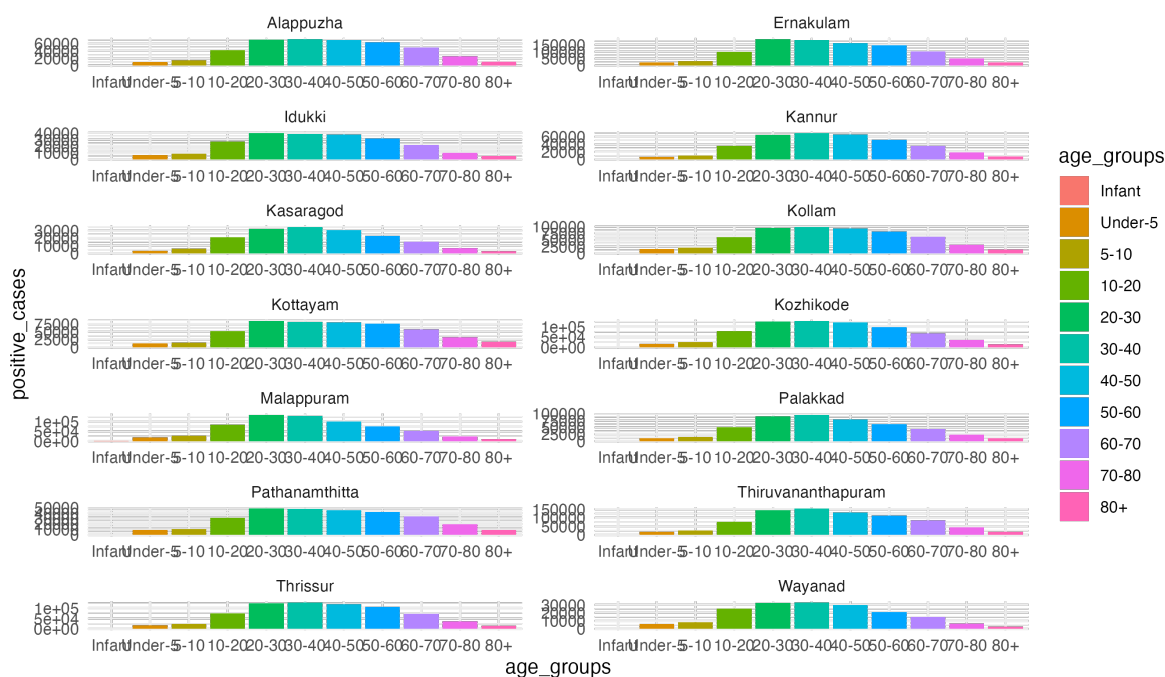
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Bagalkote	49	375	843	5,461	9,972	9,134	7,576	5,795	4,111	1,734	505	45,555
Ballari	163	1,043	2,309	13,416	27,312	27,216	21,442	15,047	9,908	3,799	1,146	122,801
Bangalore	4,054	20,756	36,732	150,083	487,862	523,667	337,837	222,663	155,301	78,884	27,940	2,045,779
Belagavi	115	621	1,471	10,585	24,124	21,569	17,110	14,354	10,984	5,305	1,683	107,921
Bengaluru Rural	148	876	2,046	11,226	25,464	26,439	17,787	10,869	7,184	3,394	1,159	106,592
Bidar	28	346	604	3,942	7,143	7,599	5,915	4,717	3,379	1,389	391	35,453
Chamarajanagara	37	325	844	6,211	9,126	9,637	8,139	5,484	4,236	1,889	609	46,537
Chikkaballapura	116	527	1,176	7,016	12,828	12,321	9,743	6,688	5,048	2,041	679	58,183
Chikkamagaluru	90	715	1,399	6,698	11,275	12,174	10,310	8,435	6,119	2,557	935	60,707
Chitradurga	71	449	1,042	6,140	9,326	9,688	8,227	6,276	4,427	1,772	535	47,953
Dakshina Kannada	310	2,080	3,529	15,081	31,027	26,078	21,794	19,851	14,649	7,956	2,952	145,307
Davanagere	71	522	1,043	5,906	10,659	12,147	10,744	8,891	6,726	2,750	883	60,342
Dharwad	125	629	1,599	10,866	19,383	18,152	14,703	12,401	9,266	4,359	1,439	92,922
Gadag	32	313	617	3,125	6,868	6,530	5,308	4,048	3,400	1,364	422	32,027
Hassan	144	1,339	3,256	21,871	29,758	31,680	26,166	19,215	13,796	5,605	1,708	154,538
Haveri	25	243	667	3,476	5,488	5,457	4,625	3,752	2,943	1,234	343	28,253
Kalaburagi	145	950	1,886	10,688	19,641	18,526	13,734	11,283	7,661	3,178	927	88,619
Kodagu	72	772	1,945	8,876	10,126	9,725	7,982	6,099	4,316	1,935	629	52,477
Kolar	76	459	1,463	8,472	16,368	14,401	11,041	7,859	5,174	2,232	767	68,312
Koppal	54	486	929	5,316	10,345	9,225	6,825	4,525	3,314	1,191	327	42,537
Mandya	167	905	2,046	16,464	19,980	21,803	17,675	12,792	9,456	3,841	1,219	106,348
Mysuru	431	1,883	4,068	26,416	53,199	53,119	41,064	31,452	20,721	9,571	3,396	245,320
Raichur	69	508	1,010	5,209	12,247	11,351	8,390	6,558	4,480	1,521	377	51,720
Ramanagara	41	238	596	3,701	7,447	7,796	5,649	3,940	2,936	1,351	418	34,113
Shivamogga	108	787	1,764	10,554	17,870	18,721	16,485	13,922	10,501	4,311	1,401	96,424
Tumakuru	163	1,254	3,119	21,508	32,450	34,482	28,746	22,011	15,858	6,663	2,196	168,450
Udupi	205	1,746	3,405	14,155	18,611	17,852	15,173	12,469	9,834	5,377	2,187	101,014
Uttara Kannada	116	1,073	2,385	11,412	13,963	13,882	11,932	9,391	6,956	3,264	1,220	75,594
Vijayapura	55	468	994	4,552	9,974	9,090	7,052	5,678	3,798	1,601	504	43,766
Yadgir	68	366	798	3,877	9,073	7,370	4,944	3,325	2,209	764	195	32,989

Note: Data includes only non-missing age groups.

### 5.4.19 Kerala

#### 5.4.19.1 District Wise Summary Table

#### 5.4.19.2 District wise Age Groups



Positive Cases by District and Age Group  
COVID-19 data categorized by age groups and district

District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Alappuzha	106	10,209	14,179	42,644	70,468	71,531	69,918	63,760	49,174	25,765	11,620	429,374
Ernakulam	183	24,034	32,455	100,053	188,714	183,233	162,582	147,130	101,803	52,134	22,887	1,015,208
Idukki	34	6,883	8,623	26,917	39,569	38,071	37,434	31,476	21,589	9,988	5,023	225,607
Kannur	115	6,885	10,444	35,914	65,081	69,227	65,435	51,568	35,822	18,651	8,524	367,666
Kasaragod	66	3,666	6,215	21,166	32,160	34,480	30,503	22,695	15,689	7,035	2,684	176,359
Kollam	170	16,468	21,034	59,957	94,073	97,865	92,193	81,002	61,304	32,309	13,881	570,256
Kottayam	92	12,759	15,896	52,819	85,495	82,988	80,677	76,435	57,776	32,982	17,810	515,729
Kozhikode	201	16,448	24,278	79,782	125,924	128,773	120,915	98,198	69,282	34,909	13,864	712,574
Malappuram	335	20,195	27,786	85,023	135,026	129,267	101,103	75,619	52,417	24,200	9,905	660,876
Palakkad	77	10,626	15,541	50,758	90,166	95,827	79,585	61,726	43,756	22,997	10,321	481,380
Pathanamthitta	44	8,791	11,114	31,465	48,044	47,565	45,048	42,291	33,970	19,311	9,499	297,142
Thiruvananthapuram	197	19,539	26,074	77,561	144,660	153,228	131,832	114,514	84,571	44,309	18,801	815,286
Thrissur	138	17,321	24,196	74,965	124,124	126,846	118,039	105,828	72,340	36,618	15,334	715,749
Wayanad	27	6,054	8,101	25,295	32,464	32,890	29,537	20,740	14,706	6,452	2,621	178,887

Note: Data includes only non-missing age groups.

Positive Cases by District and Age Group  
COVID-19 data categorized by age groups and district

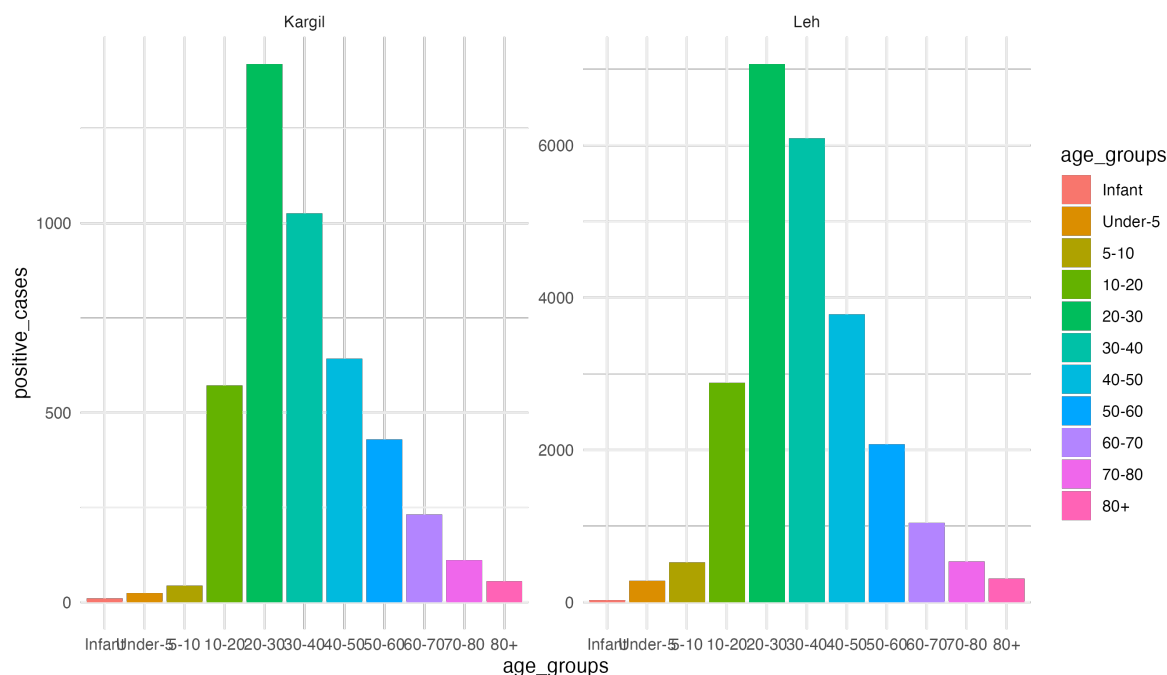
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Kargil	10	24	44	572	1,420	1,026	643	430	232	111	56	4,568
Leh	29	280	523	2,881	7,069	6,094	3,779	2,074	1,044	538	312	24,623

Note: Data includes only non-missing age groups.

## 5.4.20 Ladakh

### 5.4.20.1 District Wise Summary Table

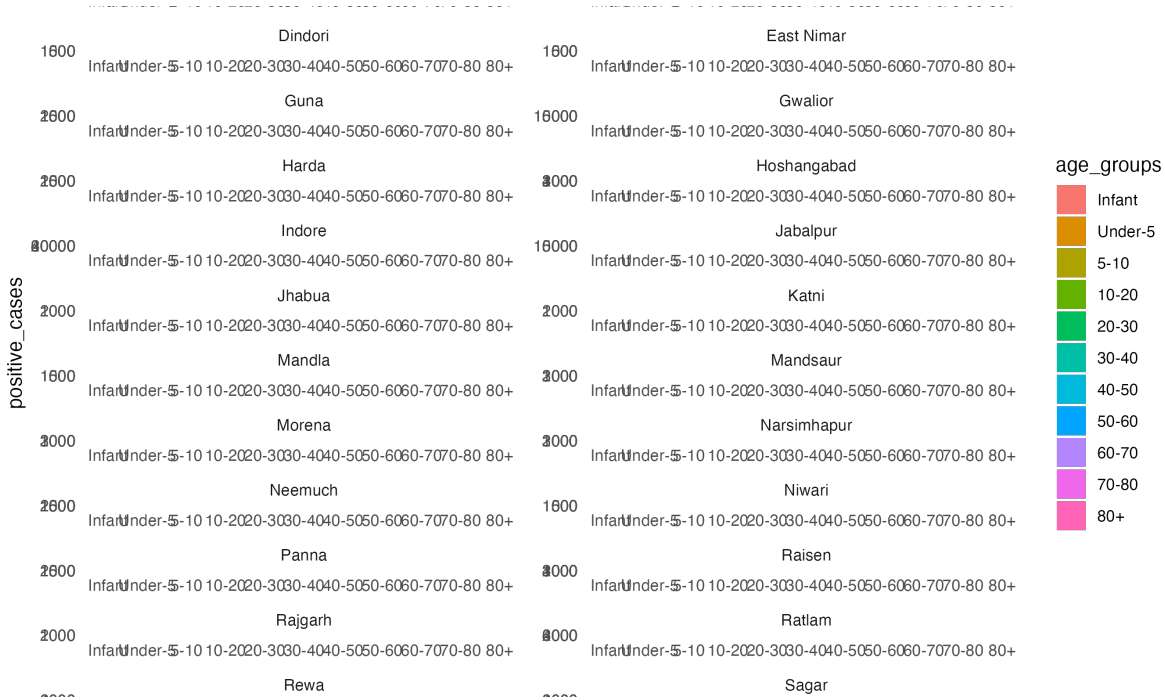
### 5.4.20.2 District wise Age Groups



## 5.4.21 Madhya Pradesh

### 5.4.21.1 District Wise Summary Table

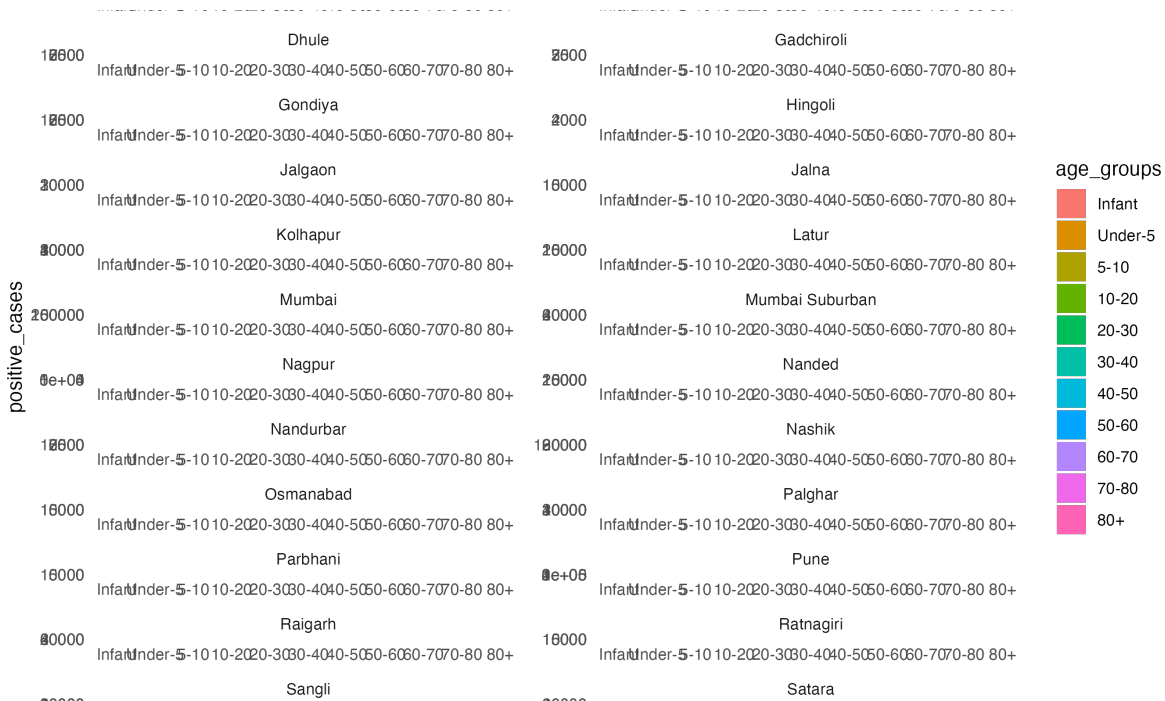
### 5.4.21.2 District wise Age Groups



## 5.4.22 Maharashtra

### 5.4.22.1 District Wise Summary Table

### 5.4.22.2 District wise Age Groups



Positive Cases by District and Age Group  
COVID-19 data categorized by age groups and district

District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Agar Malwa	4	31	61	339	1,306	1,058	810	615	406	147	50	4,827
Alirajpur	3	39	101	569	1,206	1,098	824	550	310	97	25	4,822
Anuppur	7	63	211	1,098	2,626	2,707	2,085	1,811	829	237	80	11,754
Ashoknagar	4	59	105	596	1,475	1,331	1,018	768	559	213	79	6,207
Balaghat	13	99	212	1,222	3,686	4,171	3,460	2,721	1,349	447	108	17,488
Barwani	7	42	113	818	2,426	2,541	2,197	1,694	1,145	389	151	11,523
Betul	13	155	291	1,593	4,223	4,585	3,671	2,935	1,844	607	235	20,152
Bhind	5	22	37	421	1,316	1,112	739	664	401	180	56	4,953
Bhopal	416	1,821	3,217	15,310	48,817	49,262	35,137	32,689	18,660	7,899	2,883	216,111
Burhanpur	3	26	38	221	703	765	612	632	370	120	39	3,529
Chhatarpur	26	137	263	1,283	3,459	3,136	2,410	2,078	1,220	450	175	14,637
Chhindwara	12	123	199	1,040	3,042	3,505	2,926	2,703	1,666	624	172	16,012
Damoh	7	43	170	1,070	2,882	2,761	2,174	1,893	1,297	439	127	12,863
Datia	22	95	193	956	2,879	2,412	1,755	1,431	884	345	118	11,090
Dewas	16	110	211	1,127	4,009	3,384	3,363	3,108	1,830	663	222	18,043
Dhar	15	115	260	1,844	5,777	5,514	4,424	3,234	2,000	759	207	24,149
Dindori	2	51	126	744	1,453	1,333	1,036	685	372	109	24	5,935
East Nimar	6	24	95	656	1,613	1,458	1,186	1,381	952	371	123	7,865
Guna	11	32	88	605	1,979	1,812	1,289	1,180	723	218	68	8,005
Gwalior	122	502	965	5,081	17,592	16,943	12,548	11,813	7,798	3,230	1,094	77,688
Harda	25	83	121	741	1,938	1,760	1,382	1,043	705	245	94	8,137
Hoshangabad	51	177	291	1,426	4,522	4,679	3,446	3,052	1,853	768	222	20,487
Indore	479	3,079	6,114	23,797	64,302	72,225	54,674	45,790	31,113	14,318	4,934	320,825
Jabalpur	68	416	805	4,692	14,975	17,035	13,321	12,587	8,838	3,934	1,358	78,029
Jhabua	6	95	212	1,148	2,730	2,464	2,053	1,546	925	259	67	11,505
Katni	3	77	149	891	2,636	2,454	1,836	1,751	1,030	398	117	11,342
Mandla	17	91	146	759	1,608	1,676	1,255	1,053	552	178	51	7,386
Mandsaur	6	95	159	951	3,209	3,368	2,943	2,415	1,630	650	227	15,653
Morena	13	85	196	962	2,869	2,550	1,967	1,634	1,100	408	138	11,922
Narsimhapur	13	88	172	1,210	3,680	3,454	2,684	2,103	1,354	519	156	15,433
Neemuch	3	52	129	778	2,306	2,433	1,970	1,731	1,140	463	123	11,128
Niwari	2	26	62	569	1,450	1,142	796	597	310	103	44	5,101
Panna	5	54	134	791	2,251	2,172	1,553	1,283	816	286	113	9,458
Raisen	61	274	355	1,510	4,094	3,656	2,769	2,097	1,315	531	182	16,844
Rajgarh	7	51	141	937	2,834	2,673	2,004	1,622	1,085	397	157	11,908
Ratlam	19	96	248	1,650	6,314	6,320	4,908	4,081	2,762	1,048	387	27,833
Rewa	33	229	373	2,091	6,976	6,419	4,250	3,479	1,939	792	330	26,911
Sagar	22	218	387	2,382	8,034	7,743	5,691	4,660	3,031	1,187	378	33,733
Satna	8	64	139	1,027	3,483	3,600	2,620	2,367	1,438	543	223	15,512
Sehore	32	115	221	1,379	4,267	3,640	2,617	1,897	1,150	434	158	15,910
Seoni	7	112	156	1,180	3,464	3,149	2,561	1,872	1,052	331	89	13,973
Shahdol	15	94	207	1,274	4,260	4,397	3,302	2,650	1,291	478	157	18,125
Shajapur	8	25	45	683	2,269	2,244	1,578	1,303	805	260	69	9,289
Sheopur	7	53	87	490	1,568	1,222	851	681	350	129	34	5,472
Shivpuri	19	116	241	1,366	4,442	4,078	2,974	2,246	1,380	501	175	17,538
Sidhi	3	85	194	1,381	2,889	2,672	1,699	1,213	628	216	102	11,082
Singrauli	13	80	146	660	2,663	2,438	1,667	1,702	504	170	53	10,096
Tikamgarh	7	31	72	907	2,218	2,022	1,587	1,220	804	258	90	9,216
Ujjain	52	237	399	1,962	6,321	6,947	5,693	5,774	4,005	1,716	591	33,697
Umaria	7	99	150	870	2,051	1,969	1,471	1,317	610	147	71	8,762
Vidisha	41	144	280	1,529	4,351	3,798	3,091	2,690	1,741	746	274	18,685
West Nimar	10	64	168	1,529	4,869	4,565	3,834	3,257	2,106	772	217	21,391

Note: Data includes only non-missing age groups.

Positive Cases by District and Age Group  
COVID-19 data categorized by age groups and district

District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Ahmadnagar	229	3,403	7,135	37,143	74,980	86,806	69,082	53,754	40,428	21,292	6,603	400,855
Akola	51	531	996	5,066	13,481	14,891	13,675	11,306	7,998	3,729	1,185	72,909
Amravati	90	877	1,421	7,182	20,524	23,743	20,654	18,906	13,147	6,269	2,000	114,813
Aurangabad	296	2,038	3,720	15,188	39,312	42,602	32,456	25,227	16,738	8,106	2,535	188,218
Bhandara	117	713	1,352	6,582	14,288	15,310	13,875	11,330	6,940	2,553	590	73,650
Bid	160	1,516	2,554	10,797	21,655	25,010	18,662	14,043	12,128	5,476	1,550	113,551
Buldana	72	718	1,400	7,440	17,908	21,958	19,076	15,054	10,560	4,725	1,415	100,326
Chandrapur	103	887	1,833	8,341	19,999	23,058	19,540	16,405	9,304	3,680	869	104,019
Dhule	47	361	746	3,636	8,819	10,373	9,786	8,704	6,278	2,992	850	52,592
Gadchiroli	105	402	803	4,460	8,891	8,969	6,715	5,062	2,871	1,104	185	39,567
Gondiya	72	448	835	4,145	9,586	10,372	8,949	7,009	4,099	1,570	428	47,513
Hingoli	24	161	373	1,898	4,357	5,361	4,302	3,250	2,483	1,020	427	23,656
Jalgaon	172	1,325	2,624	11,261	25,398	31,442	28,894	25,442	18,534	8,394	2,358	155,844
Jalna	33	537	1,221	5,832	13,487	15,825	12,416	9,345	7,321	2,946	1,020	69,983
Kolhapur	200	2,428	4,785	17,951	40,409	50,944	42,176	32,439	25,825	14,159	4,676	235,992
Latur	170	1,279	2,484	10,660	20,250	22,586	19,030	15,007	12,019	6,301	2,099	111,885
Mumbai	1,170	6,287	12,645	55,844	179,760	218,361	184,224	173,766	126,992	76,713	34,619	1,070,381
Mumbai Suburban	328	1,696	3,608	14,353	44,625	58,493	48,039	45,880	36,316	21,208	8,948	283,494
Nagpur	612	6,967	10,527	46,020	120,595	141,504	112,685	93,738	57,791	26,458	8,298	625,195
Nanded	79	879	1,934	8,926	20,404	24,256	20,580	15,408	11,013	4,774	1,655	109,908
Nandurbar	59	534	1,124	4,912	8,640	10,327	9,055	7,010	4,779	1,823	532	48,795
Nashik	884	5,349	9,898	40,210	94,633	116,690	92,633	69,695	45,895	23,450	6,669	506,006
Osmanabad	102	960	1,950	7,131	14,155	16,334	12,917	10,751	9,629	4,684	1,571	80,184
Palghar	338	1,754	3,254	13,238	34,006	42,985	33,321	26,517	17,064	7,610	2,411	182,498
Parbhani	33	449	934	4,621	12,132	13,639	11,014	8,753	6,838	3,179	944	62,536
Pune	2,499	20,645	36,333	115,046	322,626	414,863	264,649	183,672	134,504	71,979	25,167	1,591,983
Raigarh	339	2,942	5,521	18,454	51,140	68,562	48,871	38,891	24,320	10,594	3,715	273,349
Ratnagiri	104	998	1,961	7,978	14,541	16,837	15,811	13,332	10,079	5,467	1,932	89,040
Sangli	299	2,967	5,989	22,835	44,383	50,892	40,711	31,860	25,700	14,461	5,592	245,689
Satara	377	3,215	6,419	26,244	51,438	59,904	48,532	38,348	32,181	18,580	6,660	291,898
Sindhudurg	102	833	1,673	5,824	8,897	10,847	10,602	8,821	7,340	4,065	1,358	60,362
Solapur	246	2,094	4,539	20,559	44,952	50,500	40,644	34,086	27,158	14,807	4,374	243,959
Thane	1,319	9,514	16,932	57,675	153,756	207,693	159,185	130,696	90,141	42,531	14,318	883,760
Wardha	99	644	1,052	5,242	13,721	15,073	13,115	10,615	7,114	3,511	911	71,097
Washim	22	263	687	4,327	9,794	10,904	9,179	6,874	4,810	1,917	504	49,281
Yavatmal	61	605	1,217	6,633	15,798	18,220	17,119	13,354	8,683	3,821	1,048	86,559

Note: Data includes only non-missing age groups.

## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Bishnupur	61	305	553	1,725	2,172	2,615	2,069	1,593	1,274	689	314	13,370
Chandel	13	60	115	333	617	729	464	283	165	93	43	2,915
Churachandpur	50	282	537	1,605	2,473	2,782	1,547	1,242	867	473	192	12,050
Imphal East	113	682	1,415	3,736	6,200	7,355	5,458	4,465	3,123	1,671	721	34,939
Imphal West	187	1,015	1,880	5,209	8,219	10,254	7,998	6,251	4,797	2,719	1,139	49,668
Jiribam	4	16	31	130	251	376	203	125	50	40	8	1,234
Kakching	10	140	366	970	1,226	1,382	1,141	824	555	380	118	7,112
Kamjong	NA	11	30	101	310	242	160	129	68	28	11	1,090
Kangpokpi	15	52	77	447	899	909	475	269	211	81	34	3,469
Noney	3	14	26	116	247	270	134	96	42	19	11	978
Pherzawl	10	18	37	110	117	85	47	46	25	16	4	515
Senapati	6	24	39	301	767	631	354	245	133	60	26	2,586
Tamenglong	28	132	190	602	1,050	945	469	350	230	126	53	4,175
Tengnoupal	7	9	12	150	291	357	207	149	66	28	11	1,287
Thoubal	31	228	435	1,336	2,166	2,364	1,625	1,293	816	450	120	10,864
Ukhrul	17	52	86	371	813	866	521	366	232	130	82	3,536

Note: Data includes only non-missing age groups.

### 5.4.23 Manipur

#### 5.4.23.1 District Wise Summary Table

#### 5.4.23.2 District wise Age Groups



## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

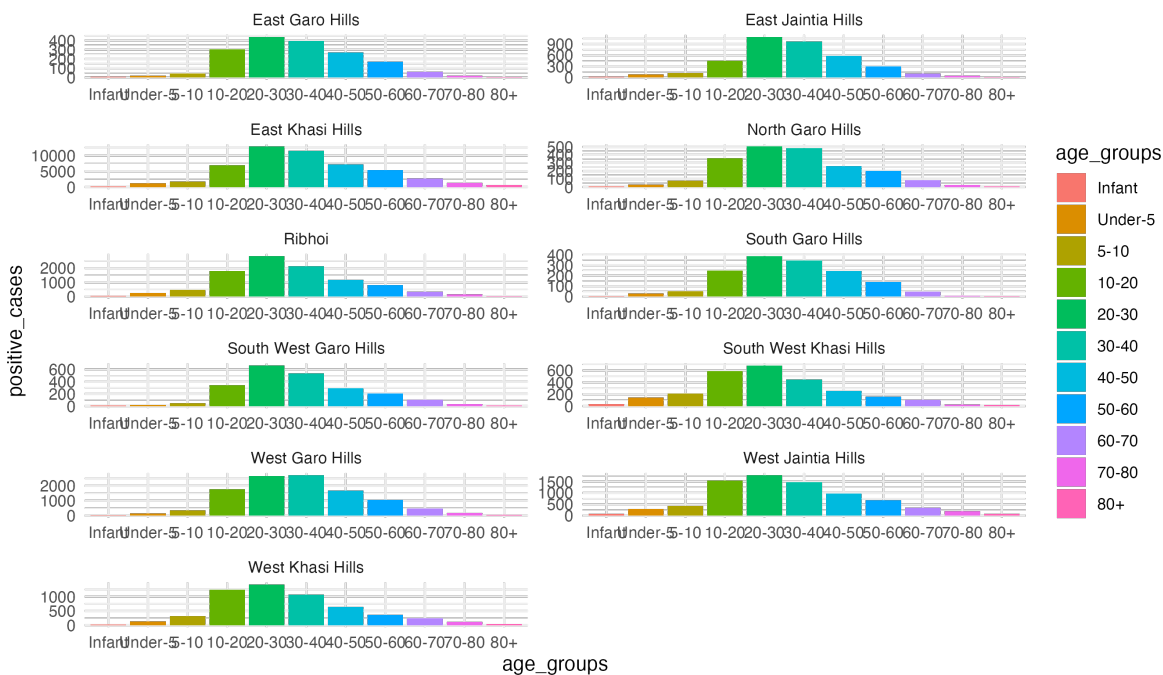
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
East Garo Hills	13	22	43	304	436	395	271	173	63	20	6	1,746
East Jaintia Hills	26	82	120	450	1,089	970	577	300	112	56	12	3,794
East Khasi Hills	226	1,148	1,787	6,920	12,816	11,520	7,199	5,341	2,777	1,355	649	51,738
North Garo Hills	11	34	79	359	504	485	259	202	88	28	11	2,060
Ribhoi	62	245	469	1,797	2,842	2,137	1,189	820	358	154	51	10,124
South Garo Hills	7	29	52	250	387	343	245	138	49	10	5	1,515
South West Garo Hills	9	16	47	345	664	536	294	207	103	36	12	2,269
South West Khasi Hills	36	146	210	586	682	447	255	161	105	27	17	2,672
West Garo Hills	44	141	355	1,778	2,653	2,728	1,691	1,072	489	168	57	11,176
West Jaintia Hills	83	298	424	1,558	1,784	1,470	975	675	355	188	74	7,884
West Khasi Hills	22	138	322	1,242	1,422	1,079	633	366	246	117	37	5,624

Note: Data includes only non-missing age groups.

### 5.4.24 Meghalaya

#### 5.4.24.1 District Wise Summary Table

#### 5.4.24.2 District wise Age Groups



## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

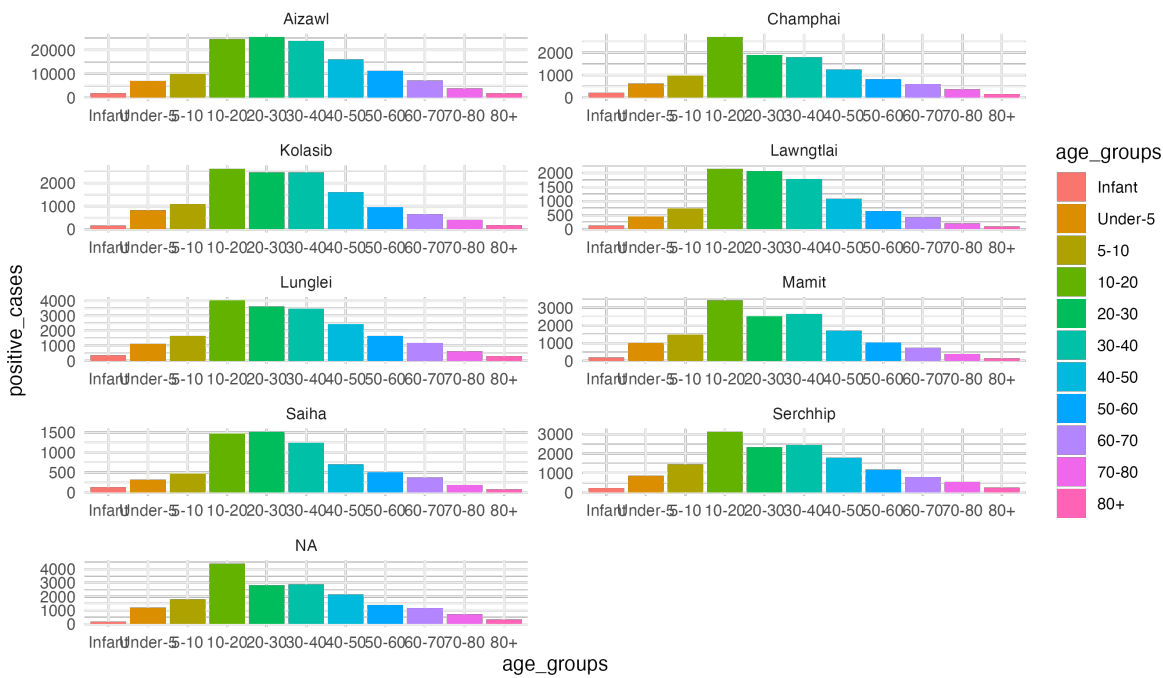
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Aizawl	1,762	7,063	9,914	24,581	25,408	23,741	15,970	11,209	7,182	3,816	1,833	132,479
Champhai	213	616	972	2,691	1,901	1,793	1,242	822	611	367	161	11,389
Kolasib	150	816	1,097	2,619	2,462	2,467	1,600	952	657	398	167	13,385
Lawngtlai	123	458	736	2,151	2,067	1,783	1,077	648	430	205	101	9,779
Lunglei	374	1,139	1,664	4,035	3,633	3,471	2,454	1,658	1,186	634	308	20,556
Mamit	182	1,009	1,471	3,416	2,516	2,654	1,695	1,024	727	390	147	15,231
Saiha	131	322	474	1,472	1,513	1,247	705	504	374	182	77	7,001
Serchhip	223	857	1,443	3,116	2,341	2,436	1,782	1,171	790	541	239	14,939
NA	169	1,182	1,786	4,402	2,830	2,878	2,144	1,372	1,164	711	319	18,957

Note: Data includes only non-missing age groups.

### 5.4.25 Mizoram

#### 5.4.25.1 District Wise Summary Table

#### 5.4.25.2 District wise Age Groups



## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

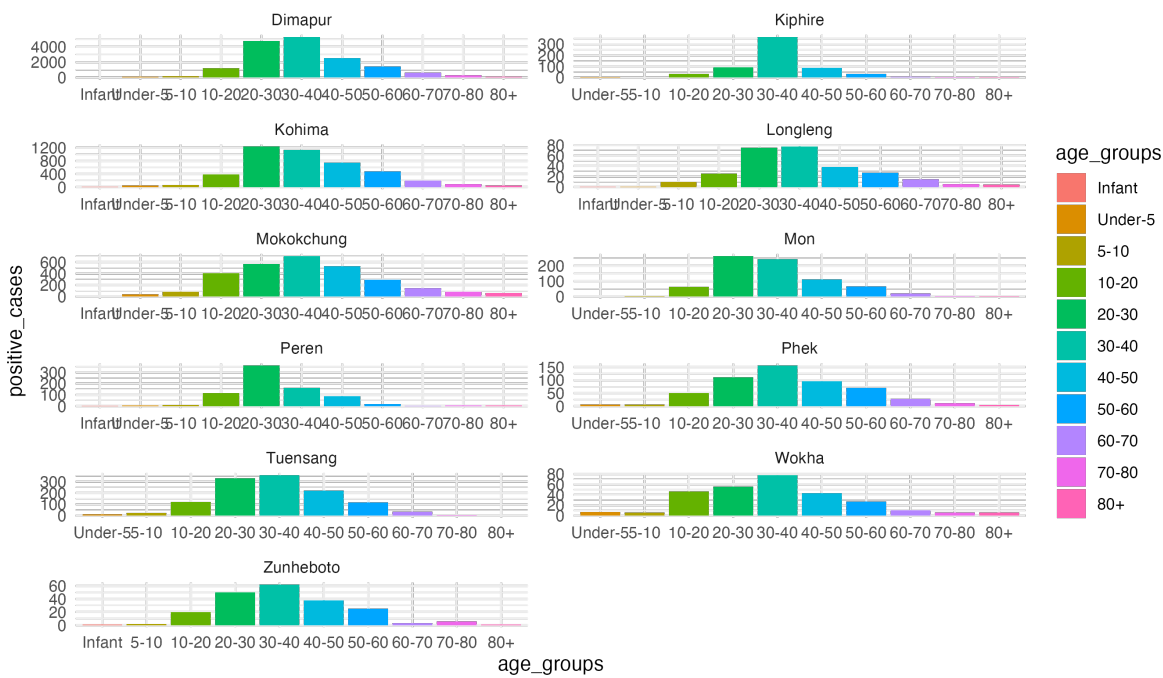
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Dimapur	21	133	189	1,234	4,779	5,288	2,517	1,420	639	293	125	16,638
Kiphire	NA	4	2	34	90	364	87	31	9	6	6	633
Kohima	15	49	67	384	1,238	1,134	748	476	197	90	53	4,451
Longleng	1	1	10	26	75	77	38	27	15	6	5	281
Mokokchung	4	42	87	409	569	704	526	292	143	89	63	2,928
Mon	NA	1	4	64	263	242	111	65	20	2	4	776
Peren	4	5	9	115	360	163	88	20	4	5	6	779
Phek	NA	6	7	50	111	156	96	71	29	12	5	543
Tuensang	NA	12	23	122	330	358	223	115	36	4	2	1,225
Wokha	NA	7	6	47	56	78	43	27	10	6	6	286
Zunheboto	1	NA	2	20	50	62	38	25	3	6	1	208

Note: Data includes only non-missing age groups.

### 5.4.26 Nagaland

#### 5.4.26.1 District Wise Summary Table

#### 5.4.26.2 District wise Age Groups



Positive Cases by District and Age Group  
COVID-19 data categorized by age groups and district

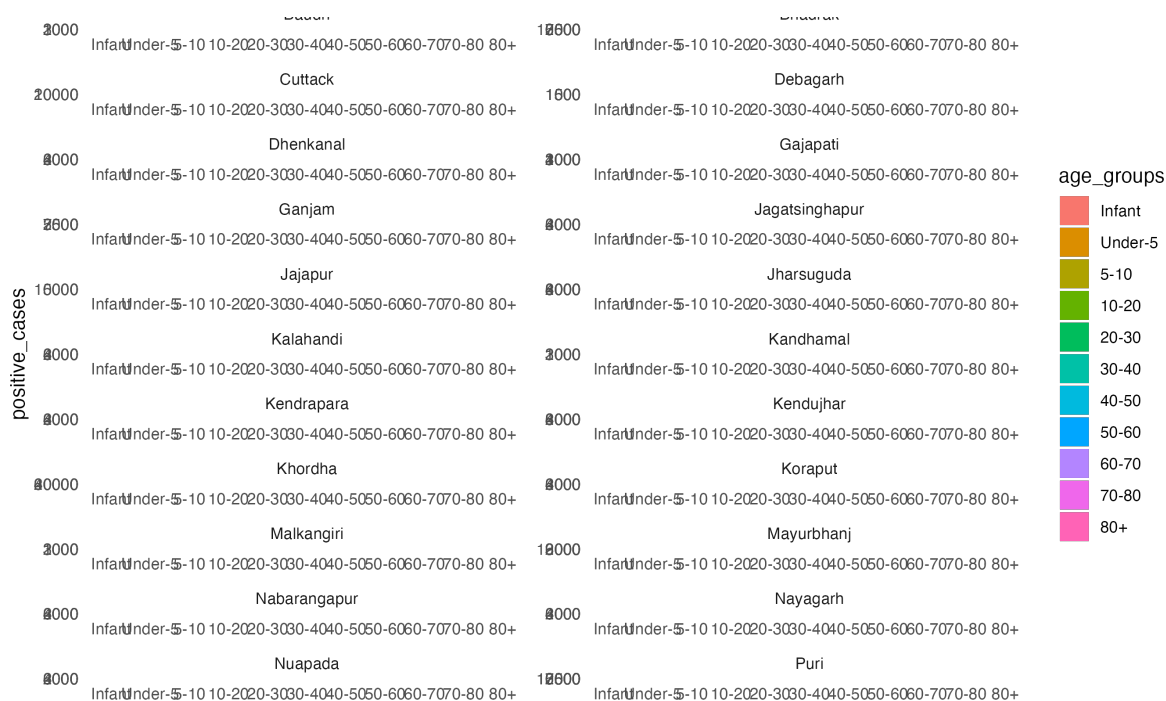
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Anugul	67	583	1,430	6,351	15,280	15,943	12,445	8,589	4,416	1,700	547	67,351
Balangir	74	335	690	3,478	8,348	8,445	6,884	5,126	3,223	1,036	330	37,969
Baleshwar	336	955	1,823	8,793	19,682	18,499	14,096	11,020	6,216	2,329	999	84,748
Bargarh	67	359	676	3,016	8,076	7,972	6,783	4,922	3,035	1,108	357	36,371
Baudh	25	180	384	2,063	3,492	3,265	2,458	1,690	1,025	349	113	15,044
Bhadrak	56	362	955	4,284	9,418	9,566	7,121	5,310	3,141	1,152	470	41,835
Cuttack	294	1,151	2,512	9,849	25,348	28,265	22,257	18,491	11,184	4,969	1,760	126,080
Debagarh	23	81	194	944	1,472	1,501	1,508	1,518	1,148	412	119	8,920
Dhenkanal	16	185	603	2,825	6,554	7,022	5,632	3,801	2,314	919	268	30,139
Gajapati	4	89	228	2,060	4,168	3,498	2,745	1,926	1,071	352	87	16,228
Ganjam	56	162	353	2,283	7,897	8,627	7,296	6,146	3,685	1,489	372	38,366
Jagatsinghapur	66	395	846	2,459	6,412	7,363	6,358	5,225	3,334	1,726	630	34,814
Jajapur	495	409	883	3,798	11,071	12,736	8,688	6,313	3,995	1,742	644	50,774
Jharsuguda	26	372	834	2,938	7,165	7,661	5,554	4,027	2,090	708	213	31,588
Kalahandi	12	206	573	3,015	7,057	7,014	5,827	4,312	2,494	717	187	31,414
Kandhamal	26	108	291	1,469	3,571	3,026	2,211	1,796	1,133	351	86	14,068
Kendrapara	51	289	593	2,289	5,533	6,090	4,561	3,612	2,566	1,144	516	27,244
Kendujhar	91	340	885	4,230	8,220	8,095	6,315	4,760	2,682	945	281	36,844
Khordha	410	2,412	5,129	22,069	68,661	72,324	55,699	48,544	28,757	12,845	4,634	321,484
Koraput	53	218	434	2,338	6,107	5,869	4,250	3,206	1,530	451	124	24,580
Malkangiri	53	157	370	1,649	3,421	2,985	2,027	1,394	789	260	52	13,157
Mayurbhanj	256	748	1,268	5,893	11,693	10,316	8,812	7,494	5,135	1,836	594	54,045
Nabarangapur	98	480	915	3,952	7,562	6,474	4,508	2,742	1,467	445	112	28,755
Nayagarh	58	274	759	3,152	5,920	6,250	5,035	4,153	2,983	1,422	522	30,528
Nuapada	48	350	767	3,162	7,011	6,054	4,719	3,362	2,157	603	178	28,411
Puri	123	447	898	3,968	10,986	11,966	9,482	7,518	4,576	2,100	711	52,775
Rayagada	41	232	696	3,205	6,571	5,661	4,418	3,003	1,714	550	119	26,210
Sambalpur	34	244	663	4,293	12,684	12,279	9,373	6,886	4,136	1,494	453	52,539
Subarnapur	25	209	509	1,983	3,834	3,681	2,862	2,005	1,621	694	235	17,658
Sundargarh	151	1,133	2,584	14,262	25,138	24,572	20,655	16,306	8,611	3,231	1,043	117,686

Note: Data includes only non-missing age groups.

## 5.4.27 Odisha

### 5.4.27.1 District Wise Summary Table

### 5.4.27.2 District wise Age Groups



## Positive Cases by District and Age Group

### COVID-19 data categorized by age groups and district

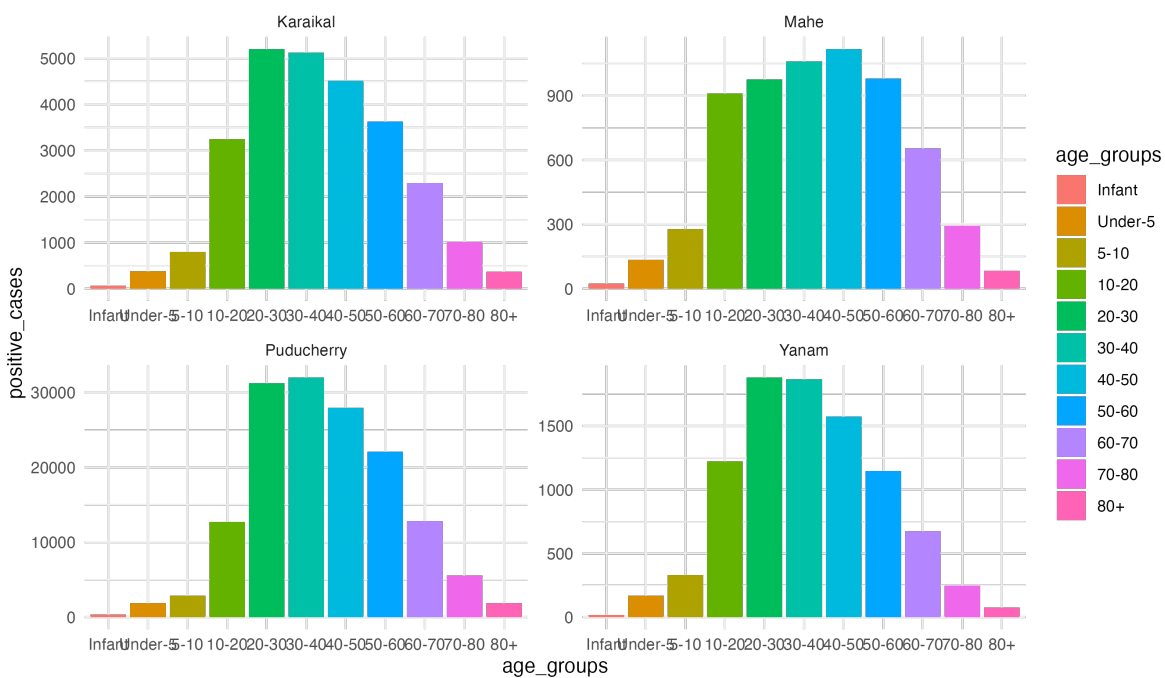
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Karaikal	60	377	795	3,250	5,204	5,127	4,513	3,630	2,297	1,019	368	26,640
Mahe	24	135	278	910	977	1,059	1,118	980	655	294	82	6,512
Puducherry	363	1,874	2,848	12,691	31,219	31,979	27,953	22,098	12,849	5,581	1,852	151,307
Yanam	17	167	329	1,221	1,880	1,866	1,574	1,148	676	247	76	9,201

Note: Data includes only non-missing age groups.

## 5.4.28 Puducherry

### 5.4.28.1 District Wise Summary Table

### 5.4.28.2 District wise Age Groups



## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Amritsar	82	443	893	4,800	16,174	17,648	14,385	12,649	9,479	5,160	1,507	83,220
Barnala	9	35	65	417	1,934	1,848	1,394	1,170	877	423	107	8,279
Bathinda	50	259	490	3,399	13,851	14,953	9,725	8,006	6,396	2,956	766	60,851
Faridkot	31	144	271	1,809	5,061	5,234	3,986	3,207	2,508	1,046	316	23,613
Fatehgarh Sahib	18	73	129	1,151	3,134	3,290	2,548	2,139	1,530	760	192	14,964
Fazilka	26	107	212	1,699	5,423	6,254	4,441	3,693	2,364	904	259	25,382
Firozpur	30	109	175	1,277	4,381	4,909	3,163	2,780	1,872	841	242	19,779
Gurdaspur	24	124	239	1,773	5,763	6,179	4,923	4,215	3,078	1,431	446	28,195
Hoshiarpur	60	301	652	3,842	9,308	10,295	8,154	7,209	5,498	3,002	1,064	49,385
Jalandhar	140	555	998	6,327	17,815	20,987	17,044	16,121	12,462	6,456	2,173	101,078
Kapurthala	31	123	264	1,701	4,523	4,971	3,923	4,024	2,580	1,216	400	23,756
Ludhiana	169	754	1,406	7,971	24,945	28,584	23,008	20,230	14,890	7,509	2,549	132,015
Mansa	9	67	145	988	3,695	5,035	3,086	2,796	1,868	794	243	18,726
Moga	16	66	146	956	3,092	3,297	2,486	2,317	1,805	829	247	15,257
Pathankot	52	195	361	1,893	5,418	6,311	4,518	4,068	2,787	1,437	470	27,510
Patiala	86	383	742	4,748	14,696	16,311	12,601	11,952	8,289	4,114	1,344	75,266
Rupnagar	26	112	290	1,920	4,734	5,271	3,950	3,896	2,416	1,125	426	24,166
Sahibzada Ajit Singh Nag*	226	874	1,589	6,892	25,126	31,463	19,686	15,348	11,517	5,833	2,071	120,625
Sangrur	26	110	223	1,285	4,801	5,099	3,800	3,588	2,500	1,266	328	23,026
Shahid Bhagat Singh Nagar	8	53	146	1,711	2,832	3,275	2,750	2,584	1,784	833	290	16,266
Sri Muktsar Sahib	17	94	232	1,668	5,332	5,895	4,611	3,857	2,771	1,199	374	26,050
Tarn Taran	9	57	98	675	2,510	2,547	1,811	1,498	1,005	442	123	10,775

Note: Data includes only non-missing age groups.

### 5.4.29 Punjab

#### 5.4.29.1 District Wise Summary Table

#### 5.4.29.2 District wise Age Groups



Positive Cases by District and Age Group  
COVID-19 data categorized by age groups and district

District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Ajmer	45	167	330	1,670	5,051	5,014	3,988	3,842	2,857	1,266	473	24,703
Alwar	162	1,241	2,217	11,445	27,639	26,905	16,757	12,589	8,600	3,763	1,078	112,396
Banswara	12	65	171	1,396	3,229	3,054	2,133	1,632	1,058	341	75	13,166
Baran	9	112	257	2,140	5,270	4,885	3,483	2,465	1,530	547	195	20,893
Barmer	14	73	225	1,627	3,113	2,671	1,603	1,104	805	381	176	11,792
Bharatpur	51	302	734	4,689	12,467	11,232	7,842	6,272	4,363	1,771	487	50,210
Bhilwara	46	322	661	4,194	11,596	11,709	9,067	6,994	4,189	1,738	529	51,045
Bikaner	81	492	1,155	6,881	18,011	18,242	13,414	10,627	7,526	3,321	1,056	80,806
Bundi	20	102	160	1,082	2,329	2,067	1,508	1,259	933	343	115	9,918
Chittaurgarh	36	322	667	3,636	9,844	10,080	7,430	5,113	2,984	1,192	350	41,654
Churu	28	235	494	2,878	6,871	6,698	4,538	3,336	2,312	996	251	28,637
Dausa	40	137	271	1,910	4,947	4,337	3,117	2,099	1,400	520	172	18,950
Dhaulpur	17	75	153	875	2,200	1,778	1,283	934	628	209	64	8,216
Dungarpur	9	152	491	4,316	5,269	5,227	4,086	2,990	2,083	626	157	25,406
Ganganagar	10	78	165	837	2,948	3,270	2,244	2,143	1,444	601	188	13,928
Hanumangarh	18	143	351	3,046	6,461	6,506	4,306	3,319	2,164	795	261	27,370
Jaipur	543	2,790	5,357	26,164	81,315	85,060	60,768	53,574	37,521	17,303	5,864	376,259
Jaisalmer	7	89	275	2,023	4,210	3,919	2,431	1,357	1,024	355	102	15,792
Jalor	8	82	185	1,094	2,870	2,551	1,682	1,230	763	277	79	10,821
Jhalawar	28	185	388	1,951	6,468	5,895	4,627	3,375	2,109	785	277	26,088
Jhunjhunun	14	47	72	340	1,557	1,589	912	808	645	367	87	6,438
Jodhpur	190	998	2,411	14,415	41,784	37,669	26,548	20,517	14,141	6,145	1,964	166,782
Karauli	17	62	116	561	1,769	1,513	977	794	598	246	76	6,729
Kota	44	430	1,080	9,648	24,603	24,267	17,967	13,592	8,355	3,045	937	103,968
Nagaur	18	174	336	1,757	4,833	4,501	3,340	2,622	1,957	925	241	20,704
Pali	62	354	849	4,895	10,939	9,912	7,558	6,081	4,124	1,700	558	47,032
Pratapgarh	21	63	194	1,593	1,674	1,513	1,030	708	497	148	58	7,499
Rajsamand	18	203	502	2,758	7,127	7,074	4,981	3,278	2,009	844	300	29,094
Sawai Madhopur	43	170	397	2,406	5,768	5,591	4,140	3,161	2,070	790	215	24,751
Sikar	62	232	401	2,021	5,887	5,850	3,747	2,779	1,923	888	278	24,068
Sirohi	20	140	320	2,067	5,026	5,048	3,786	2,783	1,572	637	258	21,657
Tonk	33	97	179	1,287	2,783	2,692	1,855	1,623	1,106	441	107	12,203
Udaipur	121	708	1,724	9,802	29,437	29,421	21,671	16,858	11,487	5,066	1,583	127,878

Note: Data includes only non-missing age groups.

## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

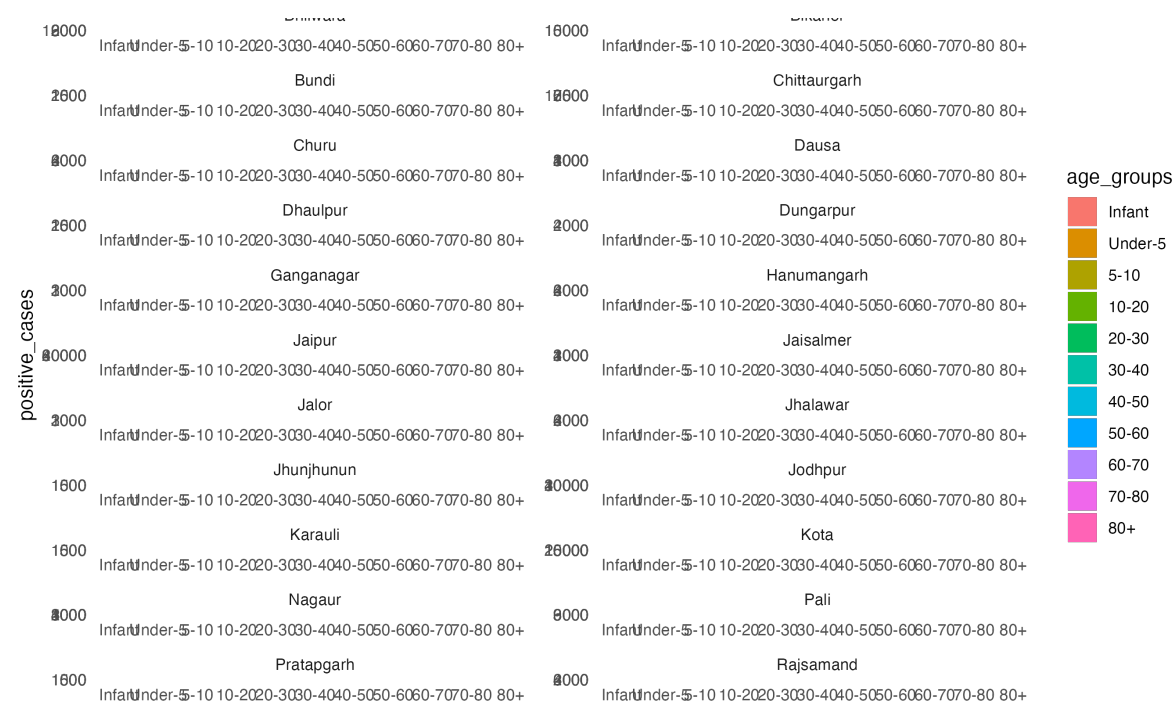
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
East District	62	182	282	1,485	4,423	4,236	2,697	1,858	1,005	535	190	16,955
North District	3	3	3	31	94	74	44	42	9	6	5	314
South District	18	49	107	662	1,043	891	514	294	179	93	45	3,895
West District	11	18	37	150	351	301	149	136	83	49	14	1,299

Note: Data includes only non-missing age groups.

### 5.4.30 Rajasthan

#### 5.4.30.1 District Wise Summary Table

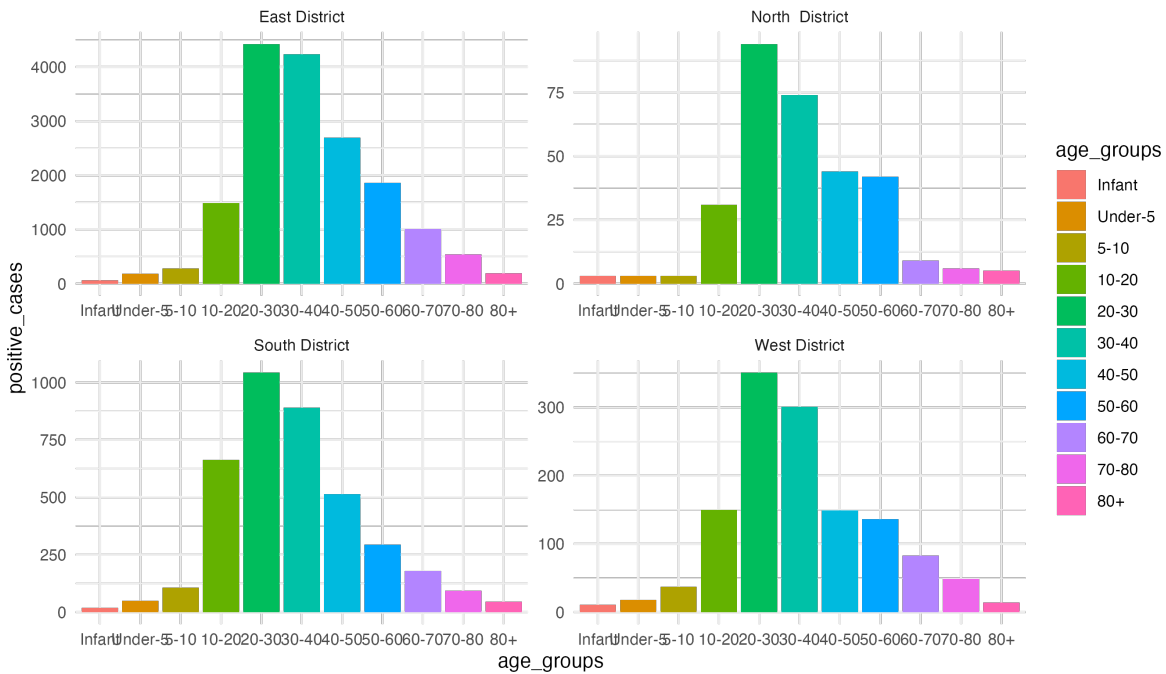
#### 5.4.30.2 District wise Age Groups



## 5.4.31 Sikkim

### 5.4.31.1 District Wise Summary Table

### 5.4.31.2 District wise Age Groups



## 5.4.32 Tamil Nadu

### 5.4.32.1 District Wise Summary Table

### 5.4.32.2 District wise Age Groups



Positive Cases by District and Age Group  
COVID-19 data categorized by age groups and district

District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Ariyalur	19	148	330	1,741	4,180	4,363	3,563	3,085	2,051	908	246	20,634
Chennai	1,122	7,115	12,962	57,402	172,435	191,664	155,664	140,511	96,363	51,575	20,936	907,749
Coimbatore	465	3,349	5,651	23,916	66,086	69,598	59,154	53,869	37,820	19,753	7,035	346,696
Cuddalore	109	636	969	5,246	16,416	16,175	14,374	13,718	7,673	3,303	976	79,595
Dharmapuri	32	226	400	2,670	8,181	8,265	6,890	5,560	3,262	1,312	371	37,169
Dindigul	53	273	515	2,350	6,640	7,314	7,293	7,298	5,399	2,617	728	40,480
Erode	131	1,075	2,368	10,842	24,534	26,206	24,263	22,100	16,309	8,137	2,716	138,681
Kancheepuram	92	644	1,150	5,064	20,153	21,233	14,418	11,354	8,339	4,260	1,390	88,097
Kanniyakumari	137	1,067	1,851	7,325	15,752	15,976	14,736	14,256	11,444	6,151	1,913	90,608
Karur	44	176	270	1,496	4,980	5,692	5,399	5,185	3,916	2,003	652	29,813
Krishnagiri	33	260	540	4,055	16,568	15,294	11,193	8,143	4,621	1,747	532	62,986
Madurai	123	681	1,141	5,066	15,867	17,756	16,185	16,637	14,988	7,269	2,280	97,993
Nagapattinam	83	388	904	4,236	10,735	10,741	9,387	8,441	5,669	2,560	732	53,876
Namakkal	52	309	633	3,756	10,321	12,504	12,647	12,369	8,894	3,888	1,155	66,528
Perambalur	30	134	223	1,088	3,060	2,859	2,333	2,002	1,287	614	188	13,818
Pudukkottai	58	342	589	2,556	6,733	6,482	5,871	5,706	3,776	1,676	472	34,261
Ramanathapuram	39	164	300	1,363	4,758	4,763	4,130	3,954	2,801	1,118	246	23,636
Salem	164	836	1,548	8,077	22,922	26,729	23,818	21,713	14,469	6,863	2,058	129,197
Sivaganga	47	206	321	1,614	4,207	4,332	4,077	4,417	3,629	1,949	543	25,342
Thanjavur	91	774	1,566	7,282	16,486	17,756	16,052	15,521	11,097	5,066	1,456	93,147
The Nilgiris	53	428	1,012	5,132	8,328	8,520	8,425	7,360	4,465	1,912	556	46,191
Theni	71	487	941	3,925	9,851	9,902	8,962	8,127	5,643	2,441	620	50,970
Thiruvallur	126	1,015	1,883	9,347	31,693	35,075	27,269	21,990	14,215	6,164	1,928	150,705
Thiruvavur	65	439	789	3,940	9,061	9,588	8,638	8,096	5,369	2,302	659	48,946
Thoothukkudi	69	549	1,060	4,908	14,285	13,185	10,865	10,380	6,901	3,438	939	66,579
Tiruchirappalli	132	746	1,494	6,412	18,191	19,816	16,981	16,677	12,813	6,346	2,056	101,664
Tirunelveli	85	669	1,103	4,747	12,942	13,057	11,275	10,506	7,714	4,069	1,230	67,397
Tiruppur	100	924	1,774	10,144	25,663	25,522	23,323	20,459	13,280	6,472	1,997	129,658
Tiruvannamalai	114	710	1,151	5,512	14,330	14,733	12,432	10,272	7,035	3,133	886	70,308
Vellore	139	857	1,448	6,519	18,869	20,577	18,008	16,346	11,597	5,179	1,473	101,012
Viluppuram	69	348	591	3,872	12,394	12,956	10,480	8,730	6,207	2,505	731	58,883
Virudhunagar	56	345	677	4,113	11,659	11,941	10,545	9,810	6,659	3,007	737	59,549
NA	354	2,607	4,770	22,115	70,378	74,464	55,211	44,521	29,930	13,544	4,130	322,024

Note: Data includes only non-missing age groups.

Positive Cases by District and Age Group  
COVID-19 data categorized by age groups and district

District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Adilabad	5	93	67	191	652	583	370	358	176	77	21	2,593
Bhadradi Kothagudem	1	8	7	21	176	213	159	128	56	22	8	799
Hyderabad	143	1,061	1,704	5,254	19,054	24,418	17,730	15,640	10,074	4,646	1,357	101,081
Jagitial	NA	3	9	22	104	125	60	52	25	10	NA	410
Jangoan	NA	7	8	17	114	97	71	70	26	11	5	426
Jayashankar	1	9	12	38	182	224	98	70	23	8	4	669
Jogulamba Gadwal	1	3	3	18	85	98	68	68	39	18	5	406
Kamareddy	1	14	14	72	131	147	102	82	45	13	4	625
Karimnagar	1	42	59	143	571	719	519	516	312	153	44	3,079
Khammam	58	178	250	789	2,652	2,941	2,284	1,819	1,084	431	137	12,623
Kumuram Bheem Asifabad	1	7	5	6	35	32	29	15	4	4	NA	138
Mahabubabad	2	19	25	58	252	348	206	148	59	23	9	1,149
Mahabubnagar	1	13	24	70	334	341	231	203	152	64	18	1,451
Mancherial	NA	7	4	22	112	117	87	89	47	10	3	498
Medak	4	24	48	88	303	456	314	238	154	66	15	1,710
Medchal Malkajgiri	9	75	115	389	1,553	1,966	1,388	1,000	515	188	35	7,233
Mulugu	1	4	6	21	90	97	42	32	11	4	3	311
Nagarkurnool	1	9	18	53	106	132	96	79	33	10	5	542
Nalgonda	4	30	33	103	421	512	354	335	202	100	23	2,117
Narayanpet	3	1	NA	9	22	24	16	12	10	2	NA	99
Nirmal	NA	6	6	13	41	40	21	15	13	8	1	164
Nizamabad	1	13	19	79	307	393	341	310	210	99	16	1,788
Peddapalli	1	6	9	22	119	126	98	86	41	14	3	525
Rajanna Sircilla	1	10	11	48	92	120	82	57	44	9	6	480
Ranga Reddy	19	171	335	871	3,024	4,429	2,924	2,378	1,375	604	143	16,273
Sangareddy	6	29	64	181	650	798	572	416	241	123	28	3,108
Siddipet	2	10	13	51	305	274	191	147	85	35	11	1,124
Suryapet	2	11	18	42	245	328	258	268	126	39	12	1,349
Vikarabad	1	7	4	38	190	186	139	116	62	23	5	771
Wanaparthy	1	4	6	7	60	61	36	28	26	13	1	243
Warangal Rural	3	20	25	105	487	566	422	360	204	87	22	2,301
Warangal Urban	4	22	51	168	744	957	763	786	447	234	72	4,248
Yadadri Bhuvanagiri	NA	2	5	27	69	72	65	45	20	7	NA	312

Note: Data includes only non-missing age groups.

## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Dhalai	1	76	182	929	2,062	2,626	1,652	1,025	436	153	85	9,227
Gomati	23	102	272	1,108	2,080	2,565	1,717	1,127	632	281	96	10,003
Khowai	14	33	56	374	873	1,184	856	540	286	110	43	4,369
North Tripura	5	72	141	927	2,159	2,456	1,410	995	562	283	102	9,112
Sipahijala	10	72	162	930	1,939	1,921	1,165	768	461	179	102	7,709
South Tripura	48	199	365	1,608	2,758	2,955	2,028	1,529	922	372	141	12,925
Unokoti	6	61	166	1,055	1,966	2,326	1,614	1,222	696	309	123	9,544
West Tripura	55	220	430	2,579	7,749	8,593	6,625	5,226	3,197	1,397	577	36,648

Note: Data includes only non-missing age groups.

### 5.4.33 Telangana

#### 5.4.33.1 District Wise Summary Table

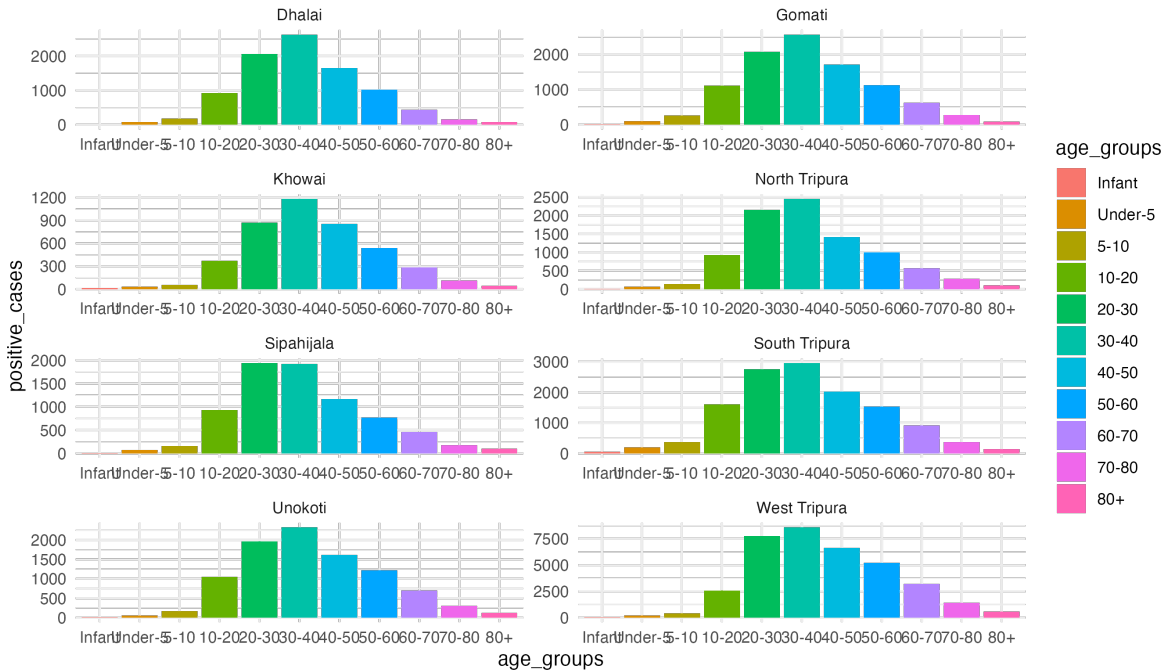
#### 5.4.33.2 District wise Age Groups



## 5.4.34 Tripura

### 5.4.34.1 District Wise Summary Table

### 5.4.34.2 District wise Age Groups



## 5.4.35 Uttar Pradesh

### 5.4.35.1 District Wise Summary Table

### 5.4.35.2 District wise Age Groups



Positive Cases by District and Age Group  
 COVID-19 data categorized by age groups and district

District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Agra	43	286	544	3,459	11,798	10,841	7,480	7,490	5,322	2,491	736	50,490
Aligarh	19	309	540	2,634	7,361	6,644	4,715	4,228	3,037	1,221	396	31,104
Ambedkar Nagar	2	54	100	849	2,294	2,179	1,514	1,174	723	317	73	9,279
Amethi	1	74	118	882	3,363	3,284	2,253	1,777	856	344	76	13,028
Amroha	7	170	275	1,969	6,508	5,877	3,948	2,871	1,664	546	138	23,973
Auraiya	3	94	215	1,145	3,254	2,741	2,237	1,703	894	384	136	12,806
Azamgarh	8	227	385	2,013	6,015	6,107	4,553	3,552	2,475	1,107	328	26,770
Baghpat	13	78	231	1,367	3,358	3,075	2,204	1,649	1,156	560	194	13,885
Bahraich	4	132	303	1,448	3,883	3,869	2,597	2,023	1,319	488	124	16,190
Ballia	6	232	448	2,442	6,476	5,863	4,419	3,508	2,414	1,015	307	27,130
Balrampur	2	28	117	857	2,449	2,452	1,742	1,151	698	232	58	9,786
Banda	2	113	223	1,210	3,806	3,497	2,515	2,000	1,306	566	223	15,461
Bara Banki	3	209	364	2,012	7,161	6,156	4,386	3,211	2,002	730	202	26,436
Bareilly	30	393	730	3,933	13,831	14,459	10,860	9,623	6,299	2,609	772	63,539
Basti	4	89	188	1,303	3,947	3,695	2,667	2,042	1,369	543	149	15,996
Bhadohi	1	114	204	1,082	2,775	2,752	1,924	1,303	872	288	60	11,375
Bijnor	16	163	261	1,451	5,622	5,349	3,729	3,267	1,711	701	175	22,445
Budaun	1	118	261	1,656	5,015	4,943	3,792	2,471	1,388	429	110	20,184
Bulandshahr	33	196	417	2,354	5,769	5,142	3,481	2,957	2,068	972	342	23,731
Chandauli	2	133	305	1,632	5,169	4,893	3,302	2,792	1,653	645	221	20,747
Chitrakoot	1	34	123	868	2,489	2,186	1,438	1,102	651	241	108	9,241
Deoria	6	199	356	2,171	5,835	5,666	4,178	3,253	2,212	972	270	25,118
Etah	7	97	193	1,032	3,409	3,293	2,253	1,683	926	336	155	13,384
Etawah	10	251	401	1,640	4,665	4,038	2,700	2,075	1,320	588	214	17,902
Faizabad	2	134	263	1,701	5,207	5,468	3,801	2,974	1,968	832	261	22,611
Farrukhabad	2	101	217	1,170	3,023	3,023	2,226	1,866	1,213	540	149	13,530
Fatehpur	8	56	155	886	2,787	2,394	1,648	1,390	832	351	104	10,611
Firozabad	15	77	199	1,187	3,446	2,868	1,934	1,698	1,262	504	134	13,324
Gautam Buddha Nagar	259	2,464	4,959	15,724	50,411	74,146	44,155	31,687	23,199	11,514	4,509	263,027
Ghaziabad	319	2,128	4,331	15,925	45,045	62,605	45,720	34,471	22,063	10,350	3,151	246,108
Ghazipur	8	155	307	2,074	6,679	6,631	4,901	3,586	2,228	973	266	27,808
Gonda	18	179	263	1,648	4,863	4,702	3,129	2,546	1,625	663	196	19,832
Gorakhpur	27	476	960	5,039	16,237	17,322	12,705	10,480	6,893	3,049	972	74,160
Hamirpur	2	50	123	748	2,121	1,767	1,216	1,008	655	289	106	8,085
Hapur	14	104	238	1,386	4,213	4,199	3,255	2,577	1,570	750	215	18,521
Hardoi	9	181	336	1,727	5,353	4,472	3,518	2,425	1,629	641	154	20,445
Hathras	4	35	85	490	1,188	976	794	713	542	228	79	5,134
Jalaun	3	123	227	1,179	4,280	3,632	2,494	2,136	1,505	665	199	16,443
Jaunpur	6	211	412	2,164	6,500	6,531	5,001	3,527	2,483	992	281	28,108
Jhansi	5	411	677	3,380	11,786	11,884	8,845	7,733	4,825	1,908	573	52,027
Kannauj	3	81	199	1,198	3,349	2,858	2,155	1,640	1,153	395	154	13,185
Kanpur Dehat	5	68	121	638	2,172	1,945	1,383	1,189	786	289	93	8,689
Kanpur Nagar	53	686	1,094	5,799	19,537	20,725	16,095	15,844	11,273	5,335	1,578	98,019
Kasganj	2	67	106	822	1,595	1,297	1,034	777	560	227	55	6,542
Kaushambi	7	44	69	615	1,532	1,259	981	645	421	180	57	5,810
Kheri	2	289	594	3,175	8,578	8,056	5,547	4,109	2,293	854	208	33,705
Kushinagar	3	138	311	1,464	4,026	3,866	3,287	2,819	2,194	771	204	19,083
Lalitpur	1	140	354	1,862	4,471	3,892	2,874	2,133	1,235	467	170	17,599
Lucknow	98	2,747	4,861	20,336	66,385	72,651	52,989	48,566	32,188	14,302	4,949	320,072
Mahoba	NA	51	58	428	1,216	1,152	863	698	479	210	87	5,242
Mahrajganj	6	120	232	1,526	4,225	3,971	3,163	2,221	1,460	472	107	17,503
Mainpuri	3	83	215	1,170	3,486	2,802	2,023	1,601	997	385	152	12,917
Mathura	17	204	478	2,581	7,797	7,295	5,053	4,611	3,493	1,699	466	33,694
Mau	1	89	167	929	2,570	2,590	1,834	1,468	986	399	143	11,176
Meerut	54	1,007	1,817	7,994	22,309	24,014	17,631	15,167	10,016	4,638	1,402	106,049
Mirzapur	5	98	188	1,124	3,350	3,636	2,613	1,919	1,244	501	162	14,840
Moradabad	28	384	812	4,608	12,780	12,867	9,164	8,090	4,917	2,026	570	56,246
Muzaffarnagar	18	362	674	3,452	10,464	9,897	7,645	6,672	3,986	1,617	472	45,259
Pilibhit	2	132	268	1,490	4,873	4,044	2,893	2,024	1,233	401	96	17,456
Pratapgarh	6	126	296	1,345	4,817	4,500	3,400	2,405	1,683	666	243	19,487
Prayagraj	63	723	1,264	6,278	21,869	22,132	17,203	15,883	10,716	4,821	1,624	102,576
Rae Bareli	5	145	273	1,856	6,040	5,984	4,271	3,220	1,970	728	268	24,760
Rampur	14	150	294	1,674	4,589	4,224	2,927	2,274	1,321	459	120	18,046
Saharanpur	37	318	656	3,906	11,441	11,018	8,032	6,760	4,364	1,750	502	48,784
Sambhal	4	73	228	1,435	3,216	2,761	2,038	1,695	881	389	95	12,815
Sant Kabir Nagar	3	57	159	1,077	2,579	2,401	1,654	1,115	809	293	78	10,225
Shahjahanpur	NA	192	504	2,540	7,325	6,660	5,179	4,171	2,525	967	291	30,354
Shamli	9	175	435	2,093	4,331	3,947	3,041	2,227	1,476	557	210	18,501
Shrawasti	NA	43	94	481	1,454	1,353	1,154	742	414	124	34	5,893
Siddharthnagar	3	152	307	1,457	3,012	2,887	2,057	1,522	1,057	311	98	12,863
Sitapur	7	135	247	1,232	4,643	4,076	3,050	2,423	1,462	543	193	18,011
Sonbhadra	NA	103	171	1,276	5,225	4,899	3,789	3,052	1,090	404	142	20,151
Sultanpur	2	130	265	1,415	4,675	4,436	3,284	2,223	1,492	534	166	18,622
Unnao	2	183	346	1,617	4,760	4,243	3,232	2,519	1,633	671	176	19,382
Varanasi	51	1,054	1,791	7,993	25,762	26,170	19,722	16,740	11,389	5,354	1,570	117,596

Note: Data includes only non-missing age groups.

## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

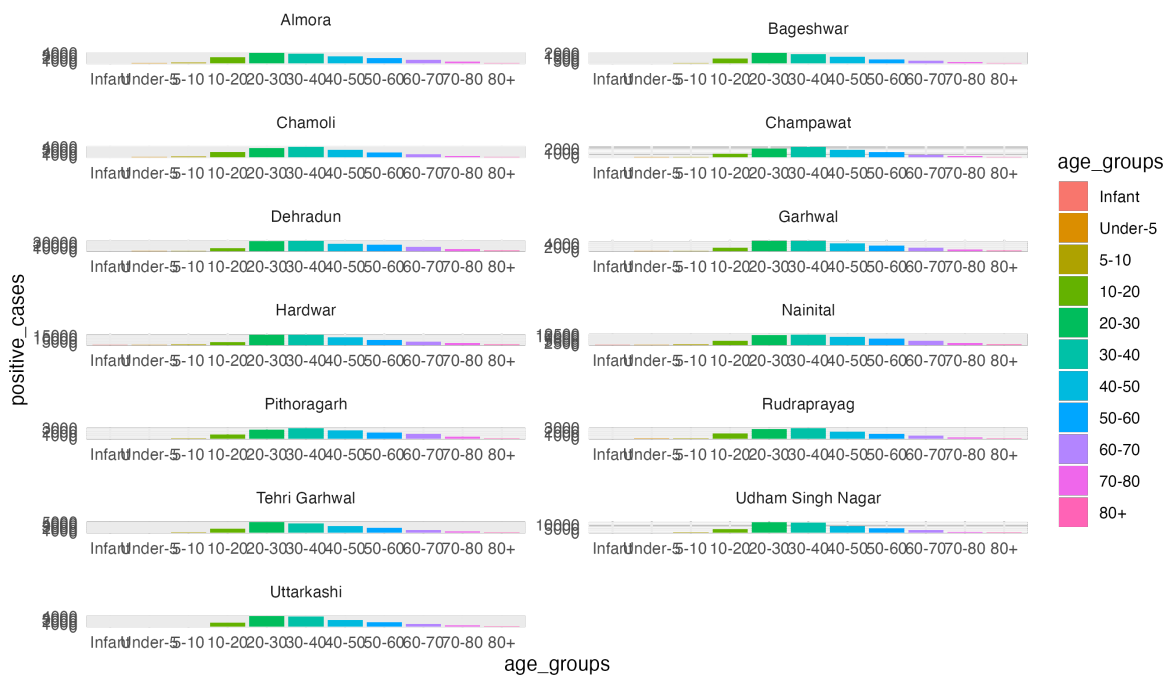
District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Almora	38	157	352	2,376	3,973	3,666	2,686	2,031	1,275	662	210	17,426
Bageshwar	12	43	87	955	2,081	1,863	1,304	820	566	257	55	8,043
Chamoli	25	88	287	1,888	3,379	3,959	2,758	1,786	1,061	451	150	15,832
Champawat	10	69	84	861	2,109	2,563	1,825	1,229	685	289	89	9,813
Dehradun	298	915	1,767	10,653	34,764	36,797	26,471	22,293	14,368	7,273	2,720	158,319
Garhwal	43	119	237	1,862	5,703	5,564	4,192	3,139	1,859	949	321	23,988
Hardwar	85	392	803	4,654	15,888	16,303	12,091	8,118	5,361	2,850	746	67,291
Nainital	112	409	774	4,867	11,468	12,185	9,457	7,636	4,646	2,132	701	54,387
Pithoragarh	24	55	129	1,280	2,602	2,929	2,337	1,796	1,338	661	227	13,378
Rudraprayag	50	136	207	1,521	2,691	2,863	1,957	1,396	909	429	104	12,263
Tehri Garhwal	3	77	266	1,917	4,849	4,321	3,087	2,226	1,229	611	191	18,777
Udham Singh Nagar	28	276	623	4,503	12,871	12,382	8,228	5,746	3,217	1,210	404	49,488
Uttarkashi	1	40	72	1,541	4,155	3,990	2,690	1,718	995	463	110	15,775

Note: Data includes only non-missing age groups.

### 5.4.36 Uttarakhand

#### 5.4.36.1 District Wise Summary Table

#### 5.4.36.2 District wise Age Groups



## Positive Cases by District and Age Group COVID-19 data categorized by age groups and district

District	Infant	Under-5	5-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80+	Total Cases
Alipurduar	41	196	323	1,742	4,537	4,555	3,204	3,017	2,040	767	226	20,648
Bankura	66	232	504	3,106	10,006	9,294	7,442	6,481	4,609	1,919	568	44,227
Birbhum	81	515	987	5,214	13,553	13,798	11,478	9,481	6,577	2,061	586	64,331
Cooch Behar	71	243	398	2,767	7,178	7,887	6,244	5,326	3,640	1,517	470	35,741
Dakshin Dinajpur	95	159	362	2,052	5,747	5,468	4,282	3,357	2,360	906	248	25,036
Darjiling	143	582	1,004	4,921	14,292	16,848	13,435	11,549	8,351	4,131	1,407	76,663
Hooghly	187	724	1,274	5,308	15,421	21,013	19,124	18,739	14,728	7,807	2,694	107,019
Howrah	170	1,405	1,633	6,464	18,524	24,508	22,553	22,094	17,021	8,679	3,033	126,084
Jalpaiguri	138	443	753	3,708	10,647	11,213	9,018	7,828	6,083	2,850	883	53,564
Jhargram	83	156	242	1,259	3,512	3,745	3,445	2,579	1,821	815	328	17,985
Kalimpong	28	95	133	828	1,861	1,979	1,410	1,164	706	375	179	8,758
Kolkata	1,249	7,132	8,897	30,655	82,435	114,133	104,958	107,407	90,150	53,515	20,693	621,224
Maldah	135	399	662	3,441	11,625	10,969	7,892	6,379	4,117	1,444	374	47,437
Medinipur West	172	832	1,296	5,681	17,856	17,806	14,736	12,924	7,942	3,485	1,245	83,975
Murshidabad	42	260	461	2,722	9,158	9,689	7,199	6,277	4,270	1,741	491	42,310
Nadia	160	780	1,246	5,158	15,187	18,275	15,432	13,707	9,323	3,966	1,273	84,507
North Twenty Four Pargan*	858	4,414	5,722	20,492	57,309	78,844	70,626	66,808	50,078	25,545	8,444	389,140
Paschim Bardhaman	128	539	966	3,924	13,399	17,159	15,073	15,723	9,290	5,349	1,791	83,341
Purba Bardhaman	203	460	849	3,941	11,279	11,992	10,076	9,066	6,526	2,741	832	57,965
Purba Medinipur	75	457	1,017	4,872	13,051	15,304	13,068	10,430	6,105	2,785	988	68,152
Puruliya	33	194	368	2,152	6,032	5,846	4,421	3,642	2,359	883	263	26,193
South Twenty Four Pargan*	221	993	1,774	8,050	22,298	27,262	23,909	20,721	14,490	7,081	2,480	129,279
Uttar Dinajpur	19	197	324	2,243	6,184	6,253	4,300	3,465	2,341	846	213	26,385

Note: Data includes only non-missing age groups.

### 5.4.37 West Bengal

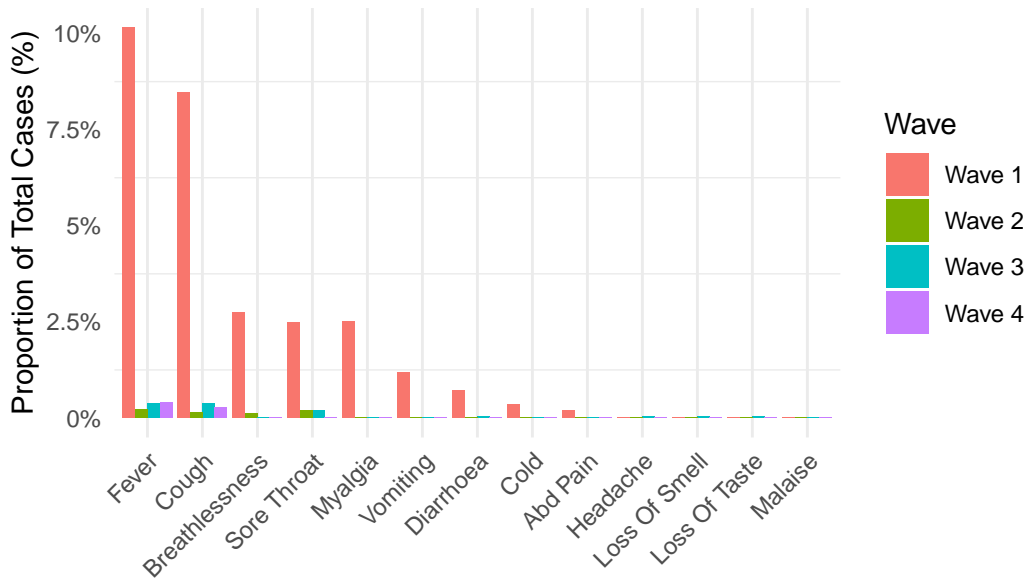
#### 5.4.37.1 District Wise Summary Table

#### 5.4.37.2 District wise Age Groups

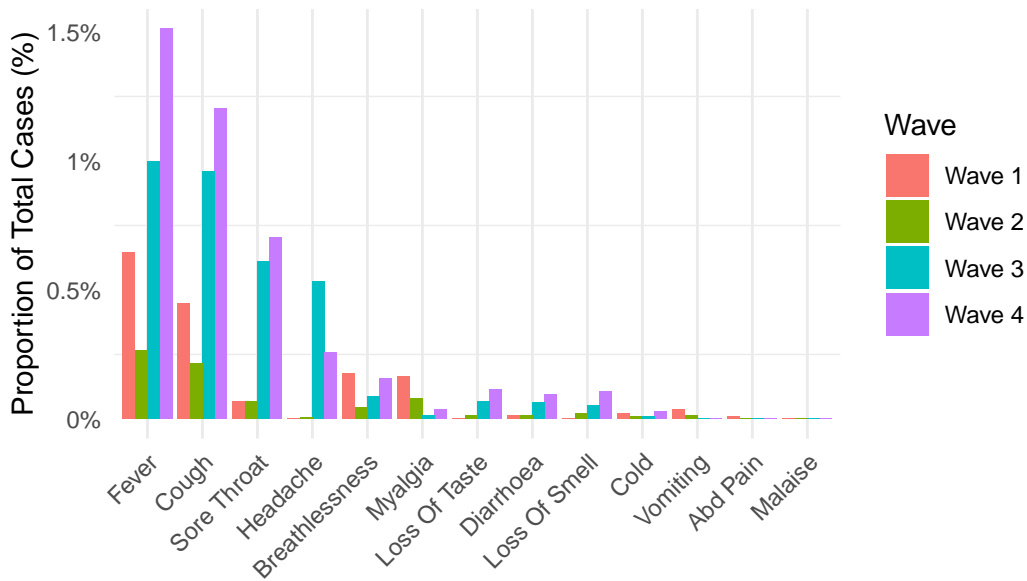


## 5.5 State-Wise COVID-19 Symptom Percentages by Wave in India

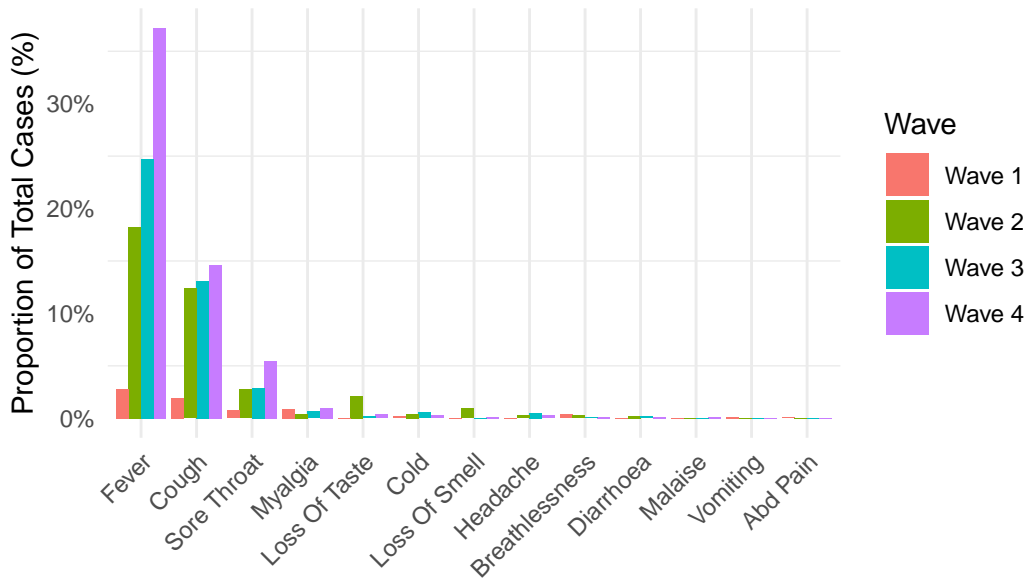
Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Andaman\_nicobar



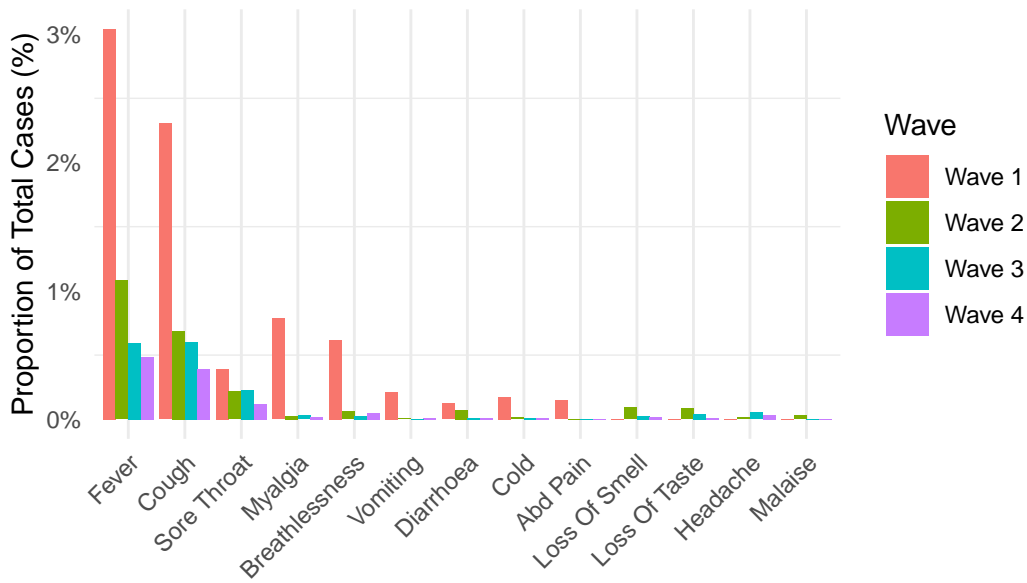
Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Andhra\_pradesh



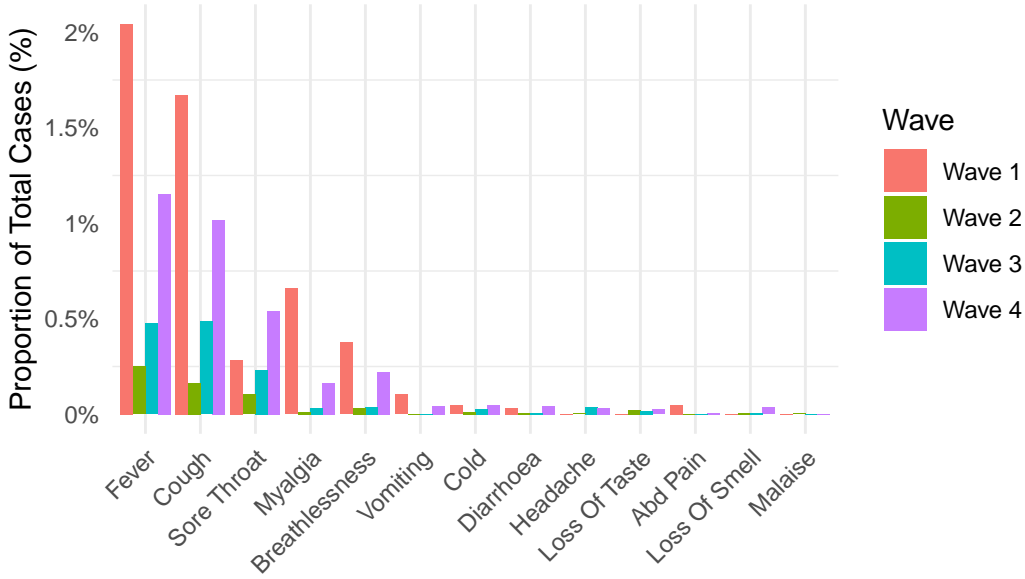
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Arunachal\_pradesh**



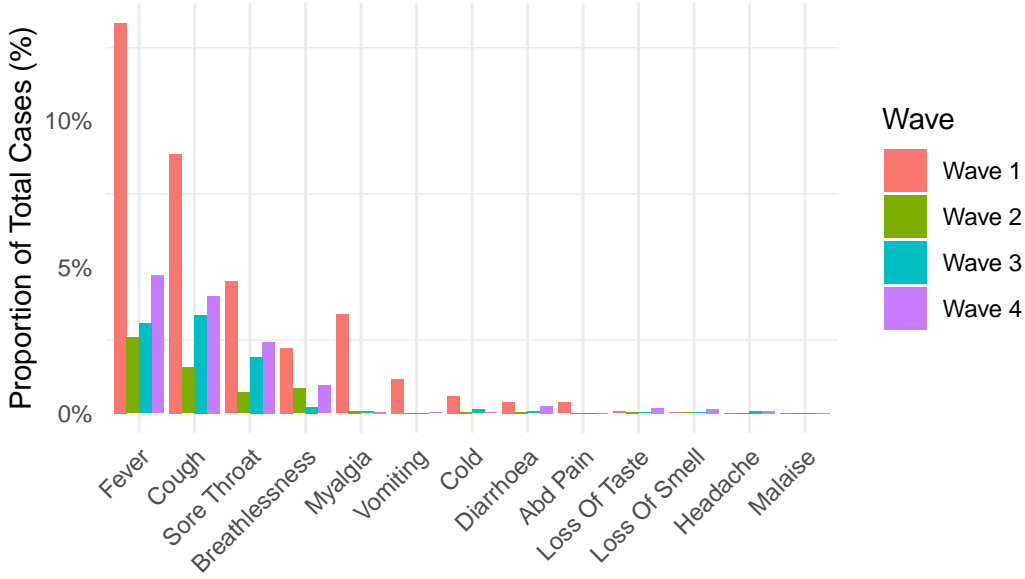
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Assam**



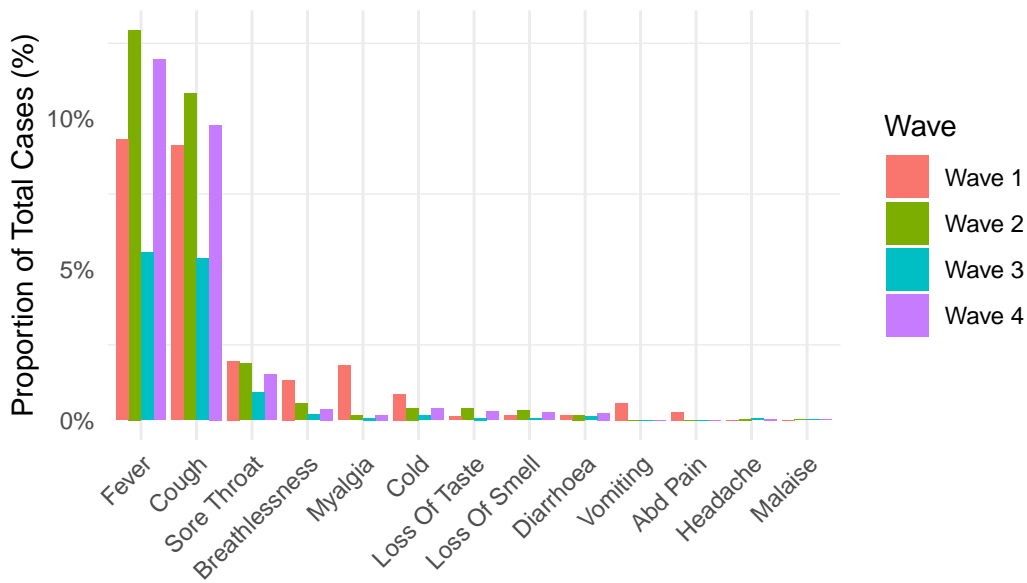
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Bihar**



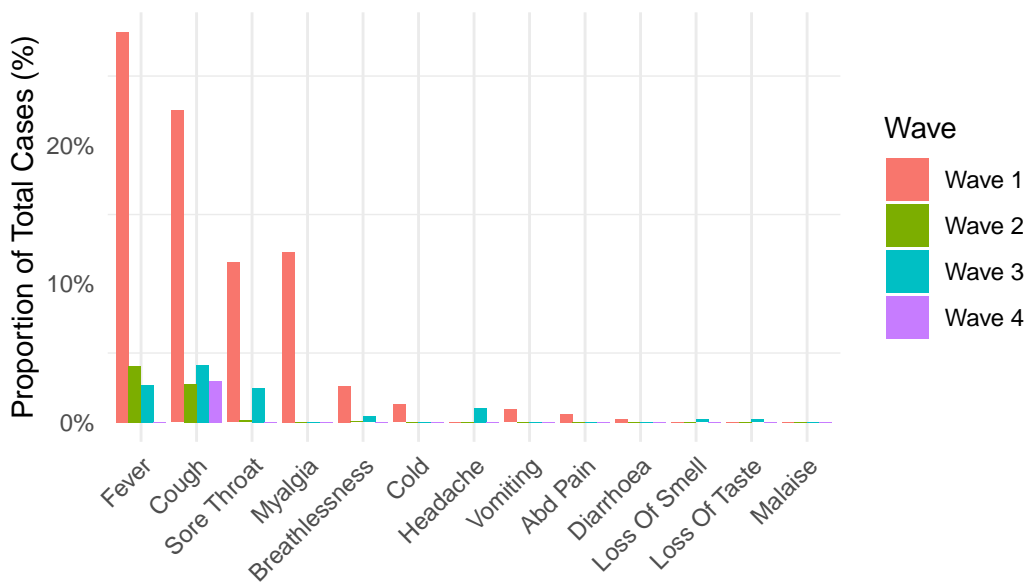
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Chandigarh**



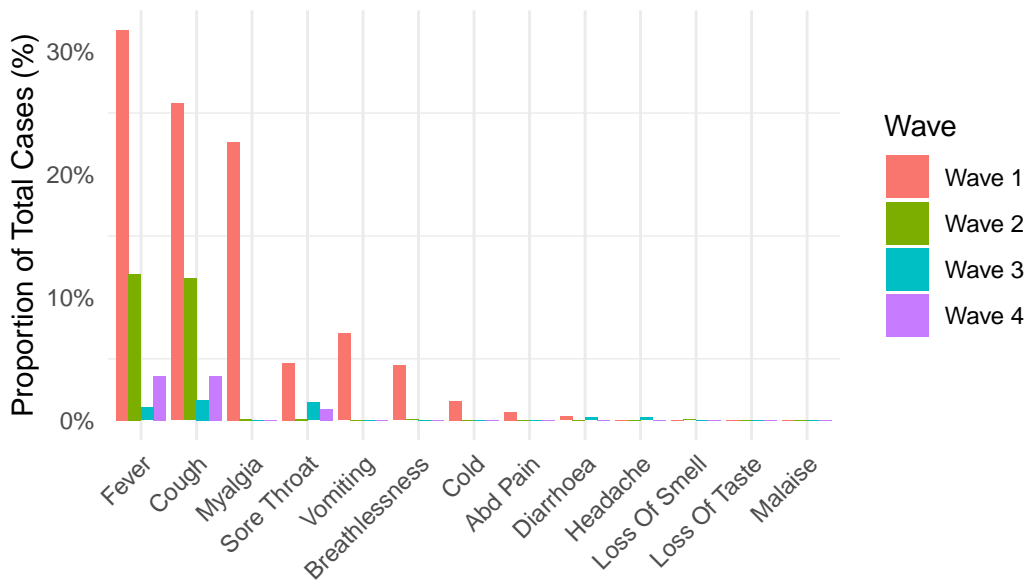
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Chhattisgarh**



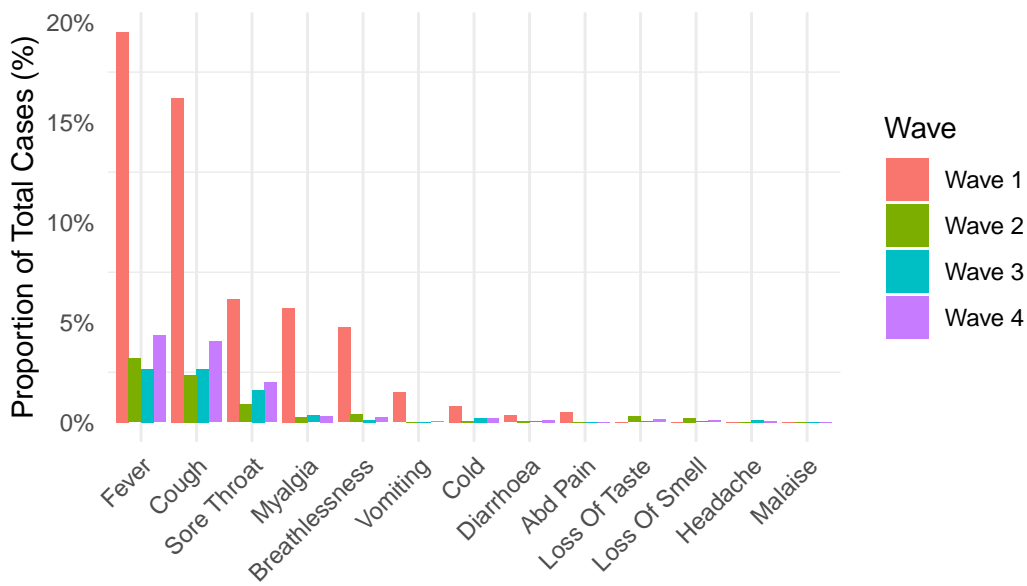
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Dadra\_nagar\_have**



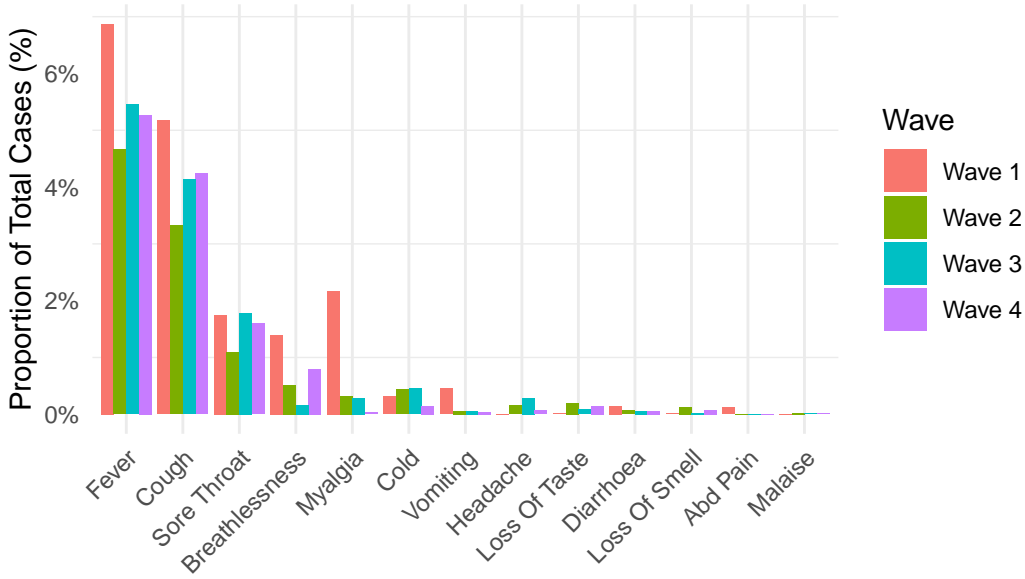
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Daman\_diu**



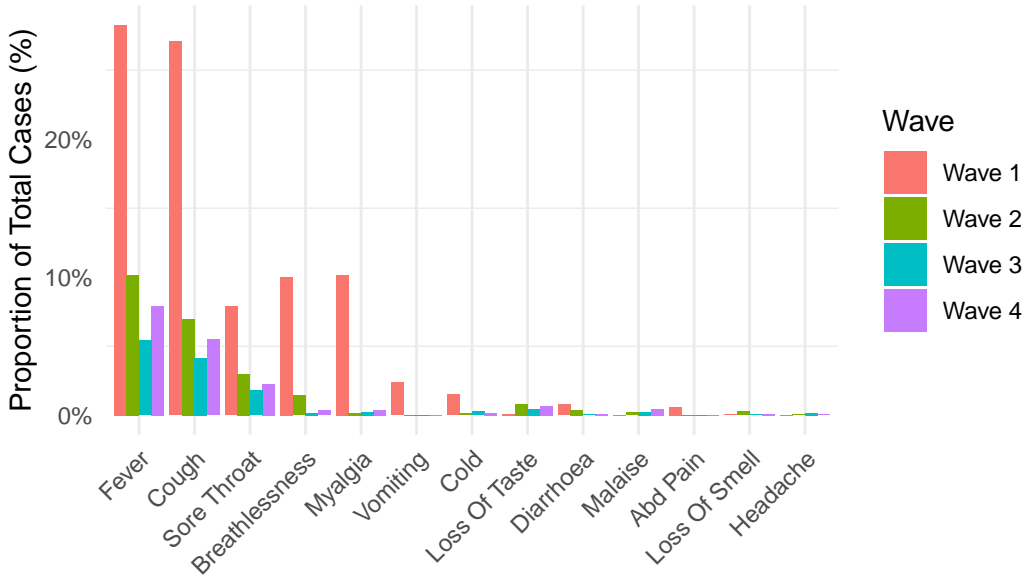
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Delhi**



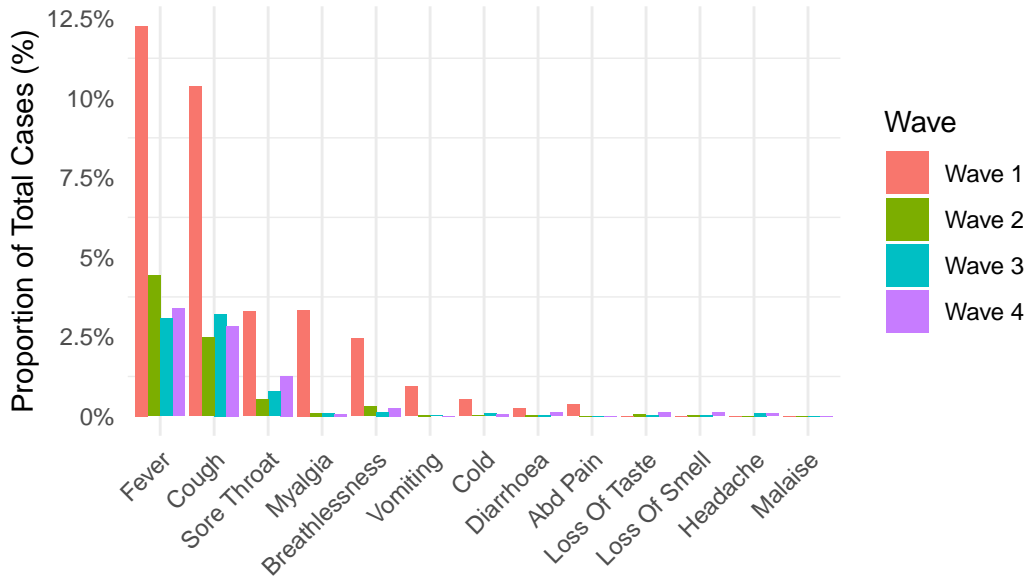
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Goa**



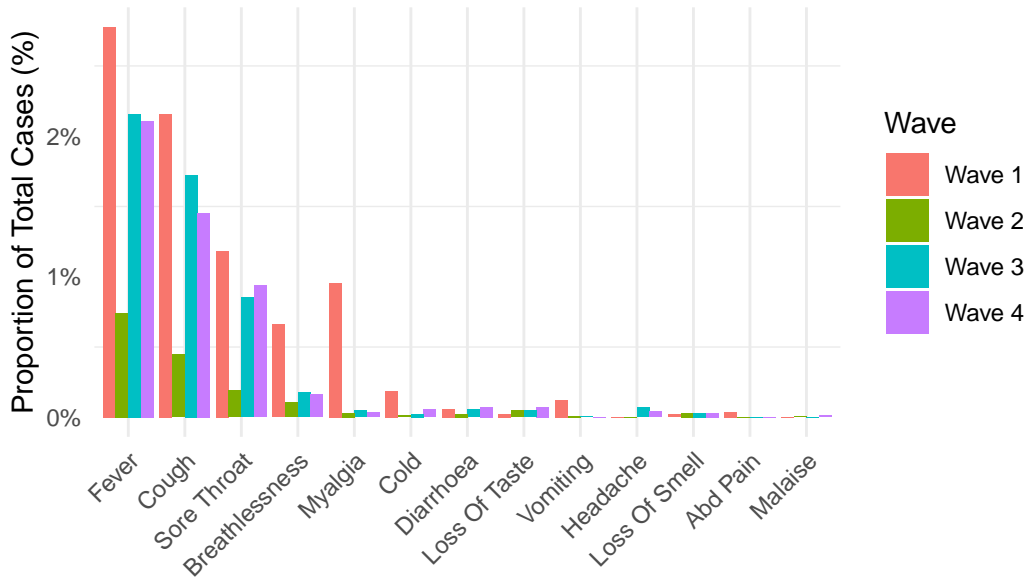
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Gujarat**



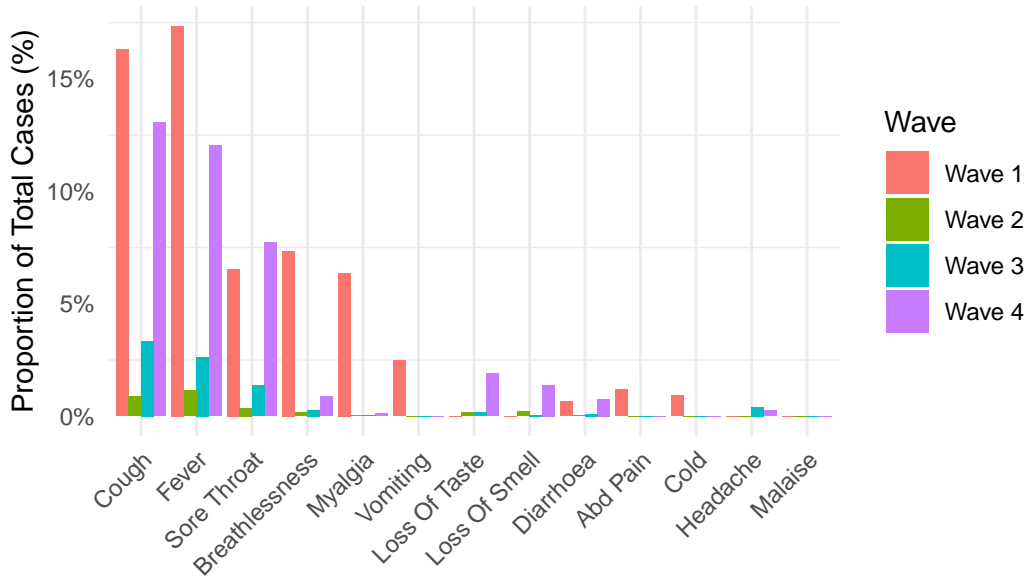
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Haryana**



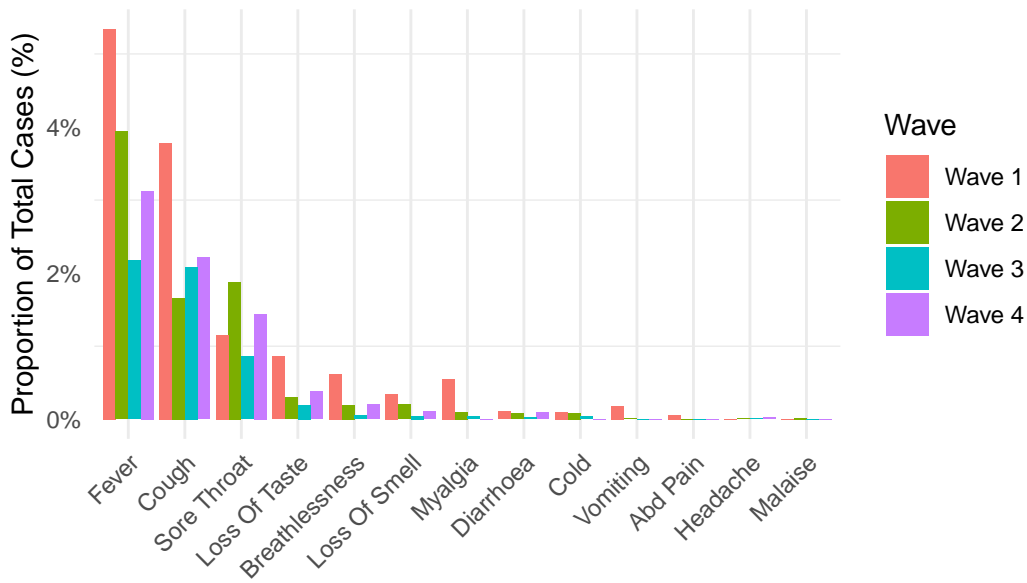
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Himachal\_pradesh**



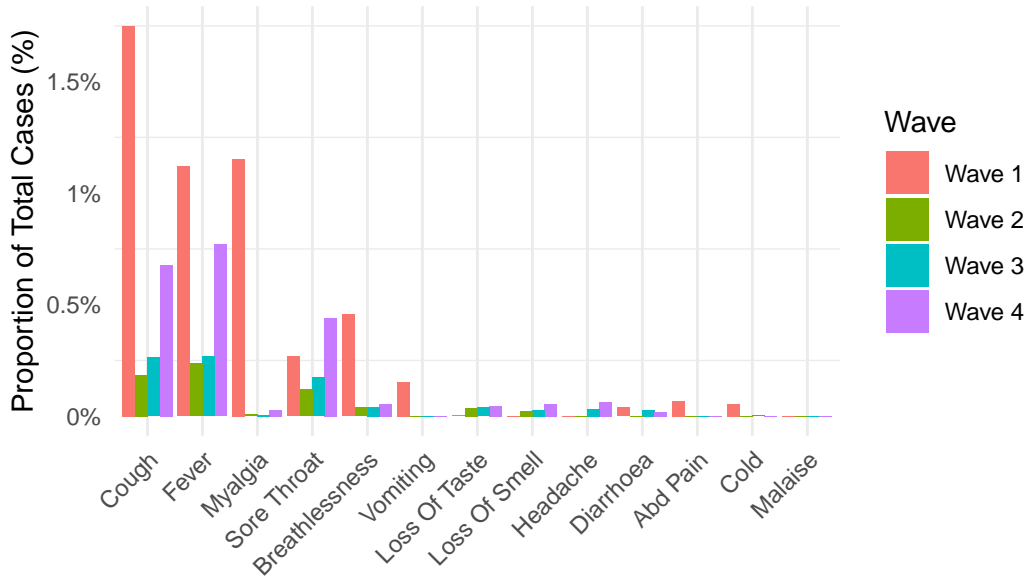
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in India**



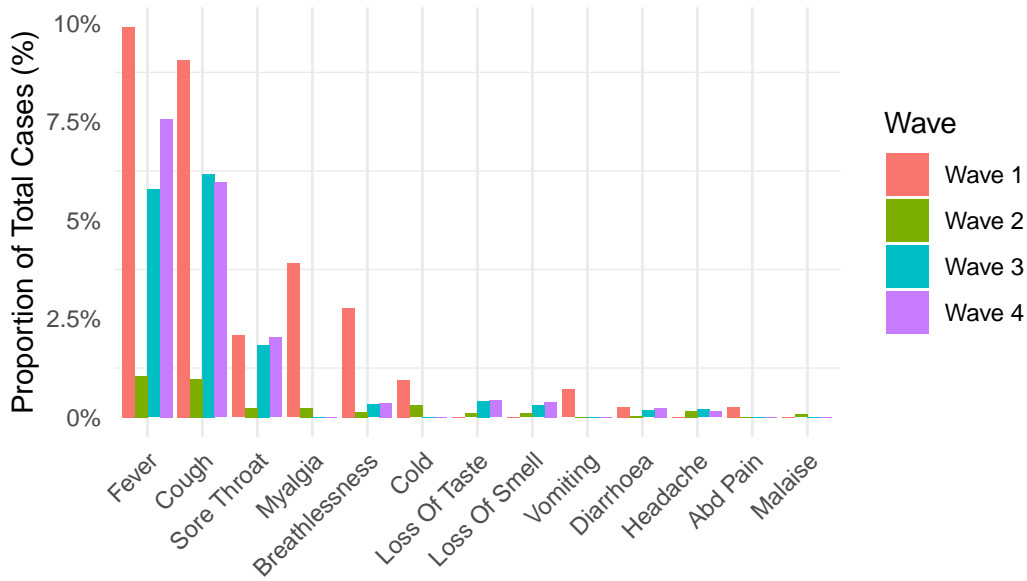
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Jammu\_kashmir**



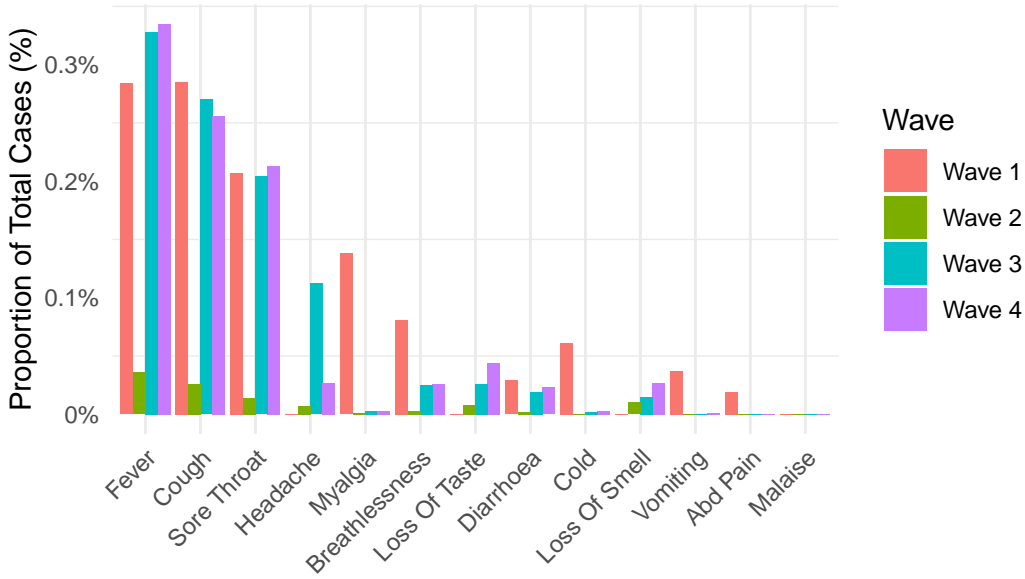
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Jharkhand**



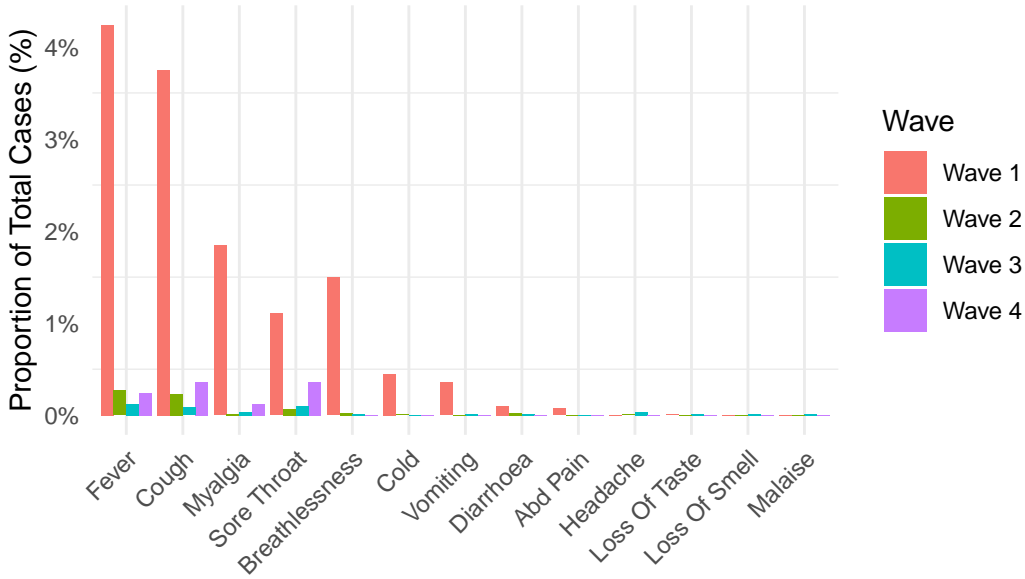
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Karnataka**



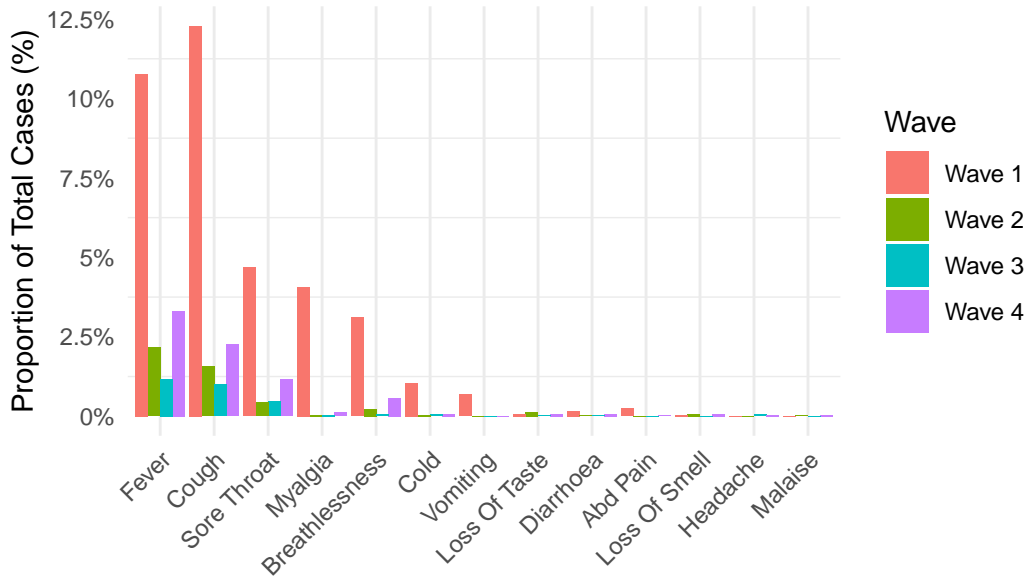
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Kerala**



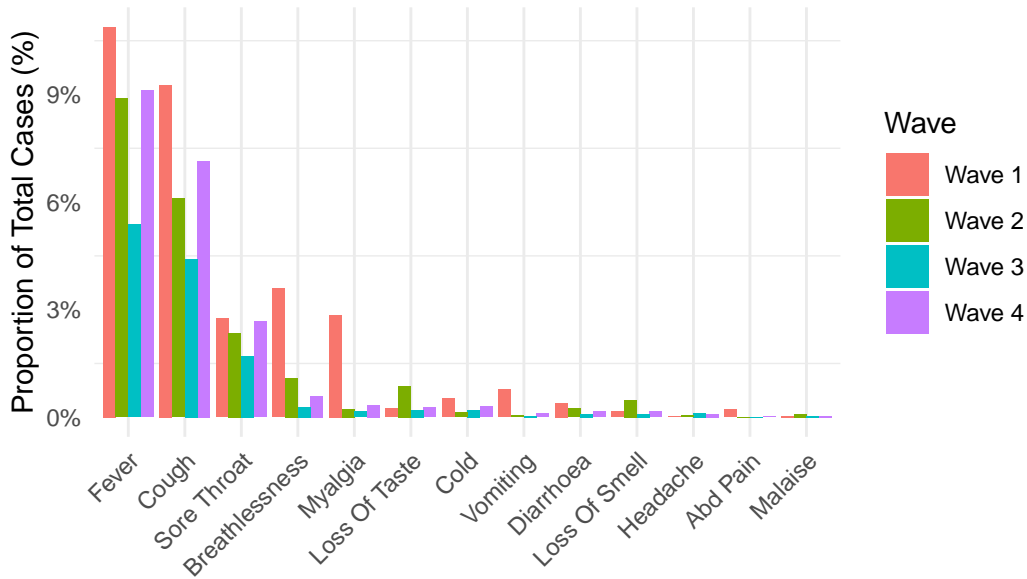
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Ladakh**



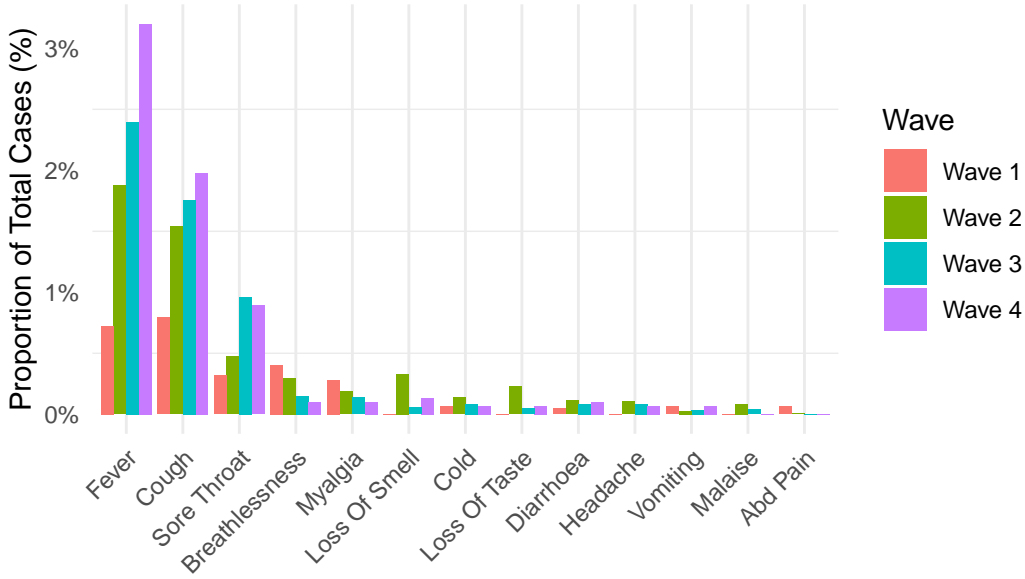
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Madhya\_pradesh**



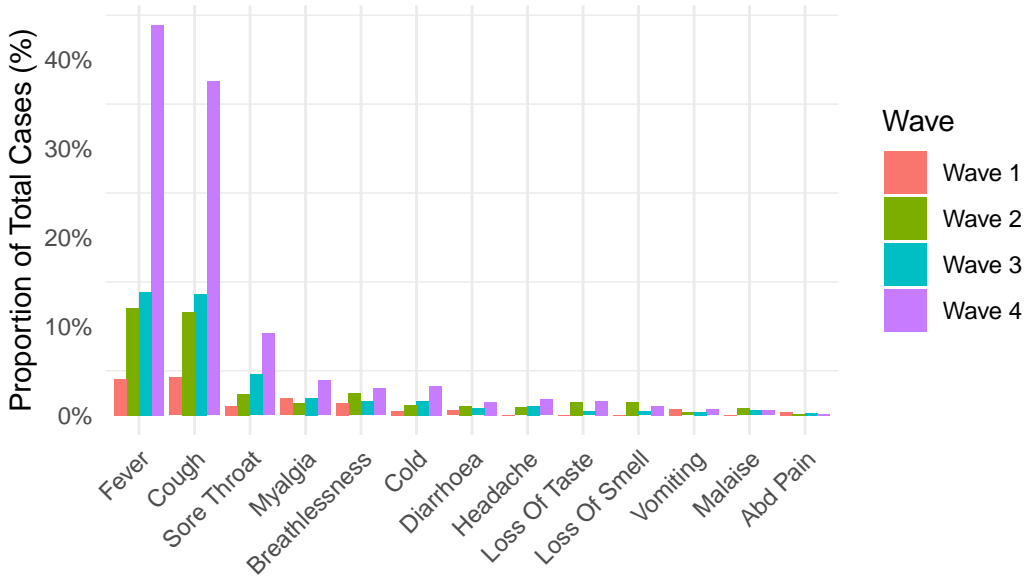
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Maharashtra**



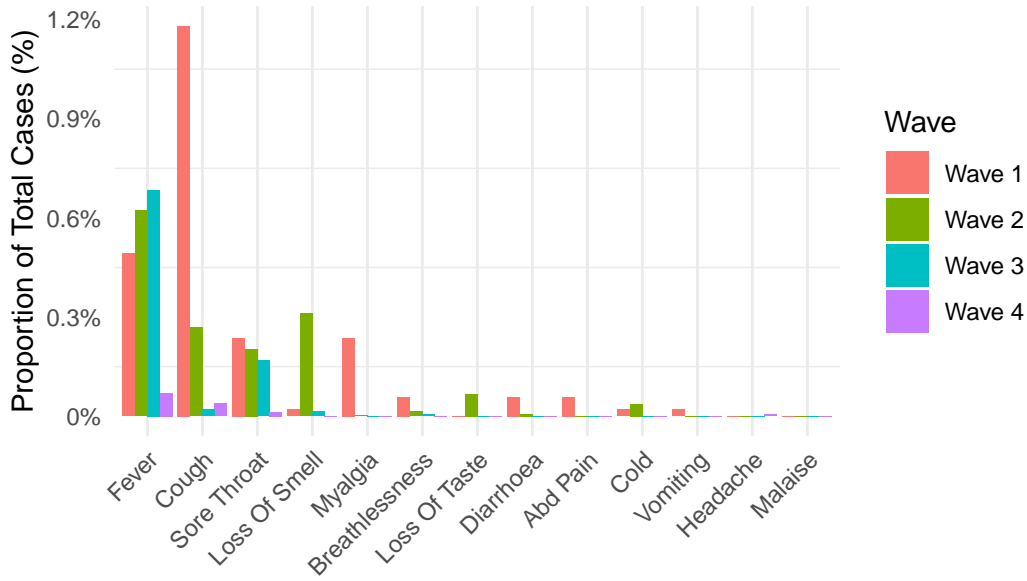
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Manipur**



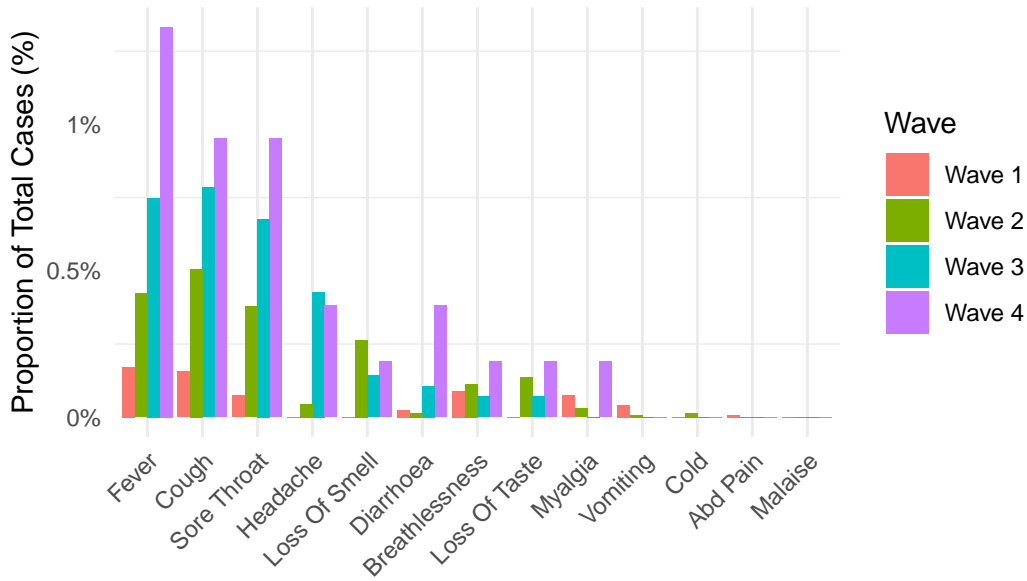
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Meghalaya**



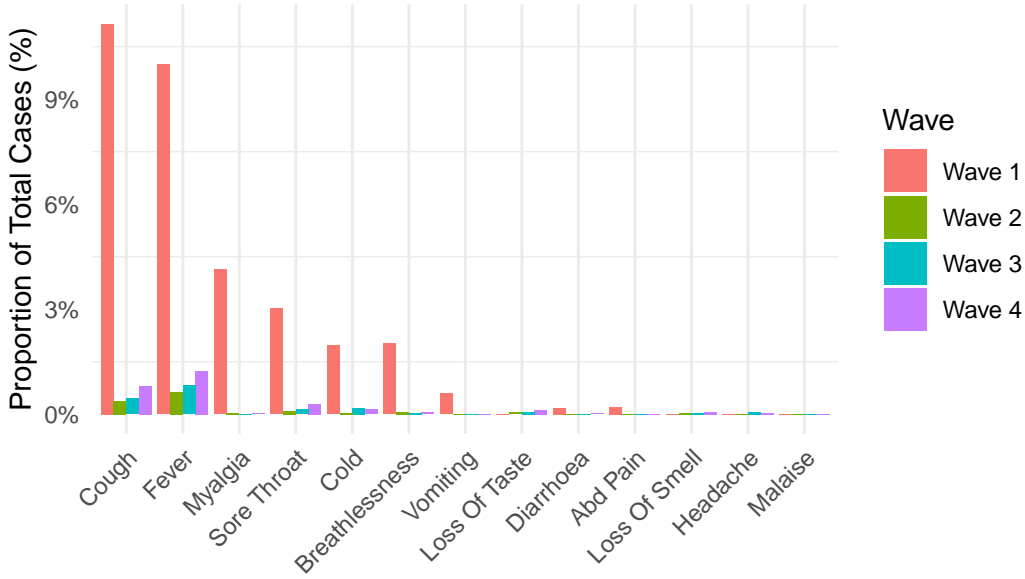
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Mizoram**



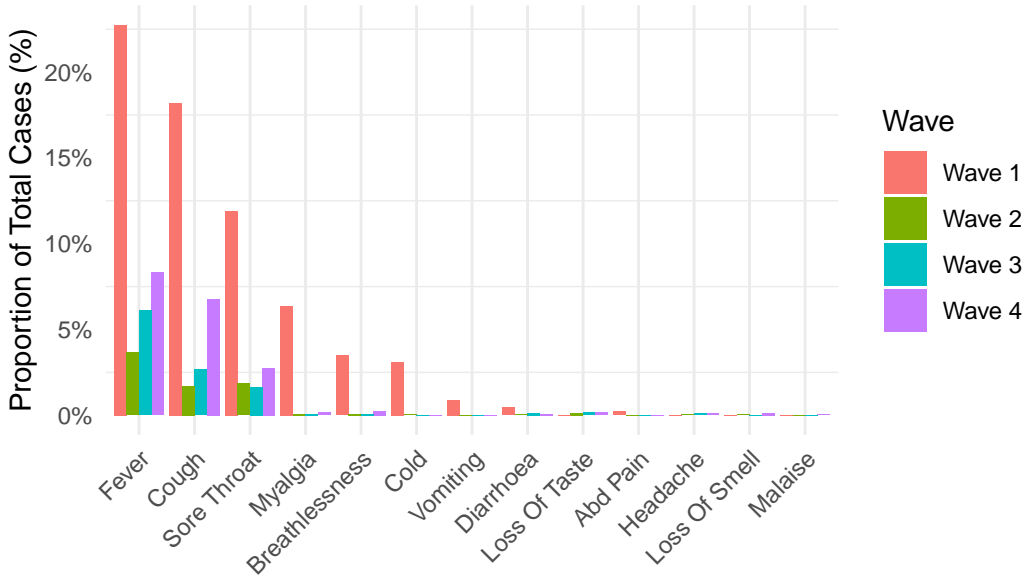
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Nagaland**



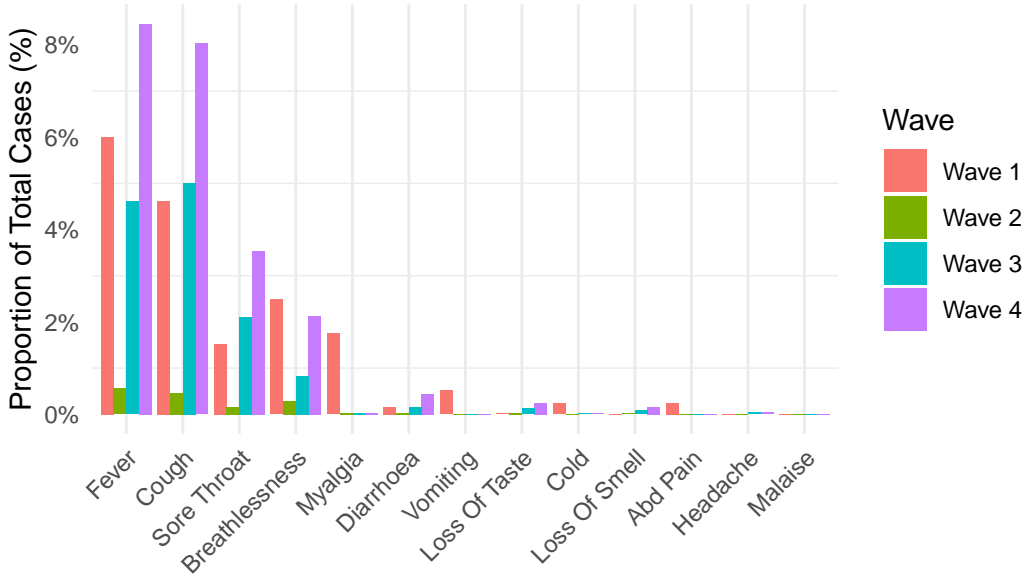
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Odisha**



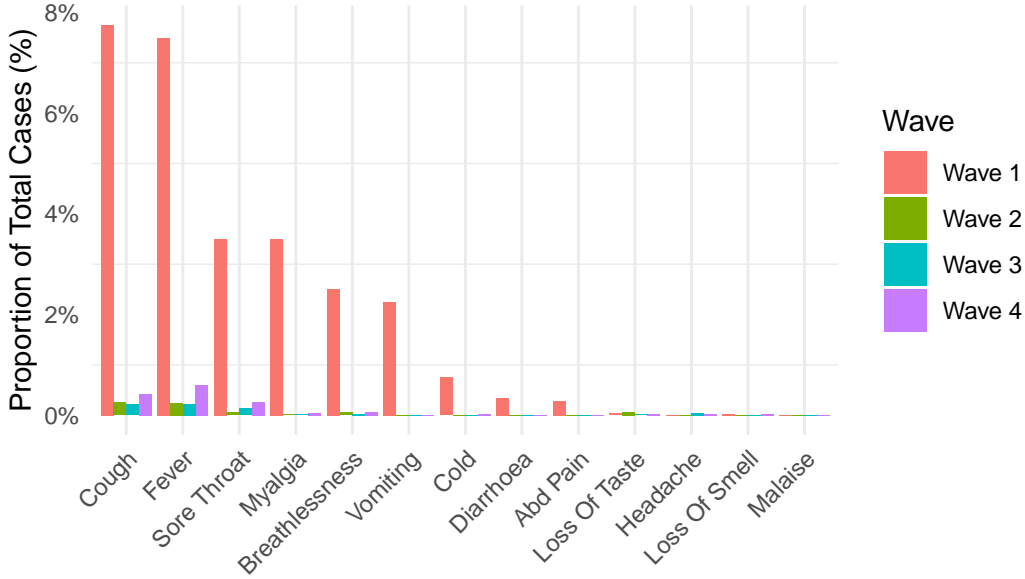
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Puducherry**



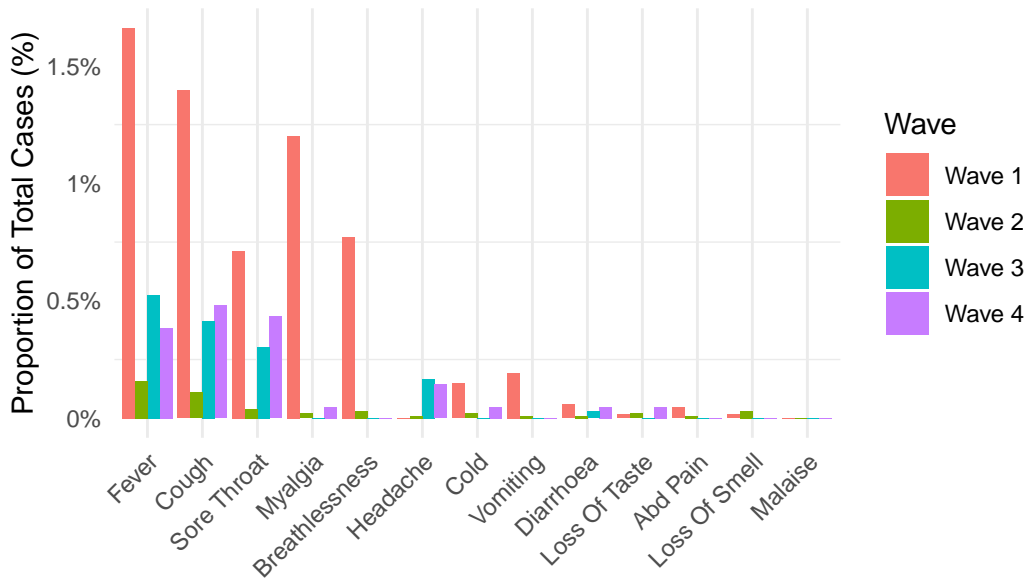
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Punjab**



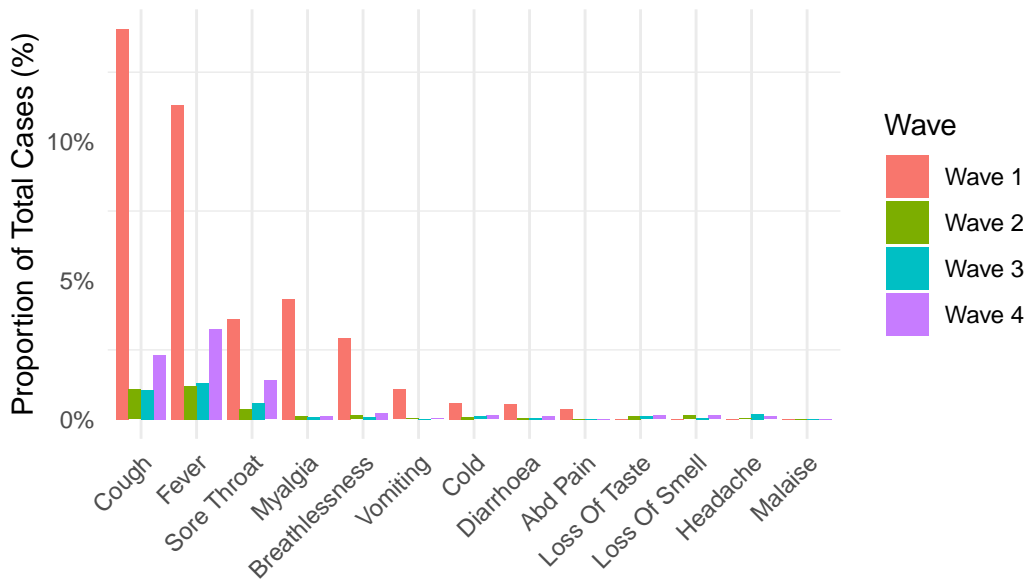
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Rajasthan**



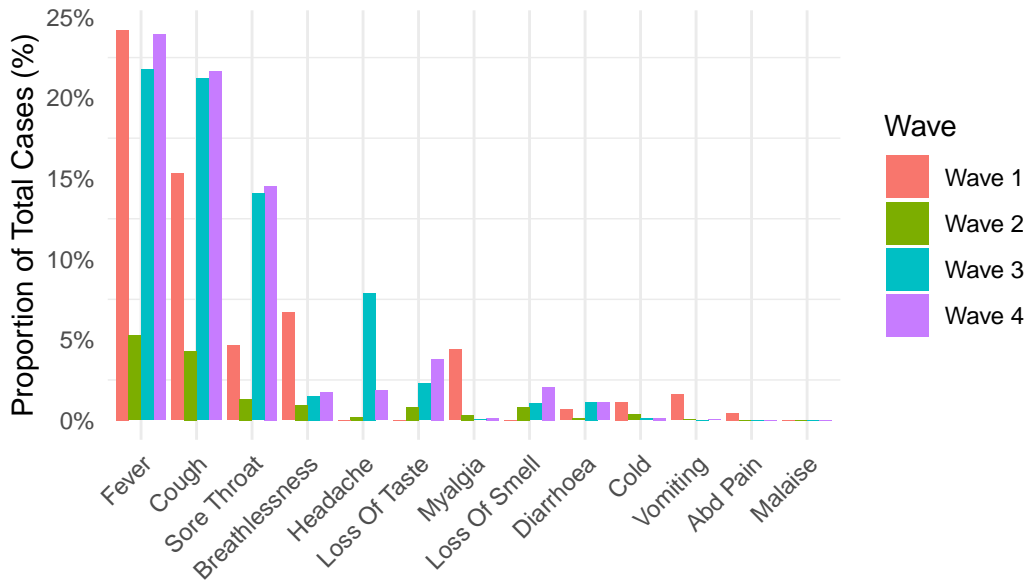
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Sikkim**



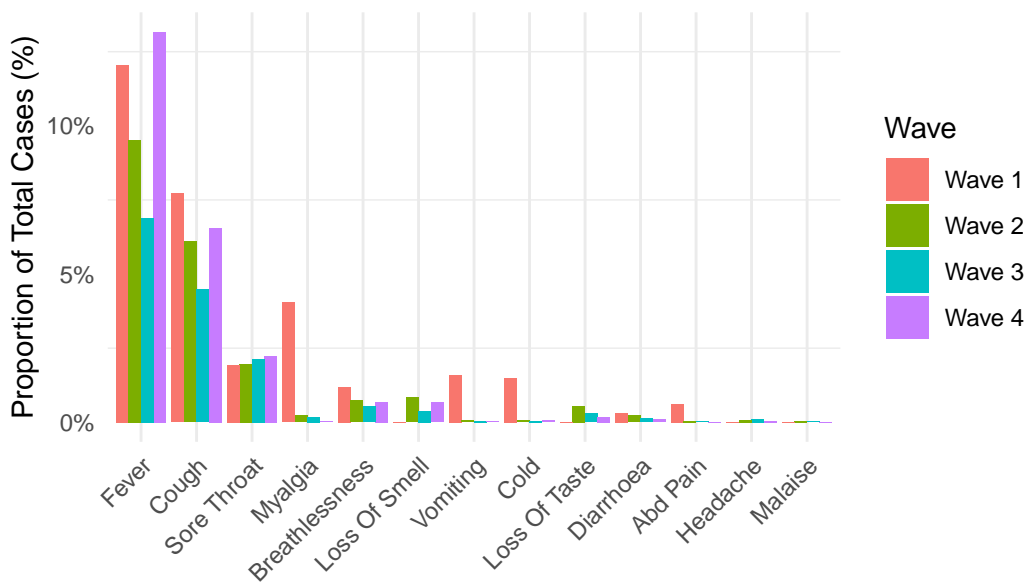
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Tamil\_nadu**



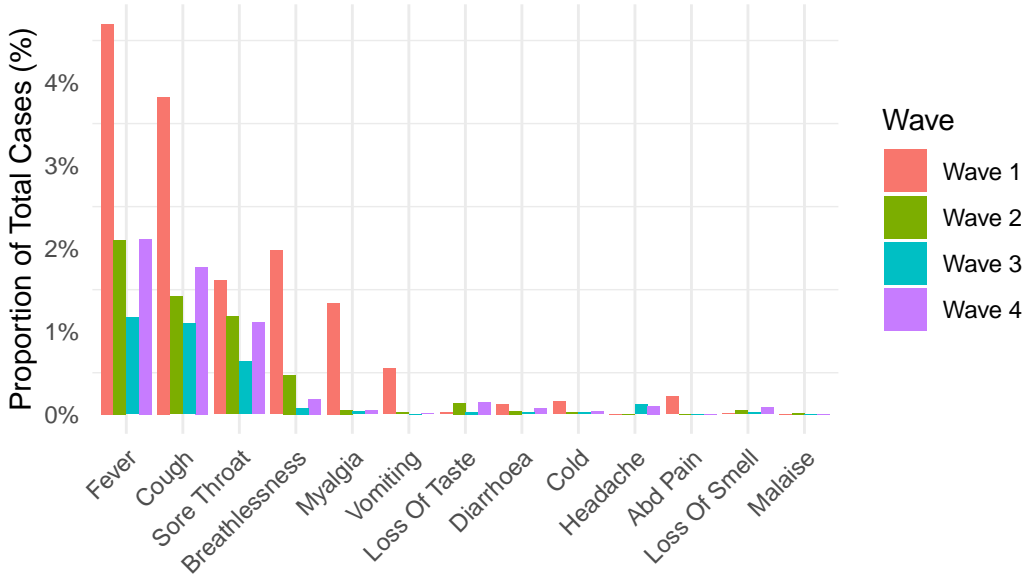
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Telangana**



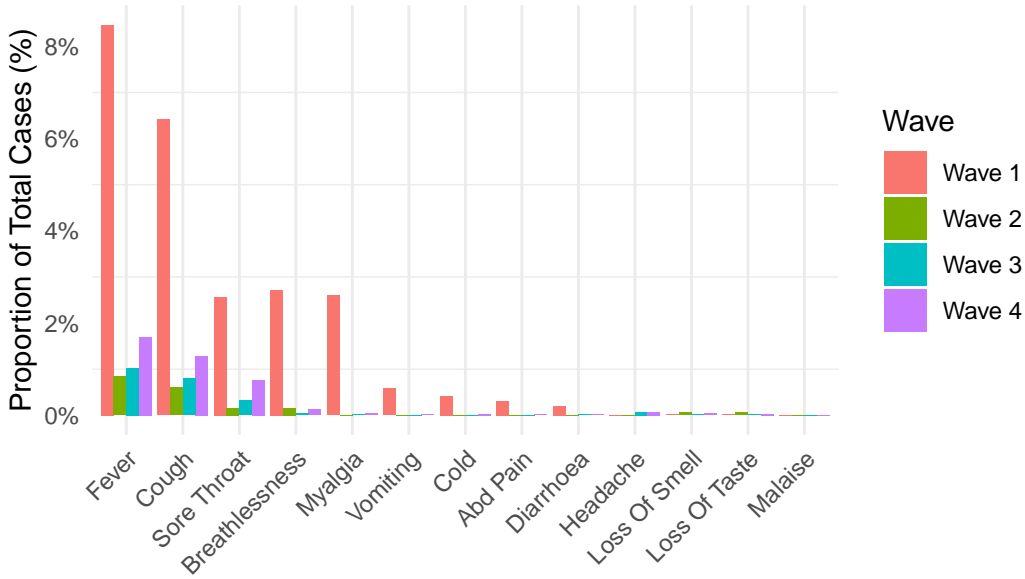
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Tripura**



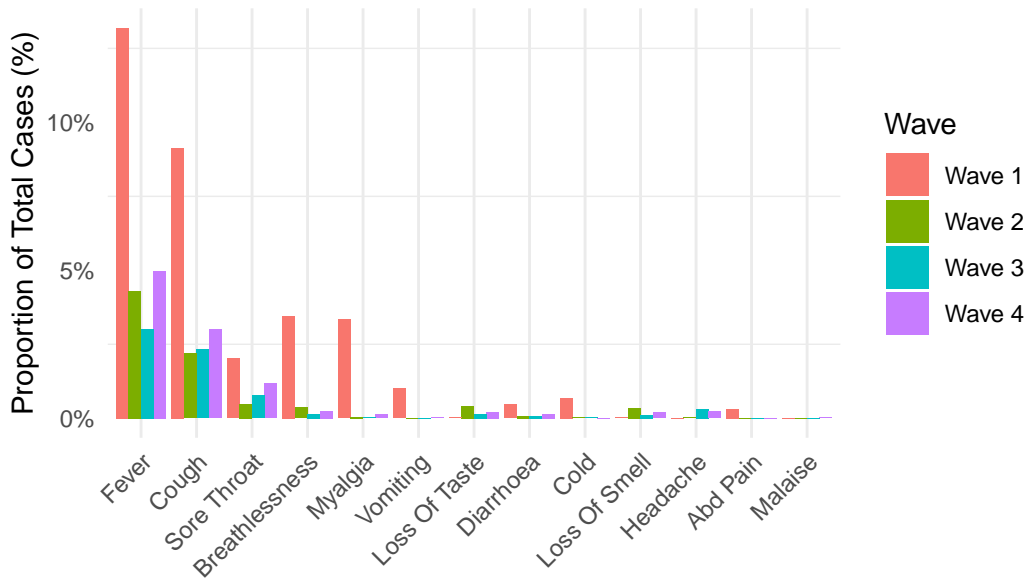
**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Uttar\_pradesh**



**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in Uttarakhand**

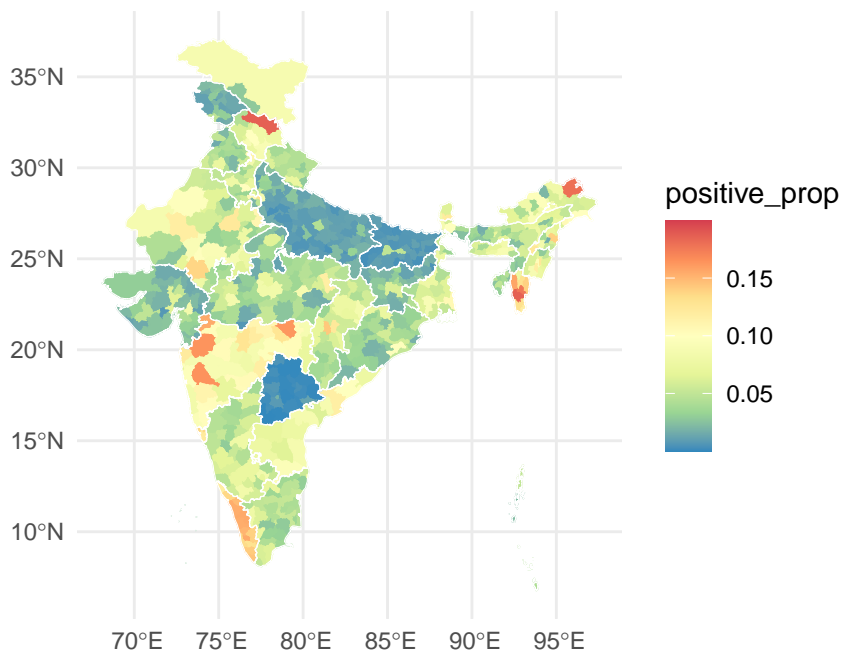


**Percentage of Symptoms Among COVID-19 Positive Cases During Wave 1 and Wave 2 in West\_bengal**

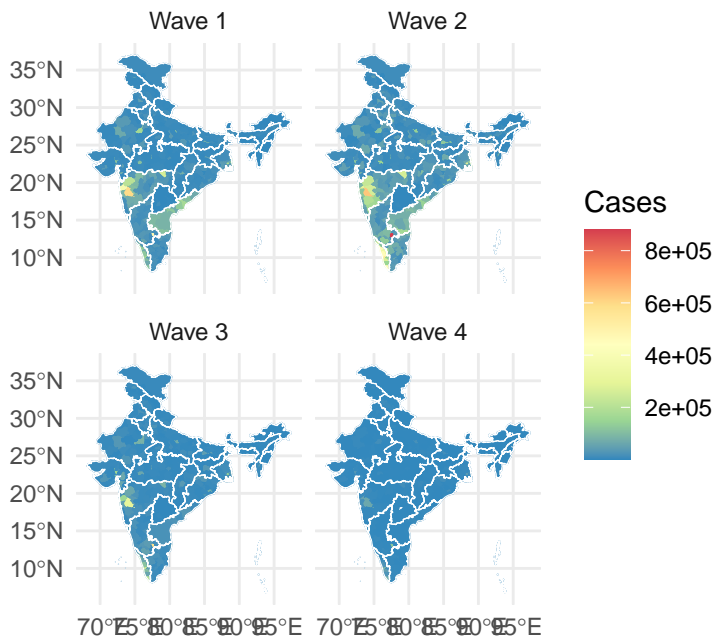


## 5.6 Spatial Distribution

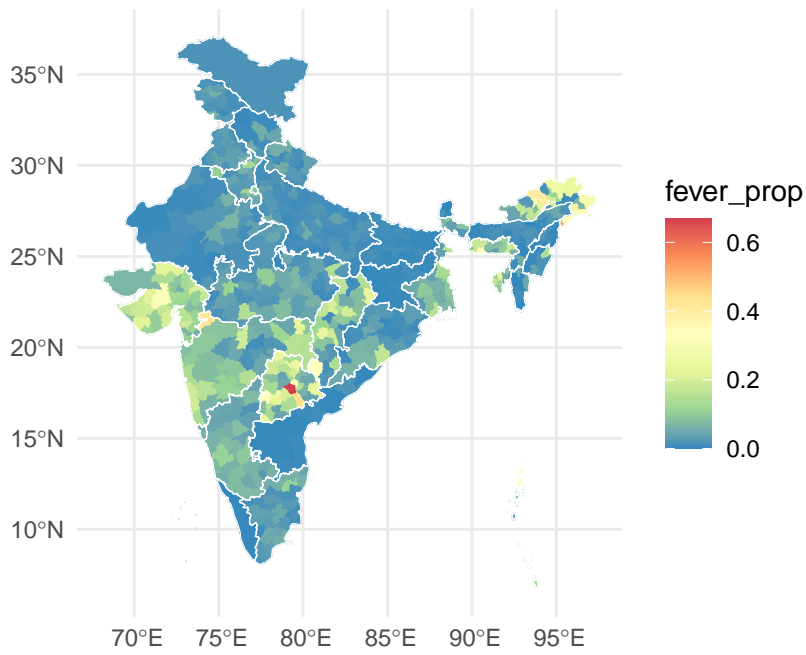
### 5.6.1 Test Positivity Rate



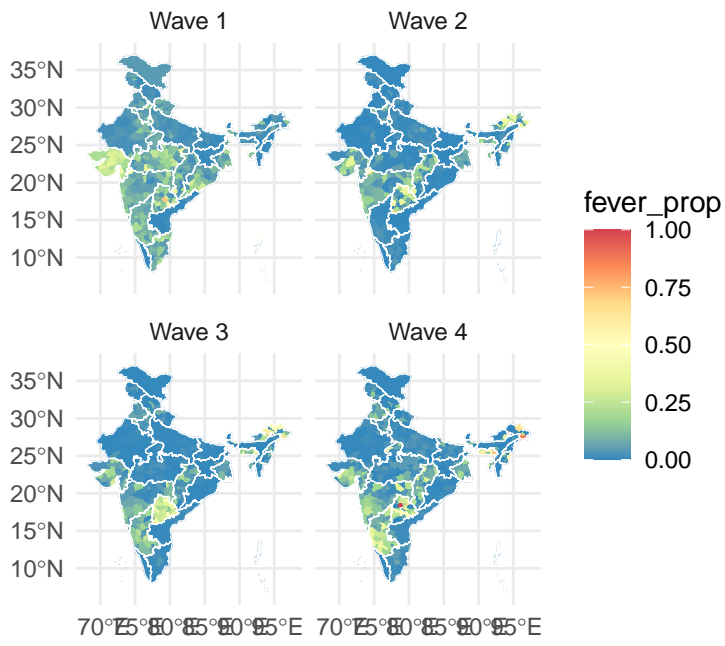
### 5.6.2 Test Positivity Rate by Wave



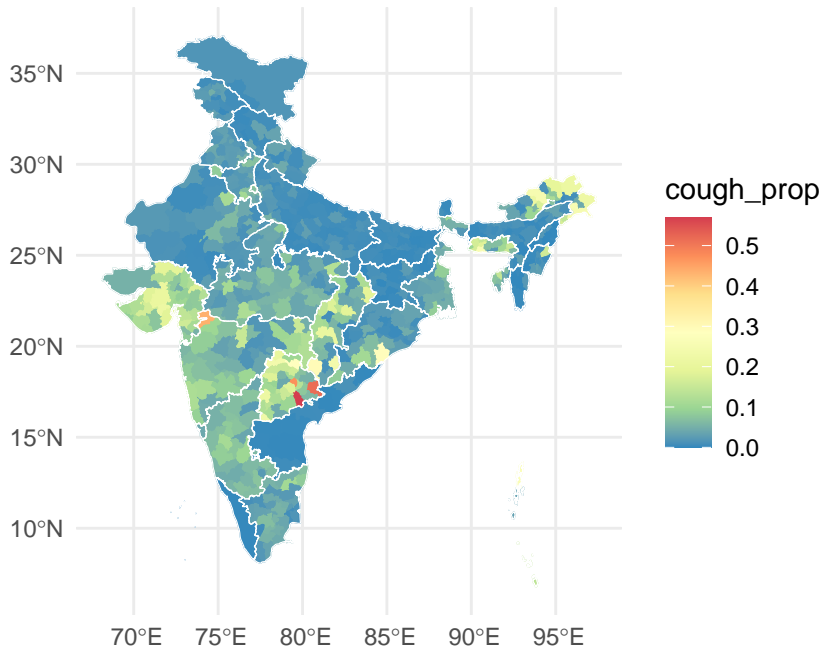
### 5.6.3 Fever Rate

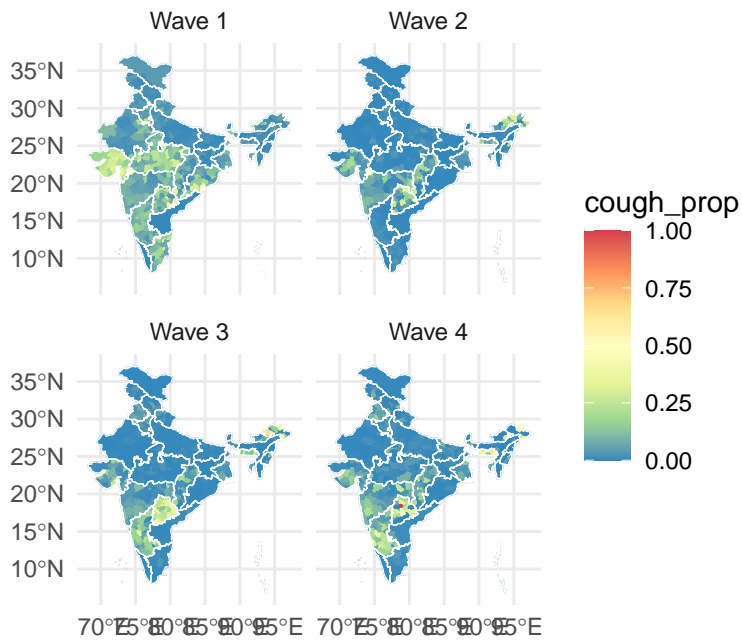


### 5.6.4 Fever Rate by Wave

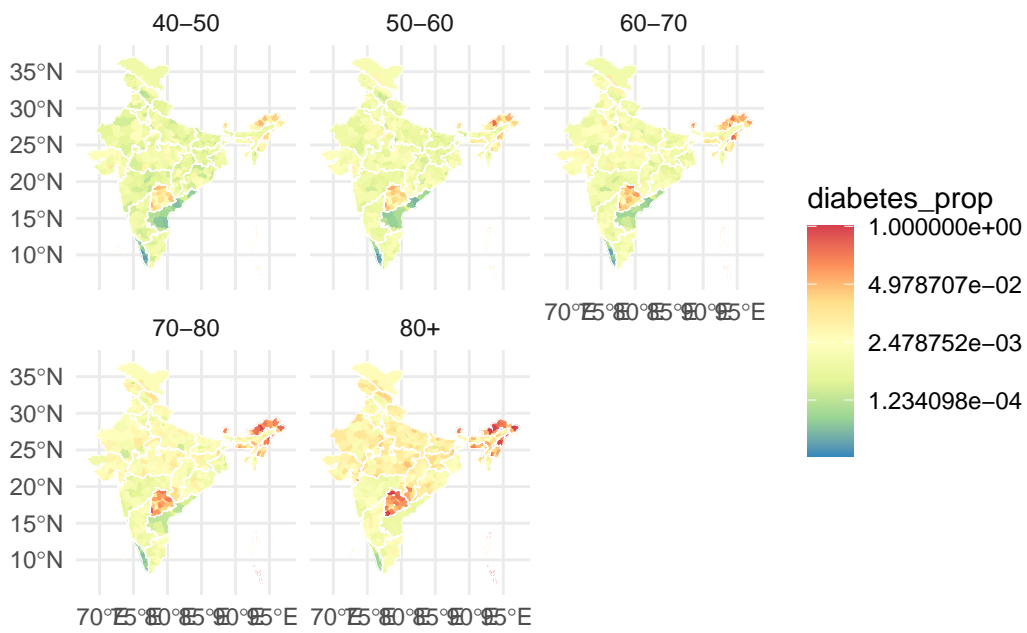


### 5.6.5 Cough Rate





### 5.6.6 Diabetes by Age Group

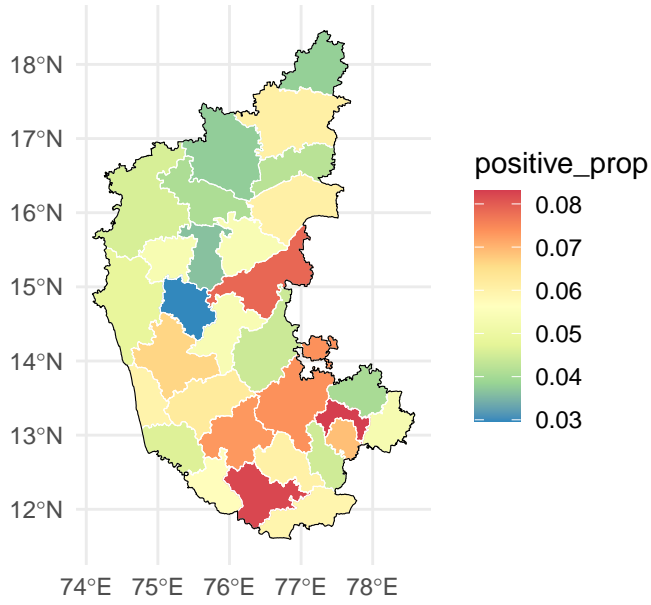


### 5.6.7 State Level TPR

[[1]]

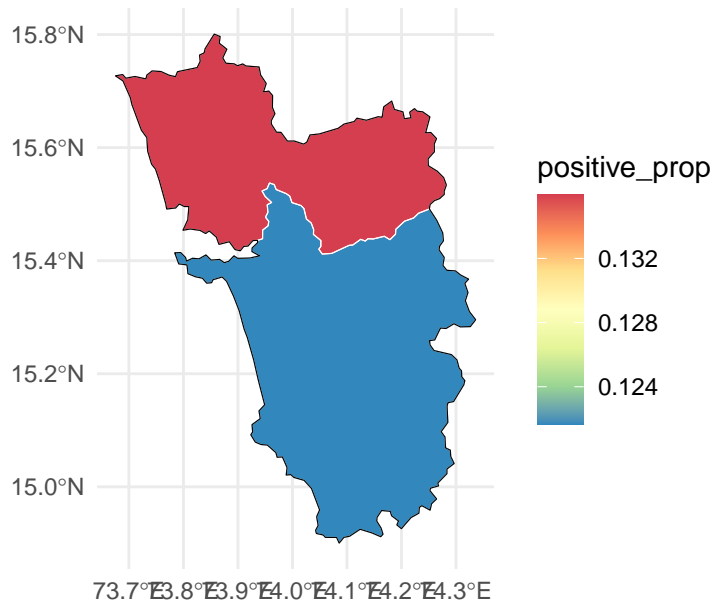


Test Positivity by Districts in the state of Karnataka



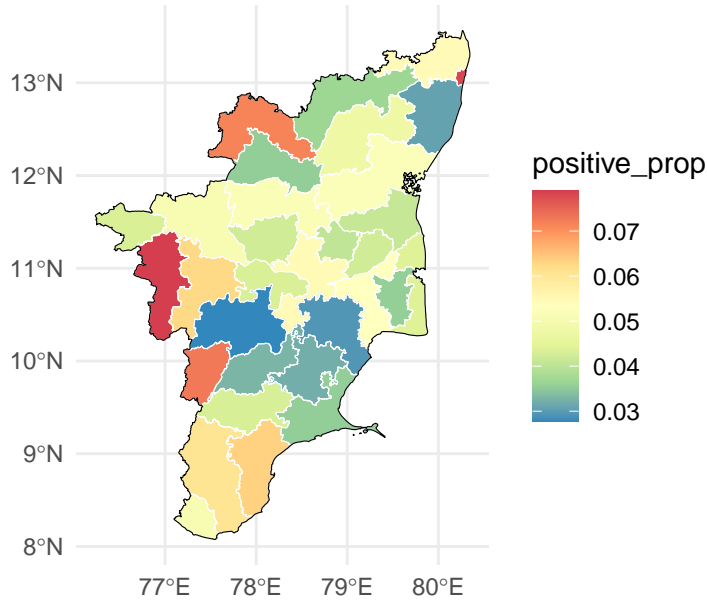
[[4]]

Test Positivity by Districts in the state of Goa



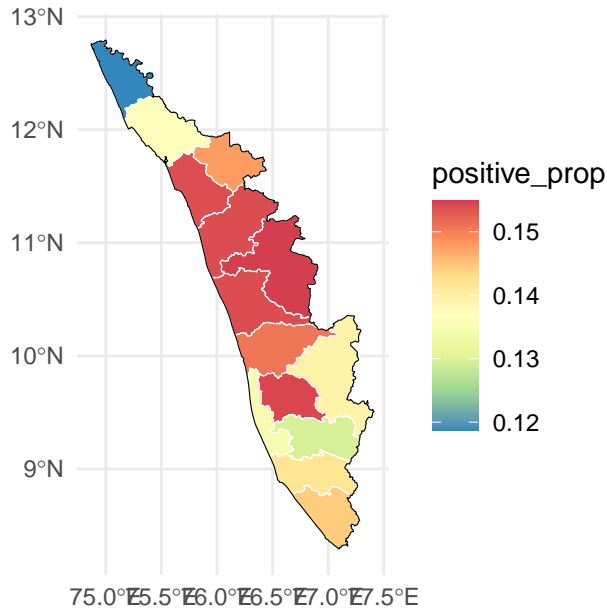
[[5]]

Test Positivity by Districts in the state of Tamil Nadu



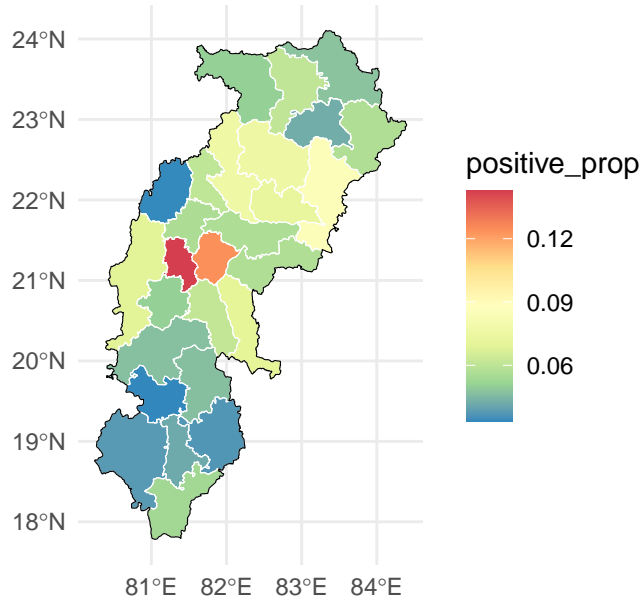
[[6]]

Test Positivity by Districts in the state of Kerala



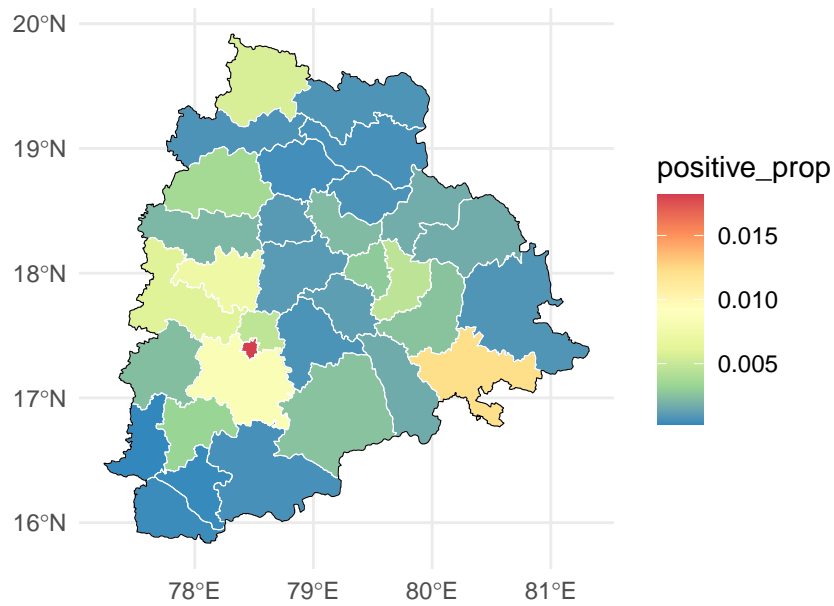
[[7]]

Test Positivity by Districts in the state of Chhattisgarh



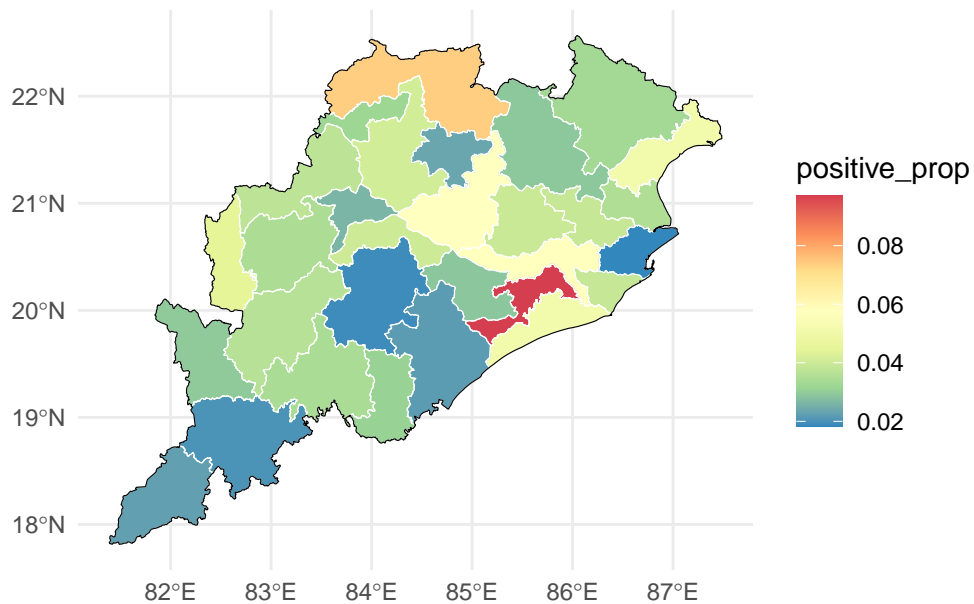
[[8]]

Test Positivity by Districts in the state of Telangana



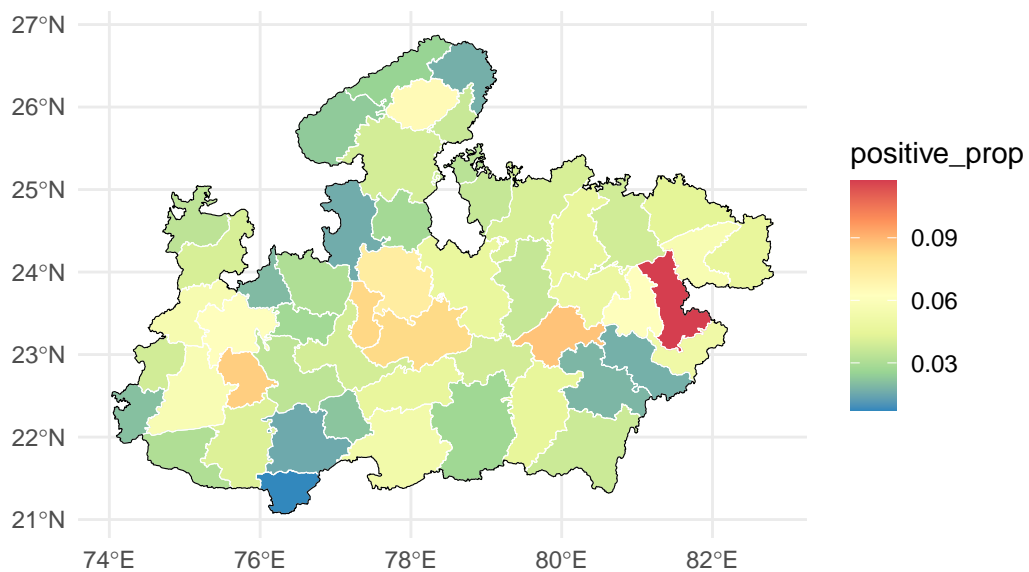
[[9]]

Test Positivity by Districts in the state of Odisha



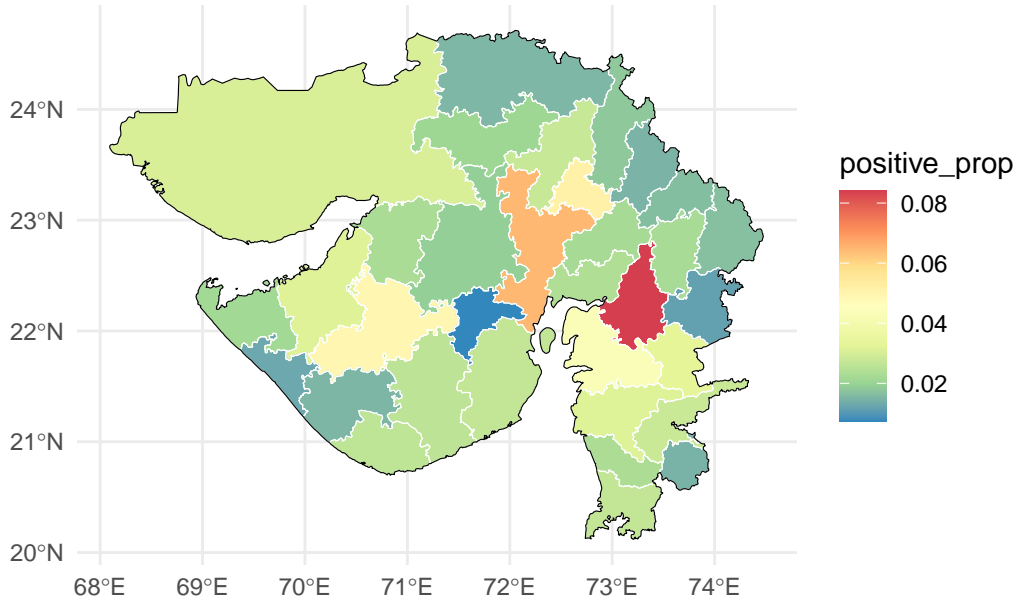
[[10]]

Test Positivity by Districts in the state of Madhya Pradesh



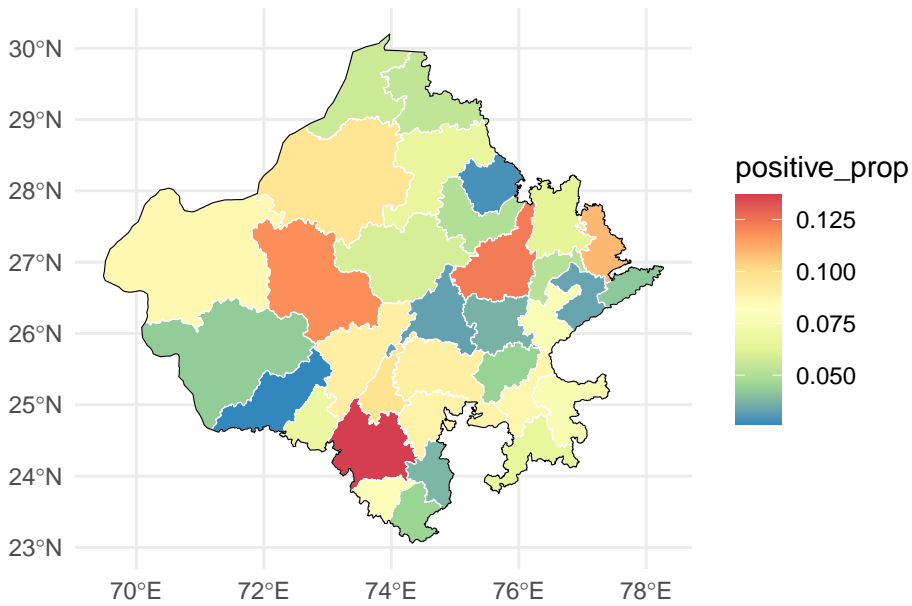
[[11]]

Test Positivity by Districts in the state of Gujarat



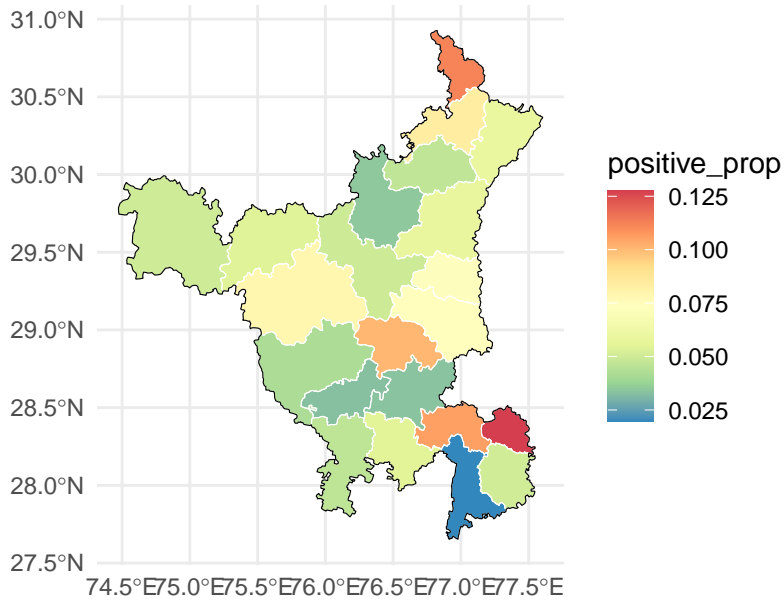
[[12]]

Test Positivity by Districts in the state of Rajasthan



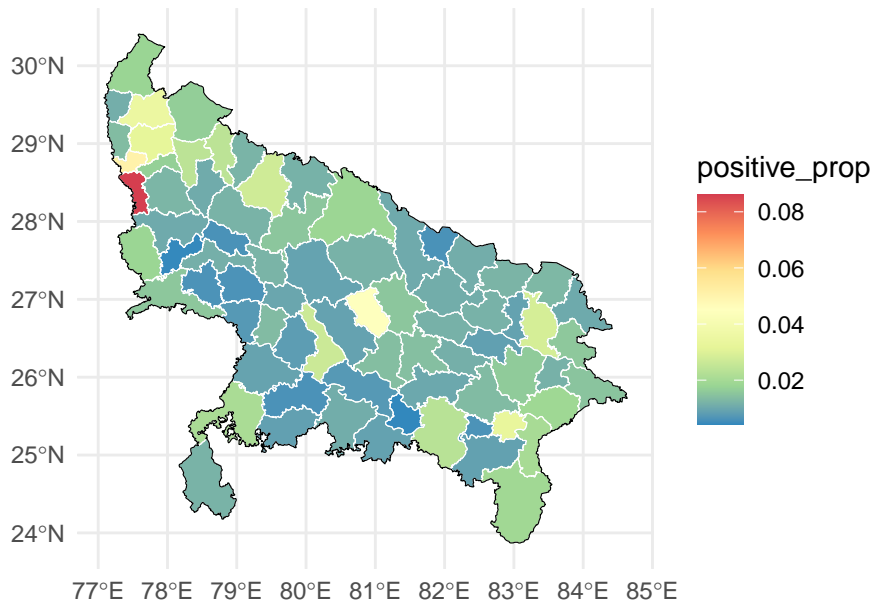
[[13]]

Test Positivity by Districts in the state of Haryana

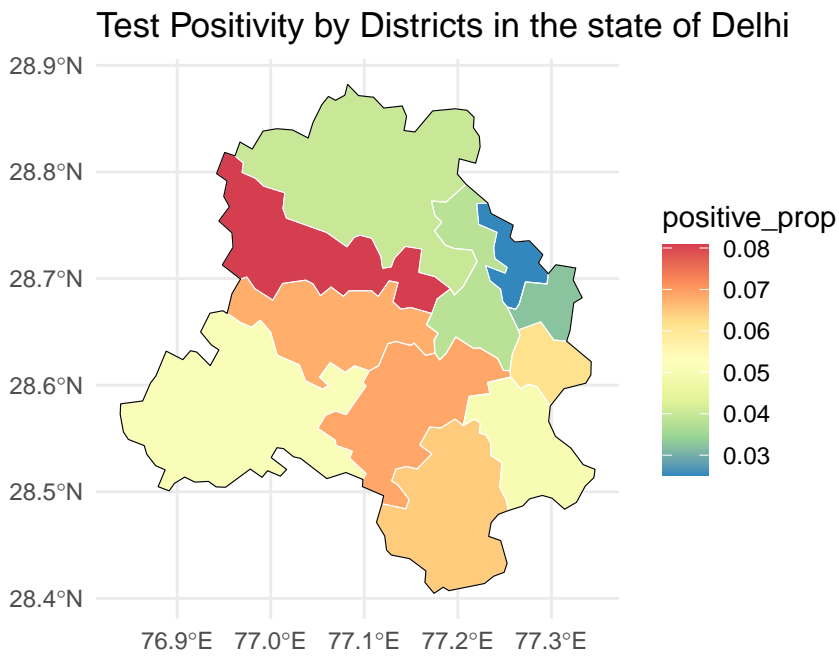


[[14]]

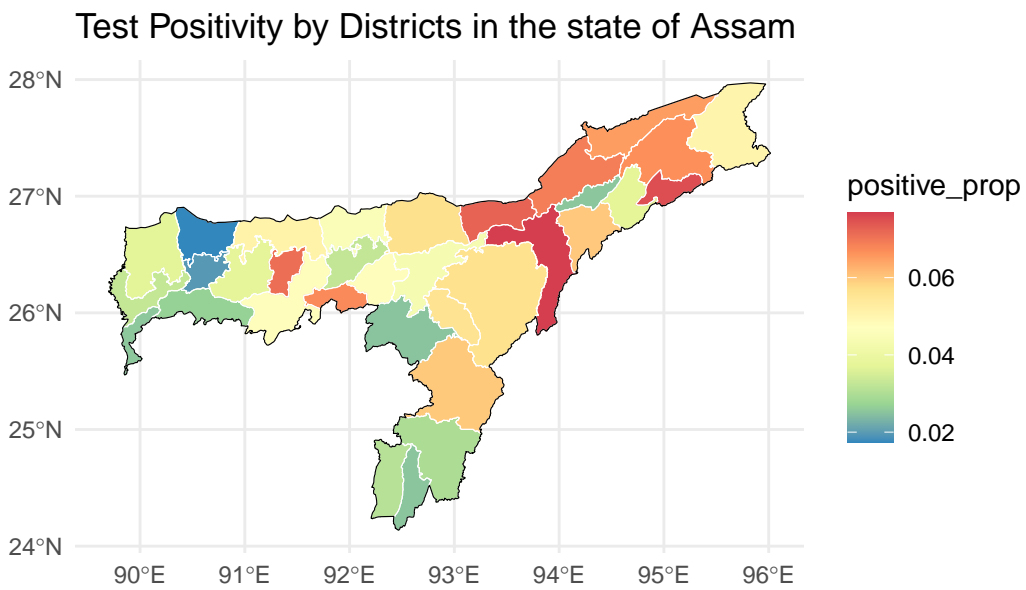
Test Positivity by Districts in the state of Uttar Pradesh



[[15]]

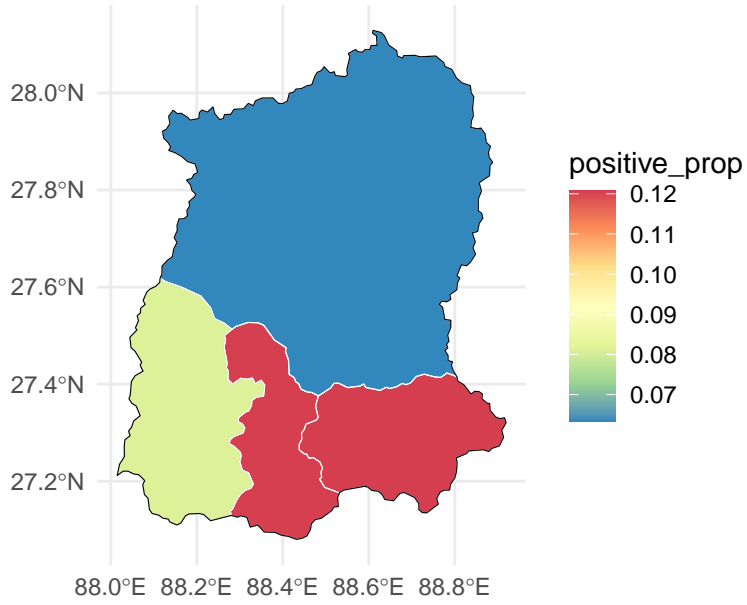


[[16]]



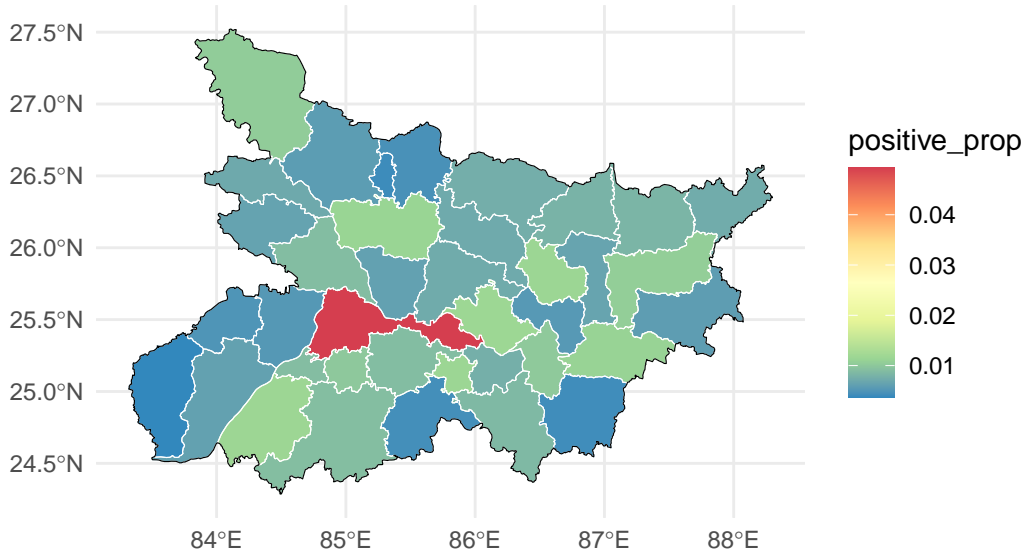
[[17]]

Test Positivity by Districts in the state of Sikkim



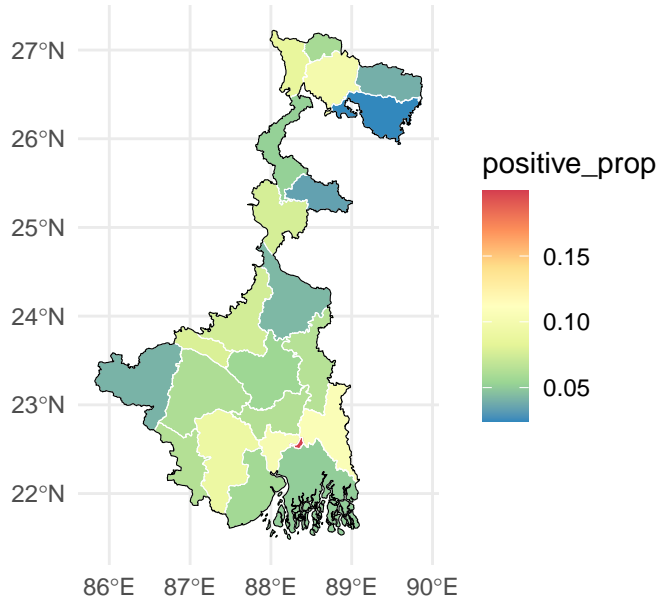
[[18]]

Test Positivity by Districts in the state of Bihar



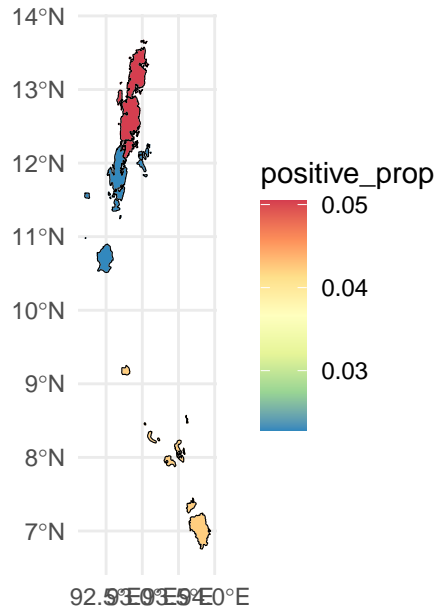
[[19]]

Test Positivity by Districts in the state of West Bengal:



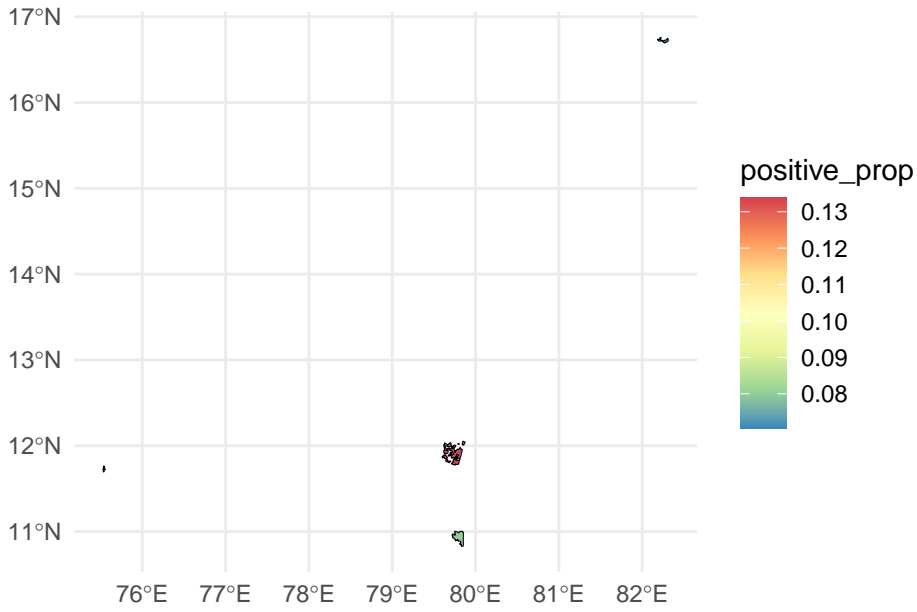
[[20]]

Test Positivity by Districts in the state of And:



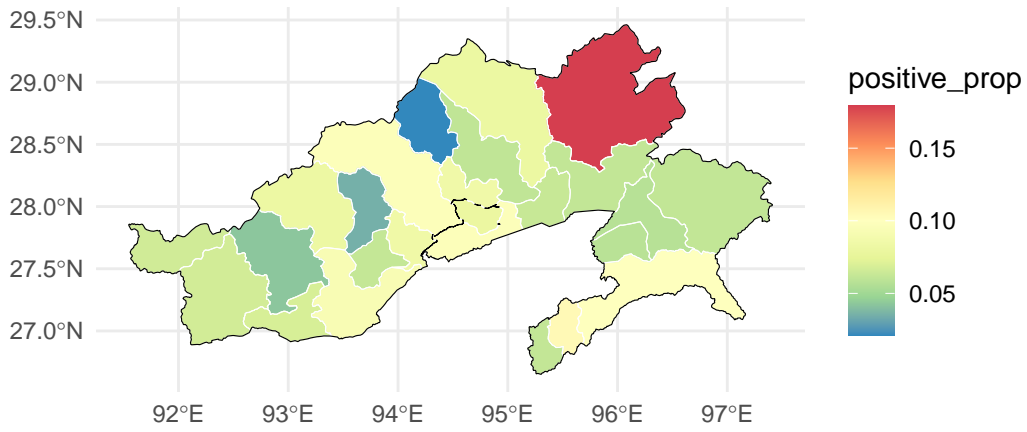
[[21]]

Test Positivity by Districts in the state of Puducherry



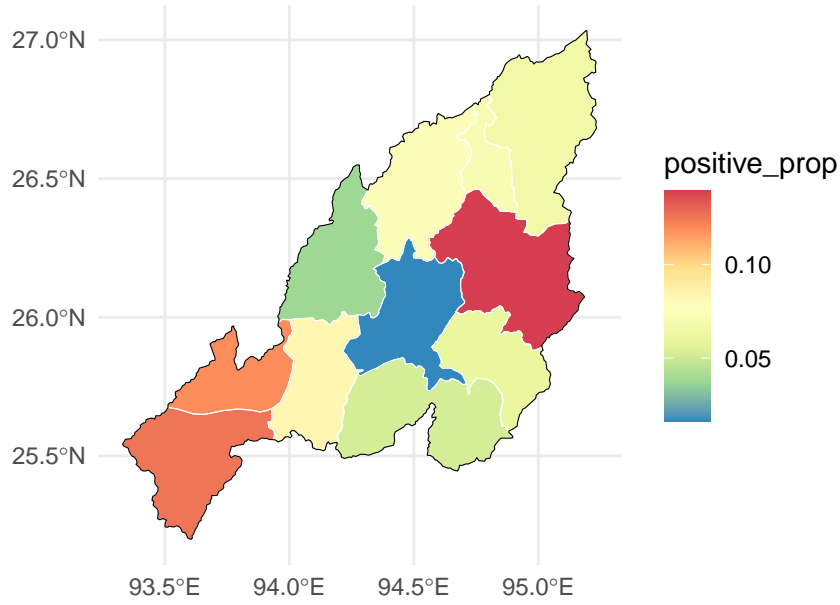
[[22]]

Test Positivity by Districts in the state of Arunachal Pradesh



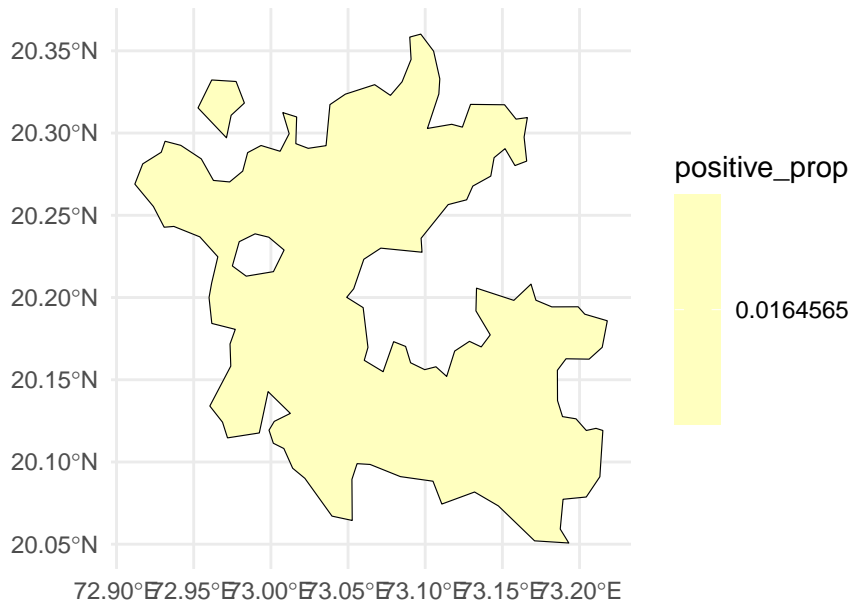
[[23]]

Test Positivity by Districts in the state of Nagaland



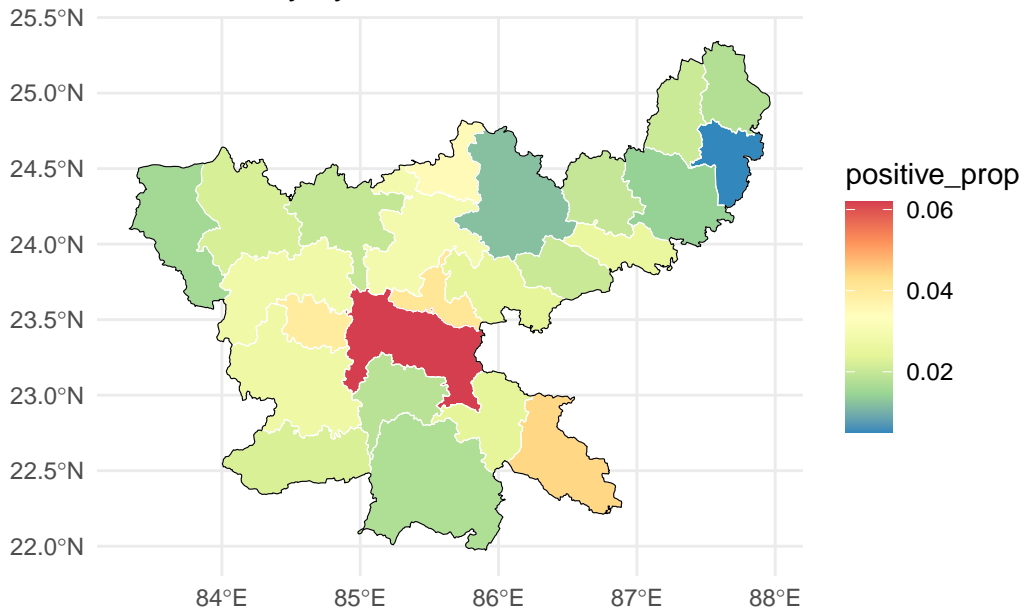
[[24]]

Test Positivity by Districts in the state of Dadra Nagar Haveli



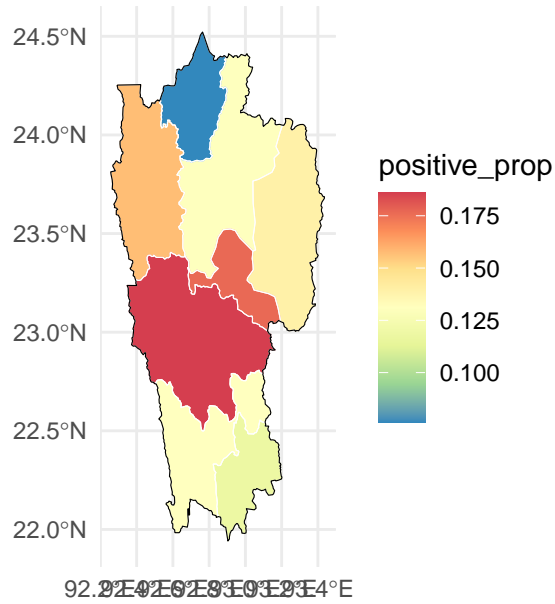
[[25]]

Test Positivity by Districts in the state of Jharkhand



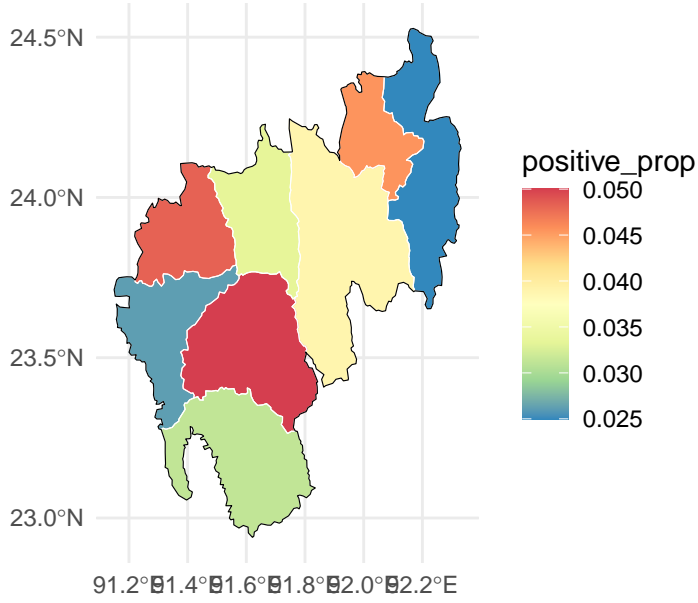
[[26]]

Test Positivity by Districts in the state of Mizora



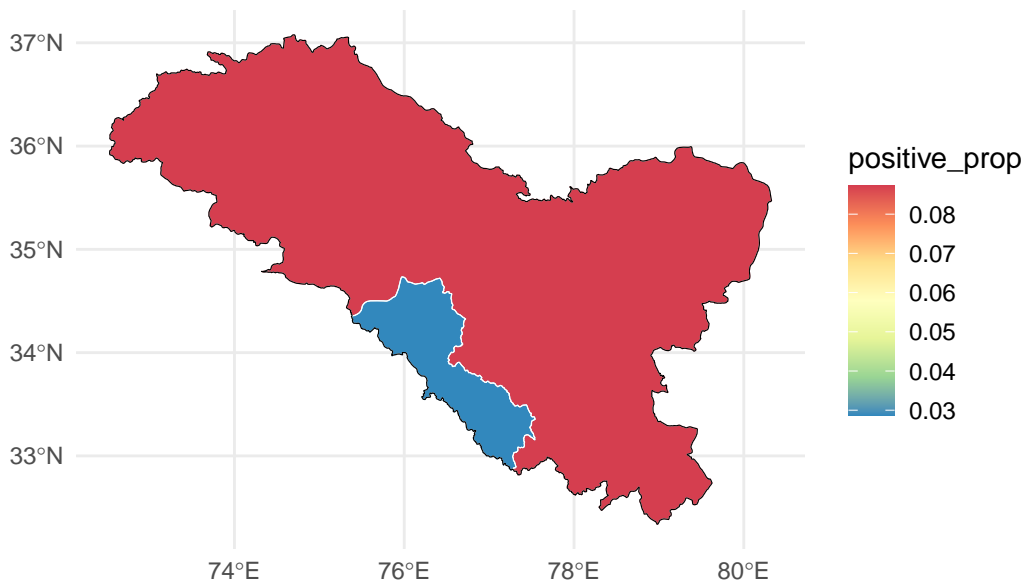
[[27]]

Test Positivity by Districts in the state of Tripura



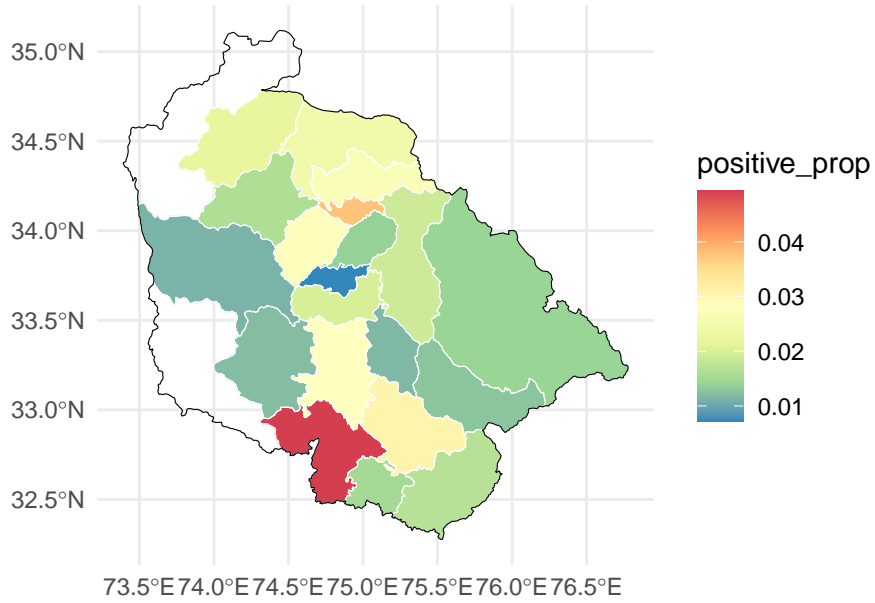
[[28]]

Test Positivity by Districts in the state of Ladakh



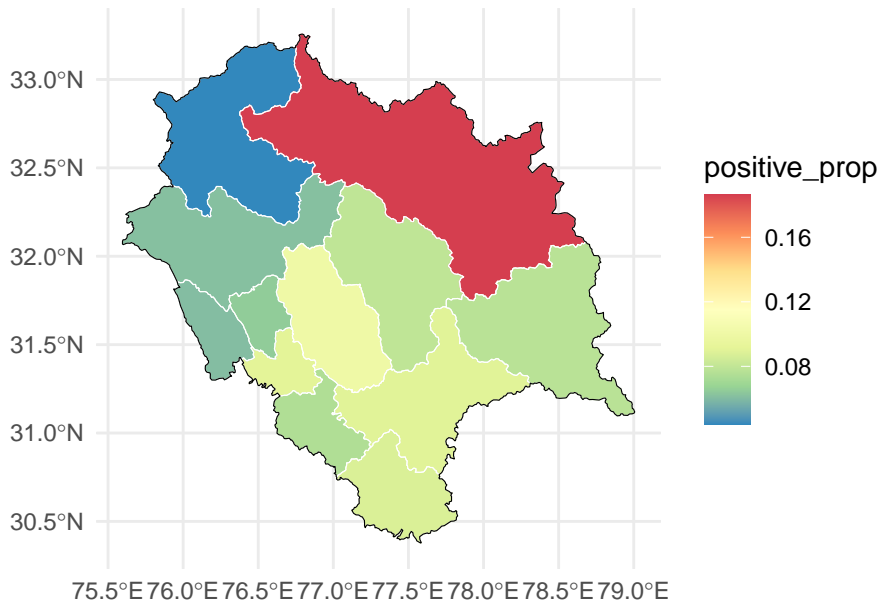
[[29]]

Test Positivity by Districts in the state of Jammu Kashmir



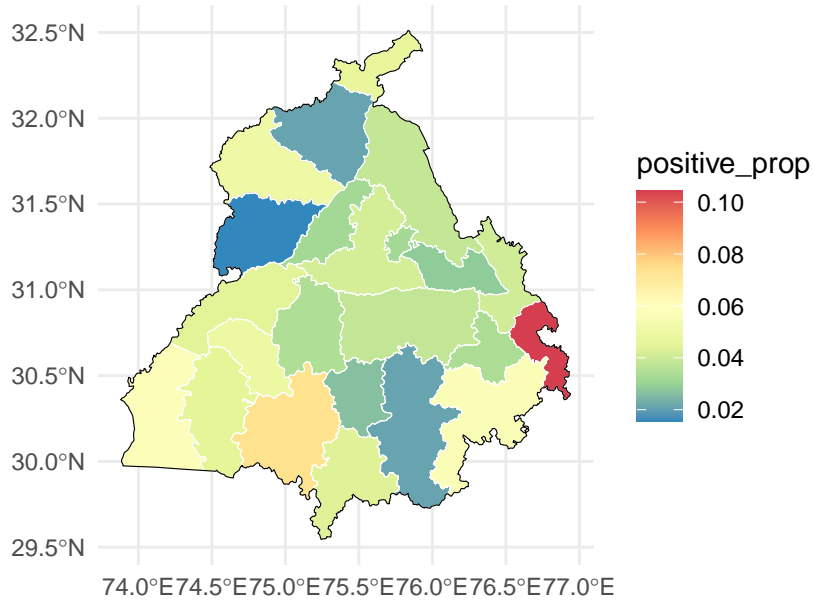
[[30]]

Test Positivity by Districts in the state of Himachal Pradesh



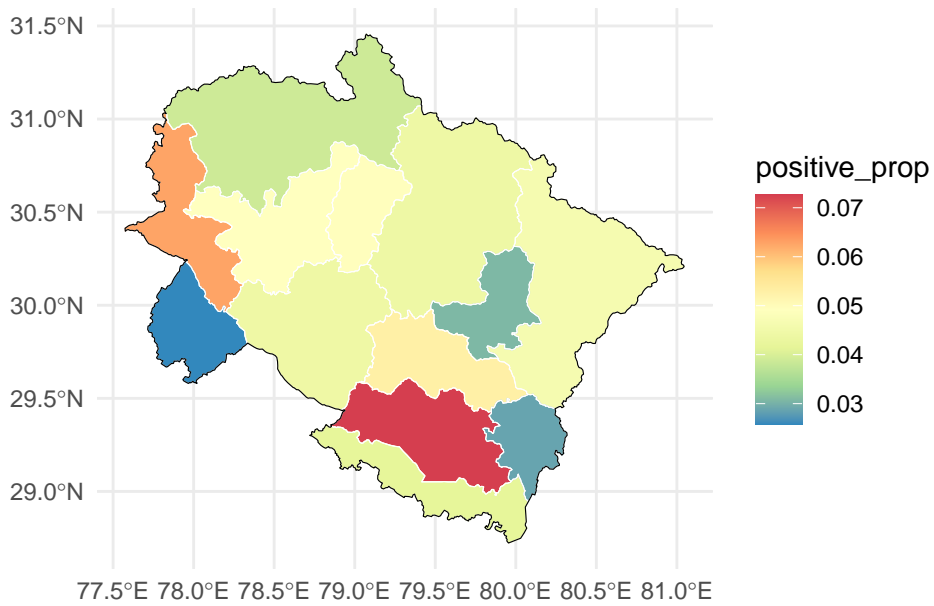
[[31]]

Test Positivity by Districts in the state of Punjab



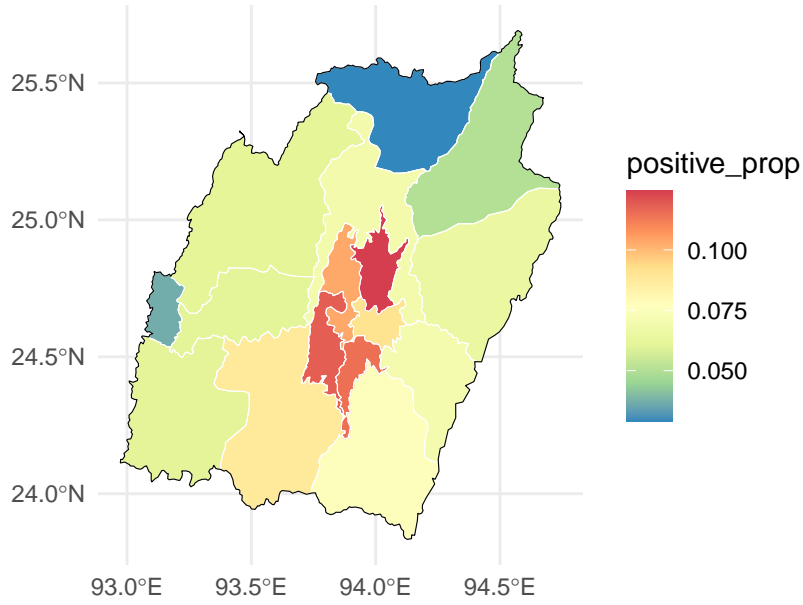
[[32]]

Test Positivity by Districts in the state of Uttarakhand



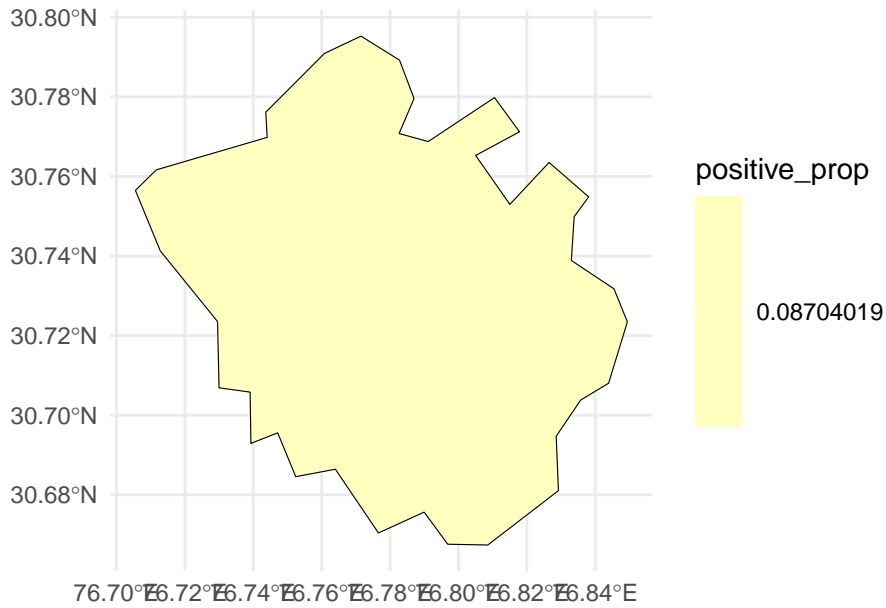
[[33]]

Test Positivity by Districts in the state of Manipur



[[34]]

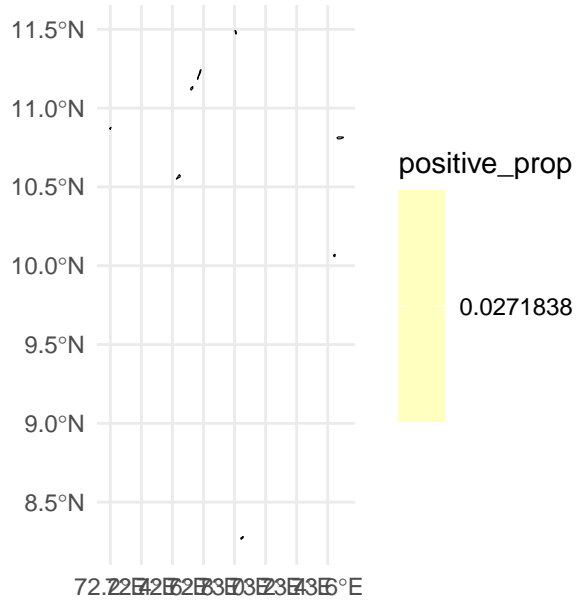
Test Positivity by Districts in the state of Chandigarh



[[35]]



### Test Positivity by Districts in the state of Lakshadweep



# 6 Conclusions

The project has immense translational potential, particularly in the context of public health policy and response in India. By synthesizing complex epidemiological data into comprehensible, actionable insights, the project aims to develop a framework for cleaning, management, analysis, interpretation and communication of large public health data. The framework allows for ‘data enrichment’ through data science and spatiotemporal techniques and enhances the overall efficiency and effectiveness of data usage in public health practice and policy making. This stands to significantly impact how public health authorities respond to ongoing and future epidemics.

The research team has presented a conference poster analysing this data. The presentation won the First Prize in the category. The details are provided below:



- Title: Navigating Challenges in Analyzing Large Public Health Datasets: Experiences working with the ICMR National COVID-19 Testing Database.
  - 4th National Conference of Epidemiology Foundation of India (EFICON 2023) Department of Community Medicine, Goa Medical College, Goa