EDA and Report

Ean Anciso Lance He Regi Acierto

2025-12-04

1 Data

The primary dataset for this analysis is the "List of Largest Cities" from Wikipedia, which compiles recent population estimates based on key definitions. A challenge in urban data analysis is the definition of "city," as political boundaries often differ from physical settlements.

To ensure clarity, we differentiate between two key terms used throughout this report:

City Proper: The population living within the city's legally defined administrative boundaries. This definition is strict and governed by a single local authority, often excluding suburbs or commuters.

Urban Area: A definition based on the continuous built-up environment, physical settlements, and population density. This metric is less rigid and often better reflects the actual "lived" city.

2 Global Overview

Table 1 presents a frequency table ranking countries by the number of their cities that appear in the top-tier of our raw dataset. While many countries appear only once, a select few dominate the list, highlighting the uneven distribution of global urbanization.

China leads the dataset, which aligns with its status as one of the world's most populous nations. India and the United States follow, showing that large megacities are a feature of major economies regardless of development path. Notably, Japan, Brazil, and Indonesia also feature prominently in the top 6. This is significant as it suggests that geography and historical urbanization patterns play a major role alongside total population size.

Table 1: Top 10 Countries by Number of Cities

Country	entries	total_pop	total_city_pop	total_urban_pop
China	18	184151927	264544408	216167475
India	9	127865581	69054182	139999000
United States	9	70248568	22409673	84186000
Japan	4	59073817	20149562	64394000
Brazil	3	34422126	21274580	41006000
Indonesia	3	57666467	15700316	47515000

Country	entries	total_pop	total_city_pop	total_urban_pop
Egypt	2	32833059	15486760	25008000
Mexico	2	22757212	10595565	27329000
Pakistan	2	36579020	28742135	32555000
Russia	2	19907753	18801911	22777000

Table 1: Countries Ranked by amount of Entries

3 Statistical Summary

We performed a comparative analysis of the top 6 countries to understand the statistical distribution of their urban populations.

Table 2 and Table 3 summarize the population and area statistics for Urban Areas and City Proper, respectively. A key observation is the difference in variability (Standard Deviation). Urban Areas tend to have more consistent definitions of density compared to the administrative "City Proper" definitions, which can vary wildly depending on local laws.

Table 2: Urban Area Statistics (Top 6 Countries)

Country	Pop_Mean	Pop_SD	Area_Mean	Area_SD
Brazil	13668667	8927824	2319	1209
China	12009304	6963228	2050	1166
India	15555444	8891563	1095	636
Indonesia	15838333	15520526	1648	1657
Japan	16098500	15380354	3865	3219
United States	9354000	5448314	6367	2461

Table 3: City Proper Statistics (Top 6 Countries)

Country	Pop_Mean	Pop_SD	Area_Mean	Area_SD
Brazil	7091527	4899796	1024	619
China	14696912	6647019	16387	19780
India	7672687	4372810	595	369
Indonesia	5233439	4268480	389	253
Japan	5037390	5671442	771	948
United States	2489964	2638432	668	488

Visual Analysis: Population vs. Area Trends

The following plots visualize the relationship between Area (x-axis) and Population (y-axis). Note that both axes use a logarithmic scale to accommodate the massive range in city sizes.

Figure 1 shows the relationship for Urban Areas. We observe a generally positive correlation—as physical area grows, population grows.

Figure 1: Urban Population by Area

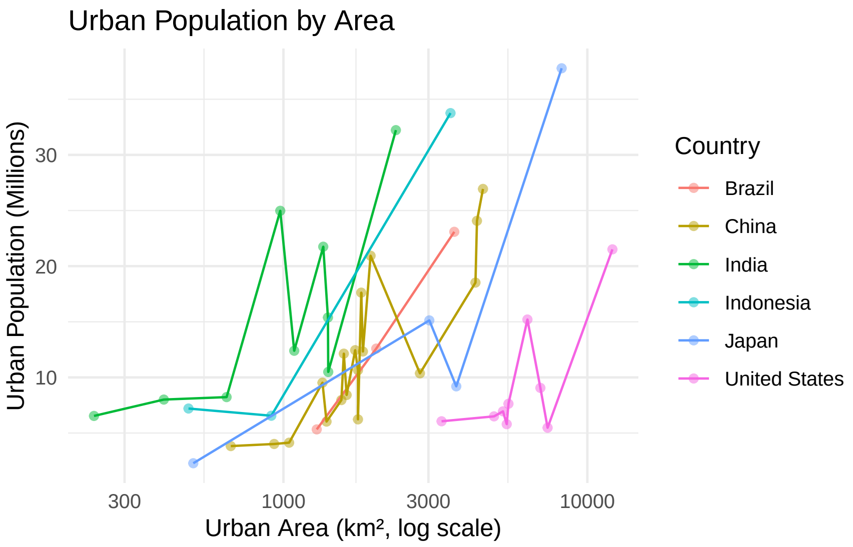
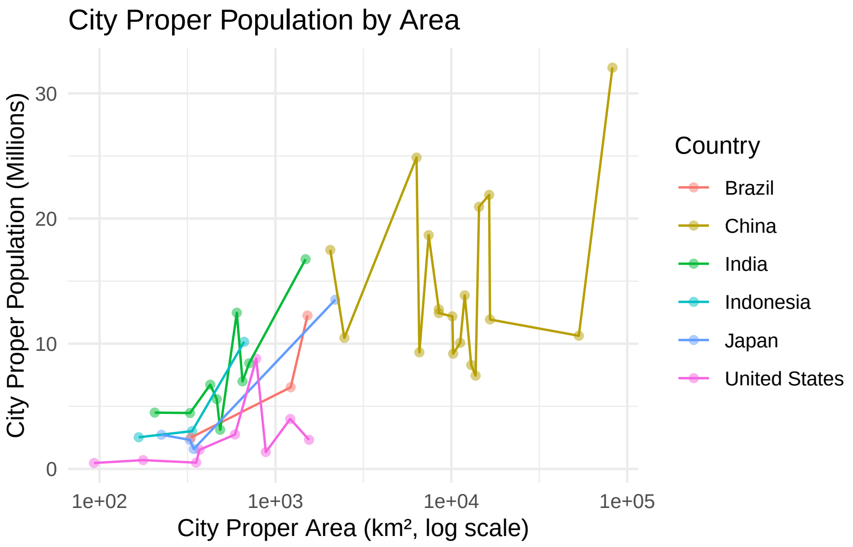


Figure 2: City Proper Population by Area



In contrast, Figure 2 shows the data for City Proper. This plot is noticeably more scattered. This supports the hypothesis that "City Proper" is a political definition rather than a geographic one, leading to inconsistent relationships between size and population across different nations.

4 Visual Analysis: Density vs Area Trends

Finally, we analyze the relationship between Area and Density. Figure 3 displays this for City Proper. The plot exhibits an inverse trend: as area increases, density tends to decrease.

Notably, China and the United States display cities with very large administrative areas but relatively lower densities. In contrast, Japan, Indonesia, and India cluster around smaller areas with extremely high densities. This discrepancy often arises because some countries include vast suburban or rural zones within their "City Proper" limits, inflating the area and artificially lowering the density.

Figure 3: City Proper, Density vs Area for Top 6 Countries with amount of large cities

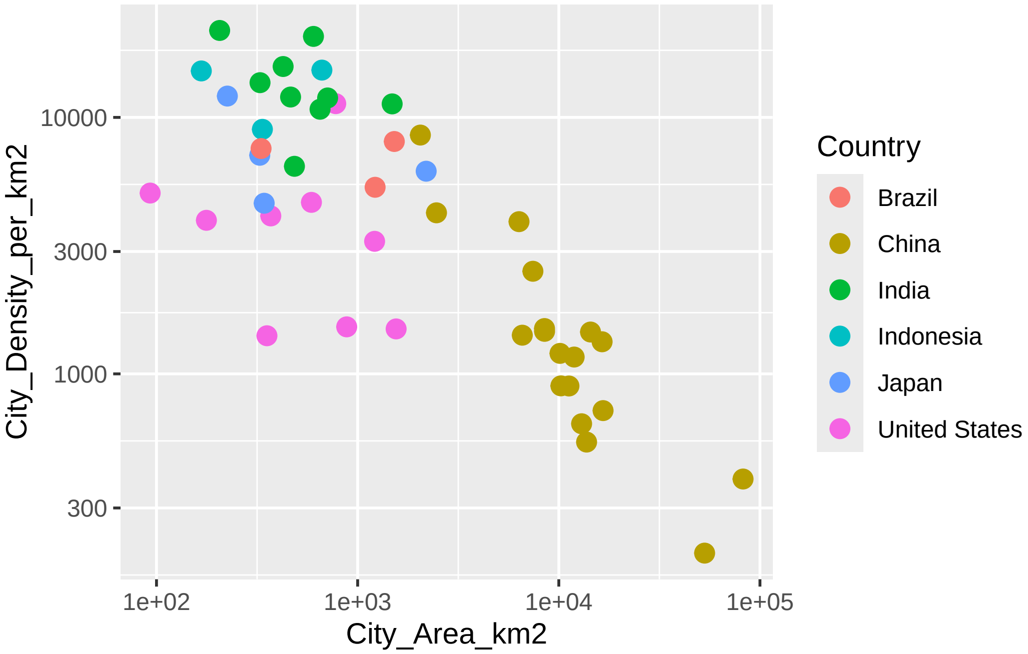
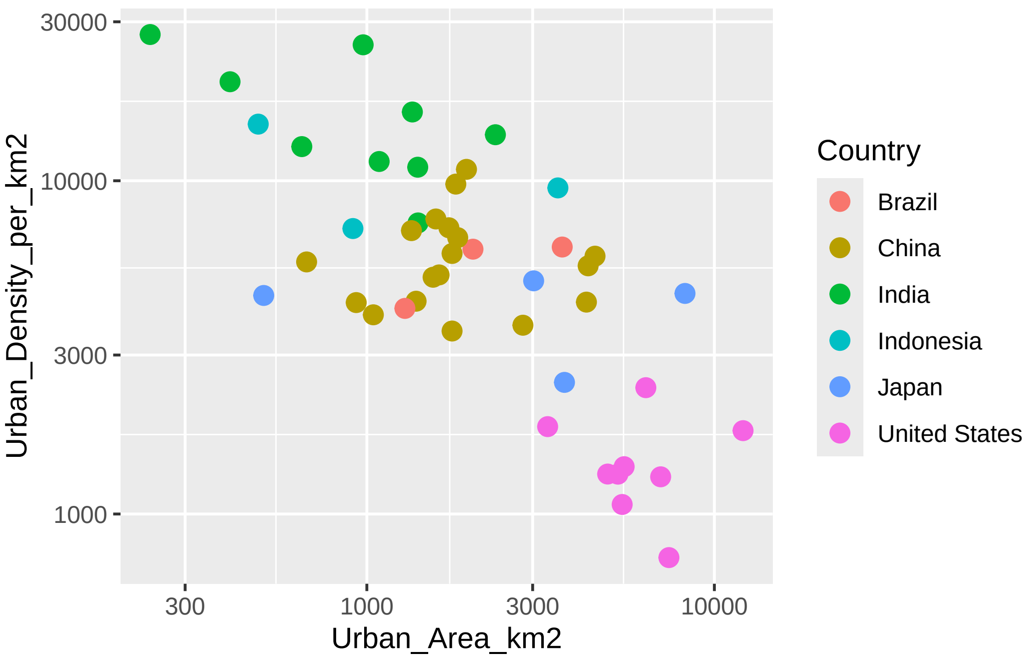


Figure 4 repeats this analysis for Urban Area. Here, the data is slightly more consistent. The United States still distinctively occupies the "High Area, Low Density" quadrant, reflecting its sprawling, car-centric infrastructure. Meanwhile, Chinese cities shift towards moderate area and density values when looking at the built-up environment rather than administrative borders.

Conclusively, Urban Area appears to be a more robust and comparable measure of a city's actual demographic footprint than City Proper.

Figure 4: Urban Area, Density vs Area for Top 6 Countries with amount of large cities



5 Code Appendix

```
# Main Author: Lance He, Reviewer(s): Ean Anciso, Regi Acierto

library(rvest)
library(dplyr)
library(stringr)
library(ggplot2)
library(knitr)
library(tidyverse)

url <- "https://en.wikipedia.org/wiki/List_of_largest_cities"
webpage <- read_html(url)
tables <- html_table(webpage, fill = TRUE)

table_index <- which(sapply(tables, function(t) "Jakarta" %in% t[[1]]))
city_table <- tables[[table_index]]

colnames(city_table) <- c(
  "City", "Country", "UN_Estimate_Pop", "Definition",
  "City_Pop", "City_Area_km2", "City_Density_per_km2",       # City Proper
  "Urban_Pop", "Urban_Area_km2", "Urban_Density_per_km2",    # Urban Area
  "Metro_Pop", "Metro_Area_km2", "Metro_Density_per_km2"     # Metropolitan
)

clean_data <- city_table %>%
  # Remove Garbage Header Row
  slice(-1) %>%

  select(
    City, Country, UN_Estimate_Pop,
    City_Pop, City_Area_km2, City_Density_per_km2,
    Urban_Pop, Urban_Area_km2, Urban_Density_per_km2
  ) %>%

  mutate(across(everything(), ~str_remove_all(., "\\[.*?\\]"))) %>%
  mutate(across(everything(), ~str_remove_all(., ","))) %>%

  mutate(across(everything(), ~na_if(str_trim(.), ""))) %>%
  mutate(across(everything(), ~na_if(str_trim(.), "-"))) %>%
  mutate(across(everything(), ~na_if(str_trim(.), "-"))) %>%
  mutate(across(everything(), ~na_if(str_trim(.), "-"))) %>%

  filter(
    !is.na(City_Pop), !is.na(City_Area_km2), !is.na(City_Density_per_km2),
    !is.na(Urban_Pop), !is.na(Urban_Area_km2), !is.na(Urban_Density_per_km2)
  ) %>%

  mutate(across(3:9, as.numeric)) %>%

  mutate(Country = str_trim(Country))

target_countries <- c("United States", "China", "India", "Japan", "Brazil", "Indonesia")
country_data <- clean_data %>%
  filter(Country %in% target_countries)
# Main Author: Ean Anciso, Reviewer(s): Regi Acierto, Lance He

firstTable <- clean_data %>%
  group_by(Country) %>%
  summarise(
    entries = n(),
    total_pop = sum(UN_Estimate_Pop, na.rm = TRUE),
    total_city_pop = sum(City_Pop, na.rm = TRUE),
    total_urban_pop = sum(Urban_Pop, na.rm = TRUE)
  ) %>%
  arrange(desc(entries))

kable(head(firstTable, 10), caption = "Top 10 Countries by Number of Cities")
# Main Author: Lance He, Reviewer(s): Ean Anciso, Regi Acierto

# Table A: Urban Area Statistics
urban_area_stats <- country_data %>%
  group_by(Country) %>%
  summarise(
    Pop_Mean = mean(Urban_Pop, na.rm = TRUE),
    Pop_SD   = sd(Urban_Pop, na.rm = TRUE),
    Area_Mean = mean(Urban_Area_km2, na.rm = TRUE),
    Area_SD   = sd(Urban_Area_km2, na.rm = TRUE)
  )

kable(urban_area_stats, digits = 0)
# Main Author: Regi Acierto, Reviewer(s): Lance He, Ean Anciso

# Table B: City Proper Statistics
city_proper_stats <- country_data %>%
  group_by(Country) %>%
  summarise(
    Pop_Mean = mean(City_Pop, na.rm = TRUE),
    Pop_SD   = sd(City_Pop, na.rm = TRUE),
    Area_Mean = mean(City_Area_km2, na.rm = TRUE),
    Area_SD   = sd(City_Area_km2, na.rm = TRUE)
  )
```

```r
kable(city_proper_stats, digits = 0)
# Main Author: Lance He, Reviewer(s): Ean Anciso, Regi Acierto

# Plot 1: Urban Population by Area
urban_data <- country_data # Uses the filtered data from setup

urban_data %>%
  ggplot(
    aes(
      x = Urban_Area_km2,
      y = Urban_Pop / 1000000,
      color = Country,
      group = Country
    )
  ) +
  geom_point(alpha = 0.5) +
  geom_line() +
  scale_x_log10() +
  labs(
    title = "Urban Population by Area",
    x = "Urban Area (km², log scale)",
    y = "Urban Population (Millions)"
  ) +
  theme_minimal()
# Main Author: Regi Acierto, Reviewer(s): Ean Anciso, Lance He

# Plot 2: City Proper Population by Area
city_proper_data <- country_data # Uses the filtered data from setup

city_proper_data %>%
  ggplot(
    aes(
      x = City_Area_km2,
      y = City_Pop / 1000000,
      color = Country,
      group = Country
    )
  ) +
  geom_point(alpha = 0.5) +
  geom_line() +
  scale_x_log10() +
  labs(
    title = "City Proper Population by Area",
    x = "City Proper Area (km², log scale)",
    y = "City Proper Population (Millions)"
  ) +
  theme_minimal()
```

```r
# Main Author: Ean Anciso, Reviewer(s): Lance He, Regi Acierto

Top_entries <- clean_data %>% # Get target countries(Top 6)
  filter(Country %in% c("China", "India", "United States", "Japan", "Brazil", "Indonesia"))

ggplot(
  Top_entries,
  aes(
    City_Area_km2,
    City_Density_per_km2,
    color = Country
  ),
) +
  geom_point(size = 3) +
  scale_x_log10() + # Scale the scale values
  scale_y_log10()

# Main Author: Ean Anciso, Reviewer(s): Lance He, Regi Acierto

Top_entries <- clean_data %>% # Get target countries(Top 6)
  filter(Country %in% c("China", "India", "United States", "Japan", "Brazil", "Indonesia"))

ggplot(
  Top_entries,
  aes(
    Urban_Area_km2,
    Urban_Density_per_km2,
    color = Country
  ),
) +
  geom_point(size = 3) +
  scale_x_log10() + # Scale the scale values
  scale_y_log10()
```