Exploring the Correlation between Healthcare Access and Mortality Rate

Mahija Mogalipuvvu

Dataset

The Global Health Statistics dataset, sourced from Kaggle, consists of an extensive collection of health-related data from 2000 to present-day 2024 and covers twenty countries. Within the dataset itself, there is a diverse range of variables, including mortality rates, recovery rate, availability of vaccines/treatment, healthcare access, and the disease category. What drew me to this dataset particularly was both its depth and breadth. Oftentimes, whether it be in research or the quantitative biology field, there is typically limited, disorganized data or a general lack of data, specifically in regards to healthcare. Being able to work with a detailed and comprehensive dataset to explore complex health patterns and trends on a global scale is what really interested me.

Hypothesis

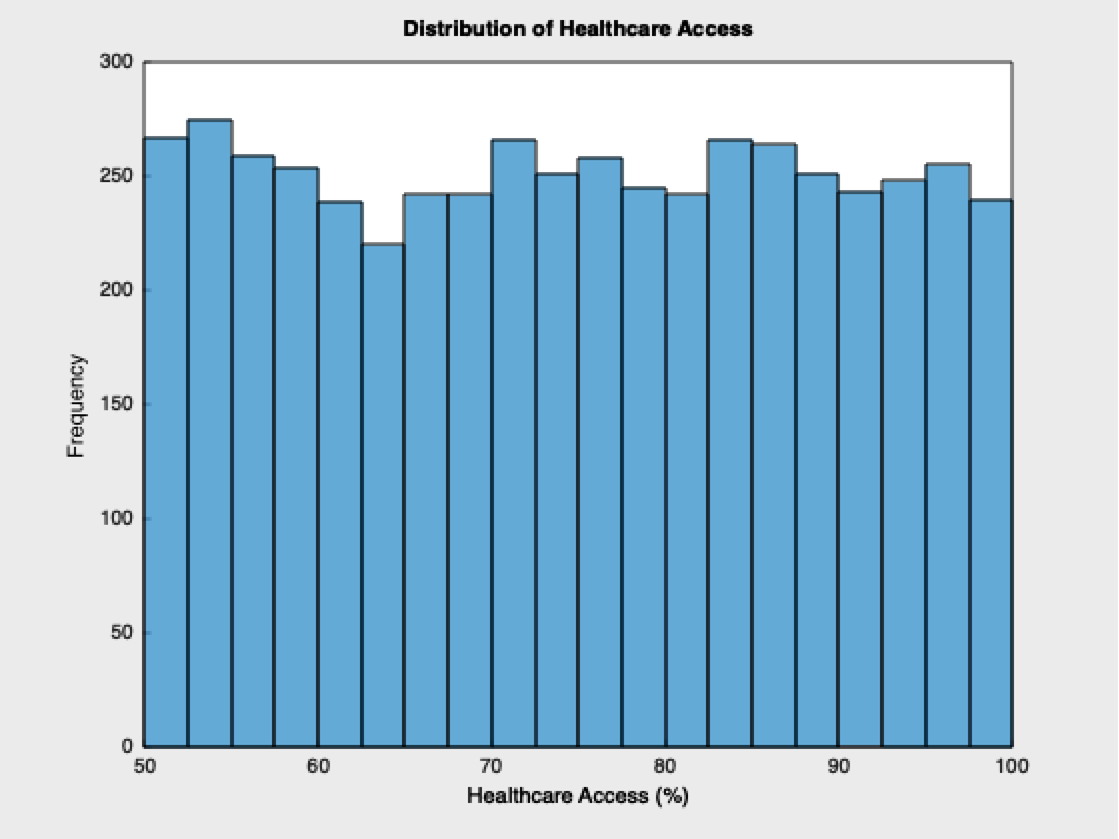
The question I hoped to answer with my quantitative analysis is whether countries with higher healthcare access percentages have lower mortality rates for both infectious and non-communicable diseases. I hypothesized that countries with higher healthcare access percentages would have lower mortality rates however, the trend may be weaker for infectious diseases. This dataset is ideal for addressing this hypothesis, as it includes detailed metrics on healthcare access, mortality rates and a host of possible confounding variables. By analyzing these variables, we can identify potential correlations while also controlling for external factors as to ensure they don't affect the relationship between healthcare access and mortality outcomes.

Data Analysis

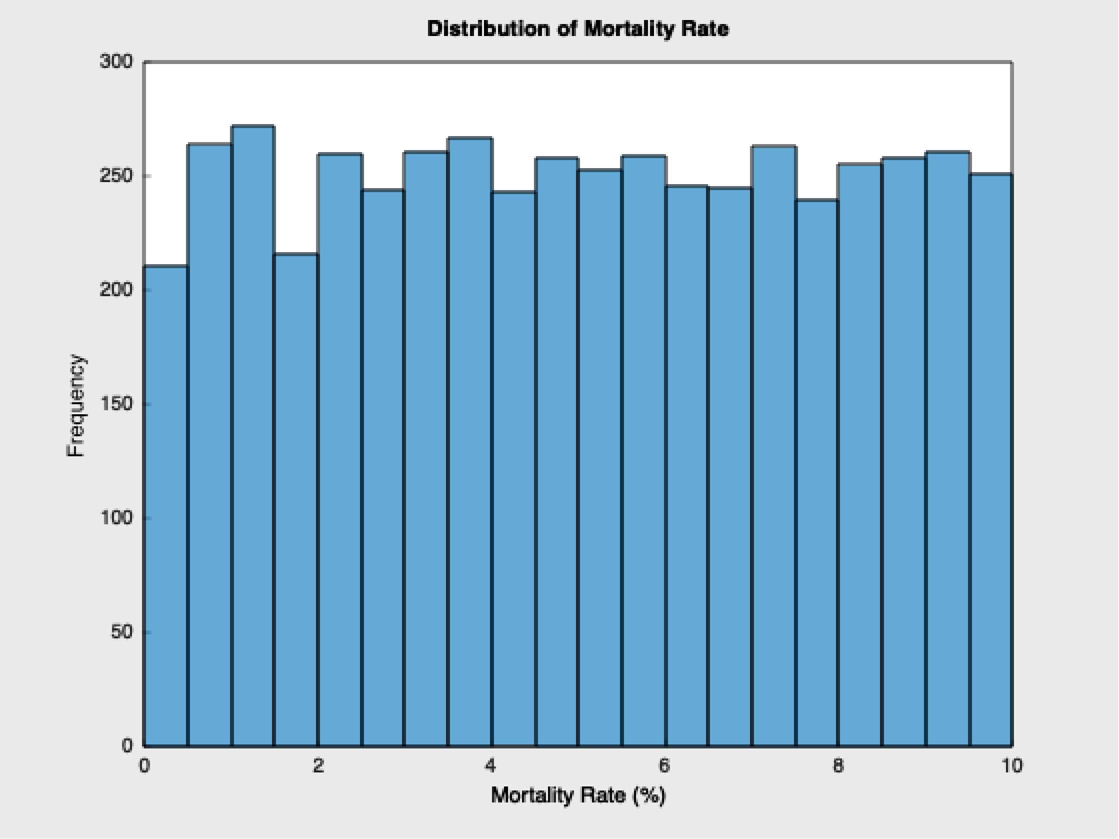
To begin data analysis, we had to read in, parse and filter the dataset to prepare it for meaningful analysis. Additionally, this analysis was designed to eliminate potential confounding variables and ensure the validity of the results. Data from cases from the year 2024, where treatment options were available and recovery rates were above the third quartile (86.8%) were the only ones included in this project. This filtering process was critical. First, excluding diseases without available treatments helped avoid skewed results, as those cases would inherently lack a connection between healthcare access and mortality rates. Second, focusing on diseases with high recovery rates provided a more reliable metric for assessing the relationship between healthcare access and mortality, as these conditions are more likely to show clear improvements with effective healthcare. Lastly, we focused exclusively on data from 2024 because the rate at which healthcare improves differs across countries. By limiting the analysis to a single year, we minimized any bias that could arise from disparities in the pace of healthcare development across countries over time. This approach ensured that the analysis focused on the relationship between healthcare access and mortality rates under more controlled conditions.

After filtering out the data, we divided the cases into "Infectious" and "Non-Communicable", or NCD based on their disease category. Then, we employed multiple techniques to investigate the relationship between healthcare access and mortality rates. For instance, we began by creating histograms to analyze the distributions of healthcare access and mortality rates, revealing patterns in variability and central tendencies. Next, we calculated the country-level averages of mortality rates for the broader disease type, Infectious and Non-Communicable (NCD), and compared them using bar charts to highlight any cross-country differences. To further investigate the relationship between healthcare access and mortality rates, we performed linear regression separately for each disease type. This allowed us to calculate the slope, intercept, and R-squared values, which measure the strength and direction of the relationship. Finally, we visualized these trends using scatter plots with regression lines.

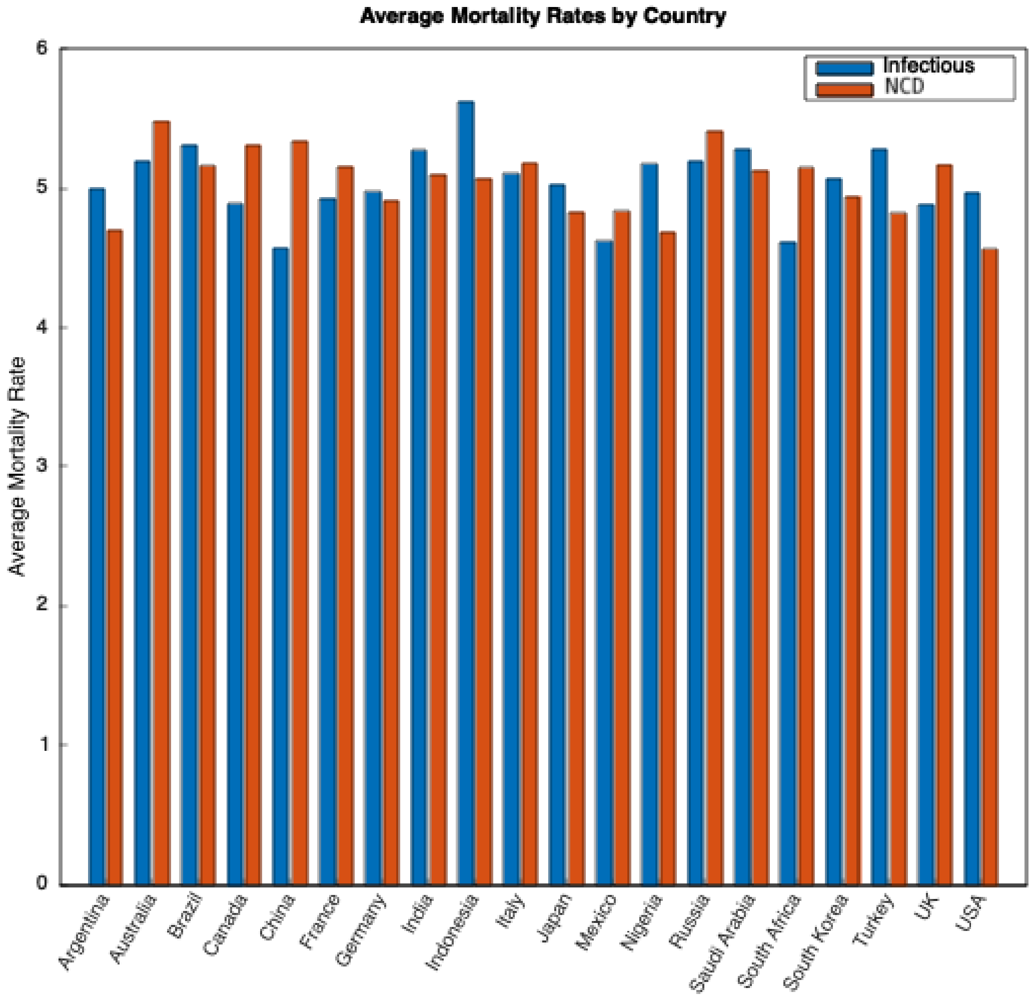
Results



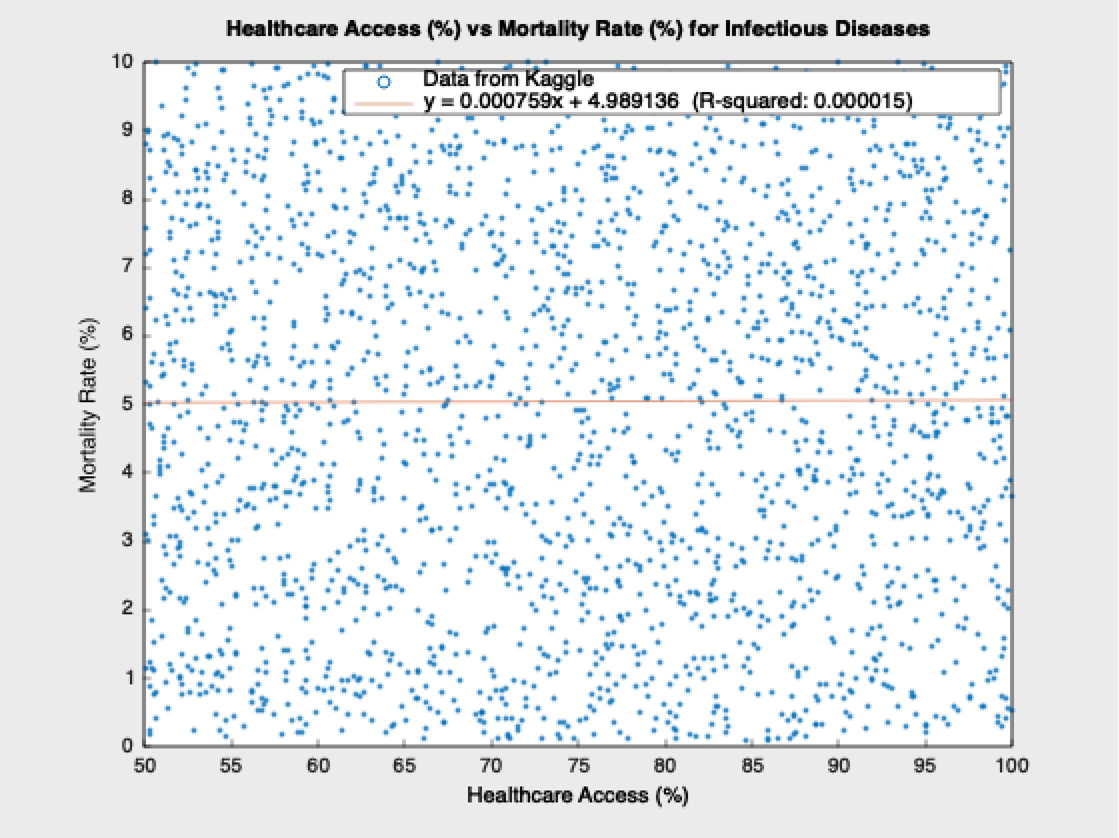
The histogram above displays the distribution of healthcare access percentages across the dataset, showing how frequently different levels of healthcare access occur. The x-axis represents healthcare access percentages, ranging from 50% to 100%, while the y-axis represents the frequency of occurrences for each range. The relatively uniform distribution indicates that healthcare access is well spread across different levels within the dataset, with no extreme clustering at either end. A balanced and well-distributed dataset for healthcare access ensures that the analysis will not be biased.



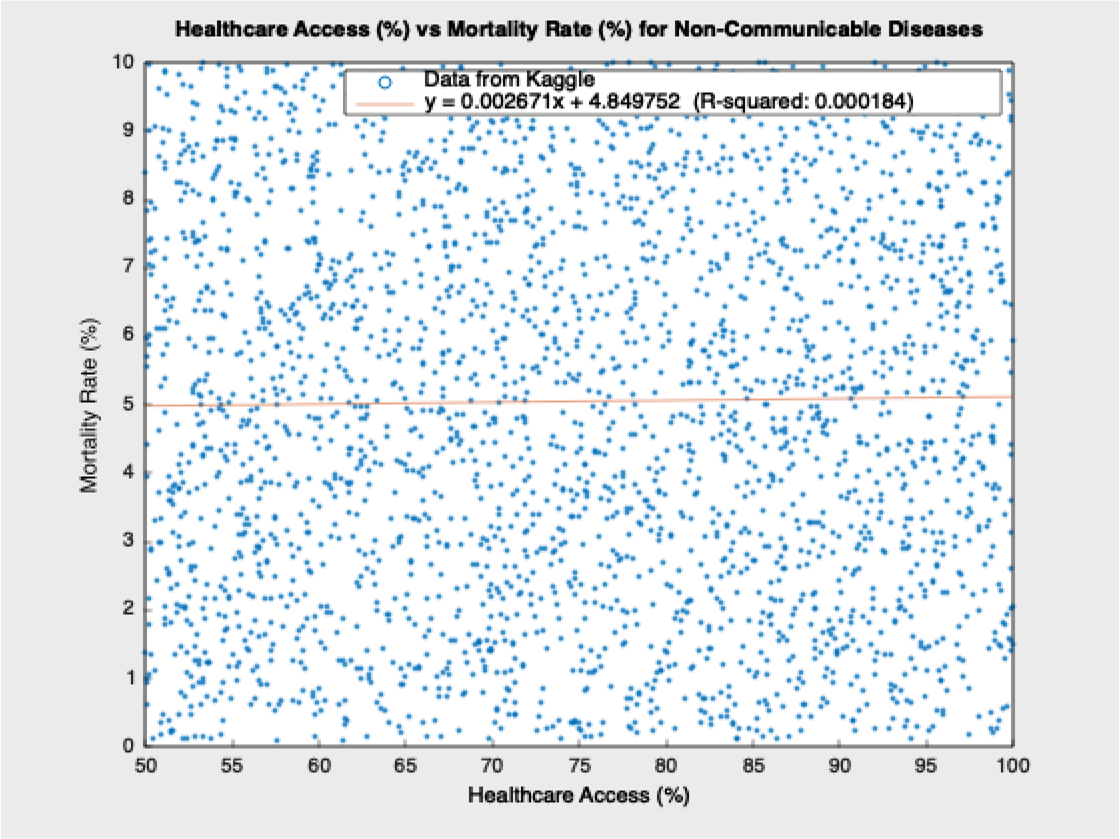
The histogram above shows the distribution of mortality rates (%) across the dataset, with mortality rates ranging from 0% to 10%. The x-axis represents the mortality rate percentages, while the y-axis represents the frequency of observations within each range. The relatively uniform distribution suggests that mortality rates are spread evenly across the dataset without any extreme concentrations at particular values. Similarly, an even and well-distributed dataset ensures that the analysis is not skewed toward specific ranges.



The bar graph above compares the average mortality rates for Infectious diseases (blue) and Non-Communicable Diseases (NCDs) (red) across different countries. The y-axis represents the average mortality rate, while the x-axis lists the countries included in the analysis. The graph shows that mortality rates vary across countries and between the two disease categories, with noticeable differences in some regions.



The scatter plot above shows the relationship between Healthcare Access (%) (x-axis) and Mortality Rate (%) (y-axis) for Infectious Diseases. The linear regression line is on the plot with the equation y=0.000759x+4.989136, and the R-squared value is approximately 0.000015, indicating an extremely weak relationship between healthcare access and mortality rates for infectious diseases.



The scatter plot above shows the relationship between Healthcare Access (%) and Mortality Rate (%) for Non-Communicable Diseases (NCDs). The linear regression line is included with the equation y=0.002671x+4.849752, and the R-squared value is approximately 0.000184. Similar to the results for infectious diseases, the extremely low R-squared value indicates that healthcare access explains very little of the variation in mortality rates for NCDs.

Conclusion

The driving question of this analysis was whether countries with higher healthcare access have lower mortality rates for both infectious and non-communicable diseases (NCDs). To address this question, we utilized the robust Global Health Statistics dataset from Kaggle that was filtered to include data only from 2024, with treatment options available and recovery rates above the third quartile (86.8%) to control for some confounding variables. We used a variety of analytical techniques, like histograms, bar charts, and scatter plots with regression analysis, to explore the relationship between healthcare access and mortality rates.

The histograms for healthcare access and mortality rates revealed that both variables were evenly distributed across the dataset. This ensured that the analysis was not biased toward specific ranges of healthcare access or mortality outcomes. The bar chart, which compared average mortality rates for infectious diseases and NCDs across countries, revealed that mortality rates for infectious diseases and NCDs are relatively close on average, suggesting that healthcare access might impact both disease types in similar ways.

The core of the analysis relied on scatter plots and linear regression models. For infectious diseases, the regression analysis resulted in an equation of y=0.000759x+4.989136, with an R-squared value of 0.000015, indicating virtually no correlation between healthcare access and mortality rates. Similarly, for non-communicable diseases (NCDs), the regression equation was y=0.002671x+4.849752, with an R-squared value of 0.000184. Again, this result showed an extremely weak relationship between healthcare access and mortality rates. The near-zero R-squared values and flat trend lines in both scatter plots suggest that healthcare access alone does not significantly explain the variation in mortality rates for either disease type.

In conclusion, the results do not support the hypothesis that higher healthcare access is associated with lower mortality rates for infectious or non-communicable diseases. While the dataset was comprehensive and the analysis was conducted under controlled conditions, the findings may suggest that mortality rates are influenced by other additional factors beyond healthcare access alone. Variables like quality of healthcare services, socioeconomic disparities, disease management practices, environmental conditions, and longitudinal healthcare may have a more critical effect on mortality outcomes. Further research could incorporate these factors to better understand the relationship between healthcare and mortality rates.

Supplementary Material: Code

To download the data and compile and run the code:

- download the Global Health Statistics dataset from Kaggle
  - https://www.kaggle.com/datasets/malaiarasugraj/global-health-statistics/data
- save the dataset as a CSV file in the same directory where you will work
- download matplotplusplus library
- to compile, navigate to your working directory and execute the following commands:

  -
    ```
    clang++ -std=c++17 -I/opt/homebrew/include -L/opt/homebrew/lib -lmatplot project.cpp -o project -g
    ```

  -
    ```
    ./project
    ```

- using #include statements to bring in existing files/libraries
- using namespace x to simplify code by not having to use prefixes (ex. std:: or matplot::)

```
#include <vector>
#include <string>
#include <iostream>
#include <fstream>
#include <sstream>
#include <matplot/matplot.h>

using namespace std;
using namespace matplot;
```

- created a struct to easily hold any relevant health information for analysis
    - country, disease category, mortality rate and healthcare access

```
struct healthData {
   string country, diseaseCategory;
   double mortalityRate, healthcareAccess;
};
```

- helper method that calculates linear regression
    - takes in two vectors of data: independent variable and dependent variable
    - calculates linear regression and r-squared using the following equations:

$$y = \alpha + \beta x$$

$$\widehat{\alpha} = \bar{y} - (\widehat{\beta}\,\bar{x}),$$

$$\widehat{\beta} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

$$R^2 = 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}}$$

$$SS_{\text{res}} = \sum_i (y_i - f_i)^2$$

$$SS_{\text{tot}} = \sum_i (y_i - \bar{y})^2$$

    - returns a pair of a pair of doubles and a double
        - the pair of doubles consisting of slope and intercept
        - the double is the r-squared value

```
pair<pair<double, double>, double> linearRegression (const vector<double>& x, const
vector<double>& y) {
   double numerator = 0.0;
   double denominator = 0.0;
   double totSS = 0.0;
   double resSS = 0.0;
   vector<double> predicted;


   double xMean = accumulate(x.begin(), x.end(), 0.0) / x.size();
   double yMean = accumulate(y.begin(), y.end(), 0.0) / y.size();

   for (int i = 0; i < x.size(); i++) {
       numerator += (x[i] - xMean) * (y[i] - yMean);
       denominator += pow(x[i] - xMean, 2);
   }

   double beta = numerator / denominator;
   double alpha = yMean - (beta * xMean);

   for (int i = 0; i < x.size(); i++) {
       double predictedValue = alpha + (beta * x[i]);
```

```
        predicted.push_back(predictedValue);

        resSS += pow((y[i] - predictedValue), 2);
        totSS += pow((y[i] - yMean), 2);
    }
    double rSquared = 1 - (resSS / totSS);
    return {{beta, alpha}, rSquared};
}
```

- reading in and parsing the data into vector of healthData objects
- assigning Infectious and Non-Communicable based on disease category
- filtering data not from 2024, without treatment options and a recovery rate lower than 86.8 out

```
int main() {
    string fileName = "Global Health Statistics.csv";

    ifstream file(fileName);
    if (!file.is_open()) {
        cerr << "error! can't open file " << "\n";
        return 1;
    }

    vector<healthData> data;
    string line;

    getline(file, line);
    while (getline(file, line)) {
        stringstream ss(line);

        string temp, country, disease, treatment;
        double year, mortality, healthcare, recoveryRate;

        getline(ss, country, ',');
        getline(ss, temp, ',');

        year = stoi(temp);

        int index = 2;

        while(getline(ss, temp, ',')) {
            if(index == 3) {
                if(temp == "Respiratory" || temp == "Parasitic" || temp == "Bacterial" || temp == "Viral" || temp == "Infectious")
                    disease = "Infectious";
                else
                    disease = "Non-Communicable";

            } else if(index == 6) {
```

```
            mortality = stod(temp);
        } else if(index == 10) {
            healthcare = stod(temp);
        } else if (index == 15) {
            treatment = temp;
        } else if (index == 16) {
            recoveryRate = stod(temp);
        }
            index++;
        }
        if(year == 2024 && treatment == "Yes" && recoveryRate >= 86.8)
            data.push_back({country, disease, mortality, healthcare});
}
```

- separating data from a vector of healthData objects into 2 vectors of only doubles(healthcare access and mortality rate) for each disease type(infectious and non-communicable)
- creating maps for each disease type to help calculate average mortality for each country
    - key: country name
    - value: pair of double and int
        - double: sum of all mortality rates for specific country and disease type
        - int: number of cases for specific country and disease type

```
vector<double> healthcareInfectious, mortalityInfectious;
vector<double> healthcareNCD, mortalityNCD;

map<string, pair<double, int>> countryMortalityInfectious;
map<string, pair<double, int>> countryMortalityNCD;

for (const auto& row : data) {
    if (row.diseaseCategory == "Infectious") {
        healthcareInfectious.push_back(row.healthcareAccess);
        mortalityInfectious.push_back(row.mortalityRate);

        countryMortalityInfectious[row.country].first += row.mortalityRate;
        countryMortalityInfectious[row.country].second++;

    } else {
        healthcareNCD.push_back(row.healthcareAccess);
        mortalityNCD.push_back(row.mortalityRate);

        countryMortalityNCD[row.country].first += row.mortalityRate;
        countryMortalityNCD[row.country].second++;
    }
}
```

- combining vectors for each disease type into larger vector for all values regardless of disease type

```
vector<double> healthcare(healthcareInfectious);
healthcare.insert(healthcare.end(), healthcareNCD.begin(), healthcareNCD.end());

vector<double> mortality(mortalityInfectious);
mortality.insert(mortality.end(), mortalityNCD.begin(), mortalityNCD.end());
```

- creating histograms for distribution of healthcare access and mortality rate

```
figure();
hist(healthcare, 20);
title("Distribution of Healthcare Access");
xlabel("Healthcare Access (%)");
ylabel("Frequency");
show();

figure();
hist(mortality, 20);
title("Distribution of Mortality Rate");
xlabel("Mortality Rate (%)");
ylabel("Frequency");
show();
```

- creating and populating a vector of strings for country names
- calculating average mortality rates from values of map
- creating a 2D vector to store average mortality rates
    - index 0: average mortality rates for infectious
    - index 1: average mortality rates for non-communicable diseases
- creating and populating a bector of doubles for x-axis indices

```
vector<string> countries;
vector<vector<double>> graphData(2, vector<double>());
vector<double> x;

for (const auto& data : countryMortalityInfectious) {
    string country = data.first;
    countries.push_back(country);

    graphData[0].push_back(countryMortalityInfectious[country].first /
countryMortalityInfectious[country].second);
    graphData[1].push_back(countryMortalityNCD[country].first /
countryMortalityNCD[country].second);
}

for (int i = 1; i <= countries.size();i++)
    x.push_back(i);
```

- creating grouped bar chart for average mortality rate by country including both disease types

```
   figure();
   bar(graphData);
   xticks(x);
   xticklabels(countries);
   xtickangle(60);
   xlabel("Countries");
   ylabel("Average Mortality Rate");
   title("Average Mortality Rates by Country");
   matplot::legend({"Infectious", "Non-Communicable Diseases"});    //NOT WORKING- same issue
as exam
   show();
```

- calling linear regression function using vectors of healthcare access percentages and mortality rate percentages
- storing slope, intercept and rsquared from function
- generating 25 x-values for linear regression line
- calculating y-values for linear regression line by creating equation from slope and intercept
- creating scatter plot with linear regression line

*done for both disease types: infectious and non-communicable disease

```
   auto [slopeInterceptInfectious, rSquaredInfectious] =
linearRegression(healthcareInfectious, mortalityInfectious);
   double slopeInfectious = slopeInterceptInfectious.first;
   double interceptInfectious = slopeInterceptInfectious.second;
   vector<double> trendInfectiousX = linspace(*min_element(healthcareInfectious.begin(),
healthcareInfectious.end()), *max_element(healthcareInfectious.begin(),
healthcareInfectious.end()), 25);
   vector<double> trendInfectiousY(trendInfectiousX.size());
   transform(trendInfectiousX.begin(), trendInfectiousX.end(), trendInfectiousY.begin(),
[slopeInfectious, interceptInfectious](double x) {return slopeInfectious * x +
interceptInfectious;});
   string str =  "y = " + to_string(slopeInfectious) + "x + " + to_string(interceptInfectious)
+ "  (R-squared: " + to_string(rSquaredInfectious) + ")";

   figure();
   hold(on);
   scatter(healthcareInfectious, mortalityInfectious, 2.0);
   plot(trendInfectiousX, trendInfectiousY);
   title("Healthcare Access (%) vs Mortality Rate (%) for Infectious Diseases");
   xlabel("Healthcare Access (%)");
   ylabel("Mortality Rate (%)");
   matplot::legend({"Data from Kaggle", str});
   show();

   auto [slopeInterceptNCD, rSquaredNCD] = linearRegression(healthcareNCD, mortalityNCD);
   double slopeNCD = slopeInterceptNCD.first;
   double interceptNCD = slopeInterceptNCD.second;
   vector<double> trendNCDX = linspace(*min_element(healthcareNCD.begin(),
healthcareNCD.end()), *max_element(healthcareNCD.begin(), healthcareNCD.end()), 25);
   vector<double> trendNCDY(trendNCDX.size());
   transform(trendNCDX.begin(), trendNCDX.end(), trendNCDY.begin(), [slopeNCD,
interceptNCD](double x) {return slopeNCD * x + interceptNCD;});
   str =  "y = " + to_string(slopeNCD) + "x + " + to_string(interceptNCD) + "  (R-squared: " +
to_string(rSquaredNCD) + ")";

   figure();
   hold(on);
   scatter(healthcareNCD, mortalityNCD, 2.0);
   plot(trendNCDX, trendNCDY);
   title("Healthcare Access (%) vs Mortality Rate (%) for Non-Communicable Diseases");
   xlabel("Healthcare Access (%)");
   ylabel("Mortality Rate (%)");
   matplot::legend({"Data from Kaggle", str});
   show();

   return 0;
}
```