Covid Data analysis project

1. Introduction with title
2. Literature review ( you can include objective also in this point)
3. Methodology
4. Data Exploration
5. Future works or Suggestions
6. Reference
7. Conclusion

Covid Data Analysis

The COVID-19 pandemic, caused by the SARS-CoV-2 virus, has profoundly impacted global health, economies, and daily life. Analyzing COVID-related data is critical for understanding the spread of the virus, evaluating healthcare capacities, and designing effective public health interventions. Data analysis enables us to identify patterns in infection rates, assess the efficacy of vaccines, and study the correlation between various demographic factors and disease outcomes. By leveraging statistical tools and visualization techniques, researchers can extract meaningful insights from vast datasets, which often include information on cases, recoveries, deaths, vaccination rates, and testing.

COVID data analysis involves exploring and interpreting data from diverse sources, such as government health agencies, the World Health Organization (WHO), and local healthcare institutions. The goal is to present actionable insights that help policymakers and healthcare providers respond effectively to evolving situations. For instance, time-series analysis can track the progression of the pandemic, while histograms and scatterplots reveal distribution patterns and relationships between variables like age, comorbidities, and mortality rates. Additionally, predictive modeling plays a significant role in forecasting future trends, enabling early preparation for potential outbreaks.

As we navigate a post-pandemic era, continuous monitoring and analysis of COVID data remain crucial. Such efforts contribute not only to managing current health challenges but also to improving our preparedness for future pandemics, fostering resilience in healthcare systems worldwide.

Literature review

**Approach for Literature Review:**

1. **Health and COVID-19 Mortality**:
   * Discuss the relationship between population demographics (age, health conditions) and COVID-19 mortality.
   * Review studies on the impact of health quality indicators like "Good Health" or "Bad Health" on pandemic outcomes.
2. **Socioeconomic Deprivation**:
   * Explore literature linking household deprivation (e.g., multiple deprivation dimensions) to health outcomes and mortality rates.
   * Highlight findings on socioeconomic disparities and their influence on disease spread or death rates.
3. **Demographics and Population Density**:
   * Investigate research about the effects of population density on health outcomes, including its role in the spread of infectious diseases.
   * Consider the impact of age distribution on healthcare needs and disease susceptibility.
4. **Economic Activity**:
   * Review studies on the economic impact of pandemics, including differences between economically active and inactive populations.
   * Discuss the role of students and working-age adults in disease transmission.

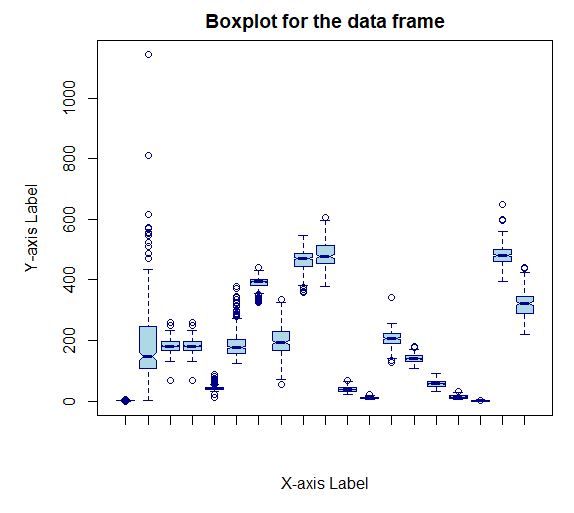
**SQL How to import data:**

|  |
| --- |
| SELECT  (CAST(Aged\_4yearsandunder AS REAL) / CAST(popu\_den AS REAL)) \* 1000 AS Agedu4 FROM joinedTABLE;  create table Datafinal as  select  mnemonic, (Economically\_active/pop\_den)\*1000 as Eco\_Active, (Economically\_active\_student/pop\_den)\*1000 as Eco\_Active\_student, (Economically\_inactive/pop\_den)\*1000 as Eco\_inactive from Neeeded  create table propotioned\_value3 as SELECT District, Area\_Code, popu\_den, (Aged\_4yearsandunder / popu\_den) \* 1000 AS Agedu4, (Aged\_5to9years / popu\_den) \* 1000 as Agedu9, (Aged\_10to15years / popu\_den) \* 1000 AS Agedu15, (Aged\_16to19years / popu\_den) \* 1000 AS Agedu19, (Aged\_20to24years / popu\_den) \* 1000 AS Agedu24, (Aged\_25to34years / popu\_den) \* 1000 AS Agedu34, (Aged\_35to49years / popu\_den )\* 1000 AS Agedu49, (Aged\_50to64years / popu\_den )\* 1000 AS Agedu64, (Aged\_65to74years / popu\_den) \* 1000 AS Agedu74, (Aged\_75to84years / popu\_den) \* 1000 AS Agedu84, (Aged\_85yearsandover/ popu\_den) \* 1000 AS Agedu85, (Household\_anydimension/popu\_den) \* 1000 as NoDimension, (Household\_onedimension /popu\_den) \* 1000 as OneDimension, (Household\_twodimensions /popu\_den) \* 1000 as TwoDimension, (Household\_threedimensions /popu\_den) \* 1000 as ThreeDimension, (Household\_fourdimensions /popu\_den) \* 1000 as FourthDimension, (Detached / popu\_den)\* 1000 as Detached, (Semi\_detached / popu\_den) \* 1000 as Semi\_detacheds, ---(Economically\_ACT\_In\_employment / popu\_den) \* 1000 as Eco\_Active, ---(Economically\_active\_Unemployed / popu\_den) \* 1000 as Eco\_inactive, (Good\_Health / popu\_den)\* 1000 as Good\_Health, (Very\_good\_health / popu\_den )\* 1000 as Verygoodhealth, (Bad\_health /popu\_den) \* 1000 as Badhealth, (Very\_bad\_health/ popu\_den) \* 1000 as  veryBadhealth from joinedTABLE |

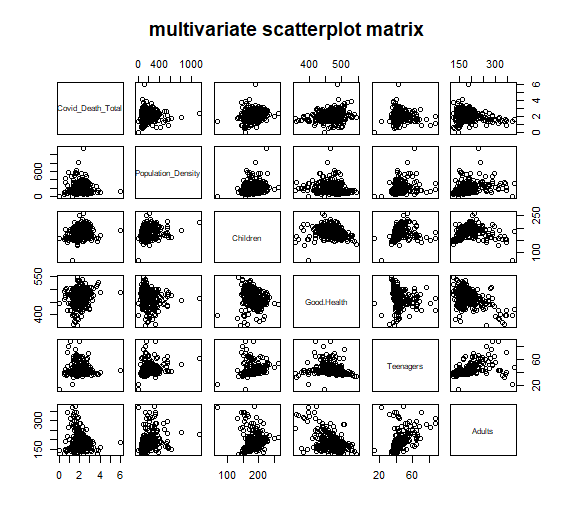
Data Exploration

The uploaded dataset contains 21 columns and 296 entries. The columns include various demographic and socioeconomic indicators such as:

|  |
| --- |
| boxplot(London.dis$Covid\_Death\_Total, London.dis$Population\_Density, London.dis$Children, London.dis$Children,  London.dis$Teenagers  , London.dis$Adults  , London.dis$Middle.Age.Adults,London.dis$Senior  ,London.dis$Good.Health,  London.dis$Very.good.health,London.dis$Bad.health,London.dis$Very.bad.health    ,London.dis$No.Dimension,  London.dis$one.dimension,London.dis$two.dimensions  ,London.dis$three.dimensions,London.dis$four.dimensions,  London.dis$Economically.active..excluding.full.time.students.,London.dis$Economically.inactive,  main = "Boxplot for the data frame",  xlab = "X-axis Label",  ylab = "Y-axis Label",  col = "lightblue",  border = "darkblue",  notch = TRUE) |

****

1. **Geographical Identifiers**:
   * LA\_name: Local authority name.
   * LA\_code: Local authority code.
2. **Health Data**:
   * Covid\_Death\_Total: Total deaths attributed to COVID-19.
   * Health indicators: Good Health, Very good health, Bad health, Very bad health.
3. **Population Demographics**:
   * Age categories: Children, Teenagers, Adults, Middle Age Adults, Senior.
   * Population Density.
4. **Household Deprivation**:
   * Dimensions of deprivation: e.g., Household is not deprived in any dimension, Household is deprived in four dimensions.
5. **Economic Activity**:
   * Economically active (excluding full-time students).
   * Economically active and a full-time student.
   * Economically inactive.
6. **Data Exploration through :**



Visualizing to find the correlation between the variables with

**A graph with numbers and a number of circles

Description automatically generated with medium confidence**

|  |
| --- |
| **#** correlations among numeric variables in  cor.matrix <- cor(london.dis2, use = "pairwise.complete.obs", method = "spearman")  round(cor.matrix, digits = 2)  cor.df <- as.data.frame(cor.matrix)  view(cor.df)  round(cor.df, 2) |

Results :

|  |
| --- |
| London.dis.Covid\_Death\_Total 0.20  London.dis.Population\_Density -0.13  London.dis.Children -0.63  London.dis.Children.1 -0.63  London.dis.Teenagers -0.17  London.dis.Adults -0.40  London.dis.Middle.Age.Adults -0.48  London.dis.Senior 0.66  London.dis.Good.Health 0.66  London.dis.Very.good.health -0.71  London.dis.Bad.health 0.66  London.dis.Very.bad.health 0.59  London.dis.No.Dimension -0.09  London.dis.one.dimension 0.69  London.dis.two.dimensions 0.48  London.dis.three.dimensions 0.26  London.dis.four.dimensions 0.00  London.dis.Economically.active..excluding.full.time.students. -0.84  London.dis.Economically.inactive 1.00 |

**Know from the above results we will select the variables which have positive and negative correlation**

** Positive Correlation:**

* **London.dis.Senior**
* **London.dis.Good.Health**
* **London.dis.Bad.health**
* **London.dis.one.dimension**
* **London.dis.Economically.inactive**

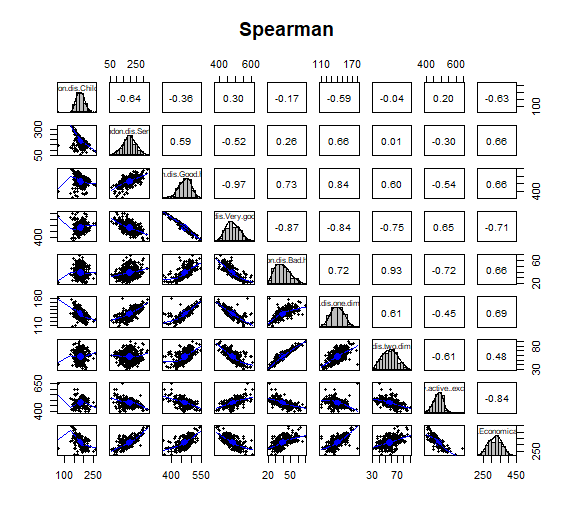
** Negative Correlation:**

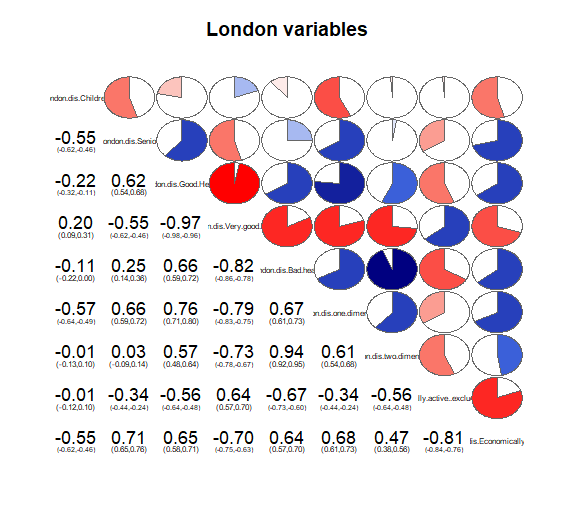
* **London.dis.Children**
* **London.dis.Very.good.health**
* **London.dis.Economically.active..excluding.full.time.students**

**Taking all the needed variables into a new dataframe**

|  |
| --- |
| London.dis3 <- data.frame( London.dis$Children,  London.dis$Senior  ,London.dis$Good.Health,  London.dis$Very.good.health,  London.dis$Bad.health,  London.dis$one.dimension,  London.dis$two.dimensions ,  London.dis$Economically.active..excluding.full.time.students.,  London.dis$Economically.inactive) |

Visualizing the elected variables correlation





|  |
| --- |
| c**or.matrix <- cor(London.dis3, use = "pairwise.complete.obs", method = "spearman")**  **round(cor.matrix, digits = 2)**  **cor.df <- as.data.frame(cor.matrix)**  **View(cor.df)**  **# rename rows and columns**  **#dimnames(cor.df) <- list(c("Life\_Male","Dom\_Build", "Smoking", "Obese", "Episodes", "Benefits", "Crime"),**  **# c("Life\_Male","Dom\_Build", "Smoking", "Obese", "Episodes", "Benefits", "Crime"))**  **round(cor.df, 2)** |

Results

|  |
| --- |
| **London.dis.Children -0.63**  **London.dis.Senior 0.66**  **London.dis.Good.Health 0.66**  **London.dis.Very.good.health -0.71**  **London.dis.Bad.health 0.66**  **London.dis.one.dimension 0.69**  **London.dis.two.dimensions 0.48**  **London.dis.Economically.active..excluding.full.time.students. -0.84**  **London.dis.Economically.inactive 1.00** |

from the above results we are gonna eliminate the variable high and weak correlation and keep the other variables

|  |
| --- |
| **London.dis5 <- data.frame( London.dis$Children,**  **London.dis$Senior**  **,London.dis$Good.Health,**  **London.dis$Very.good.health,**  **London.dis$Bad.health,**  **London.dis$one.dimension,**  **London.dis$Economically.active..excluding.full.time.students.)** |

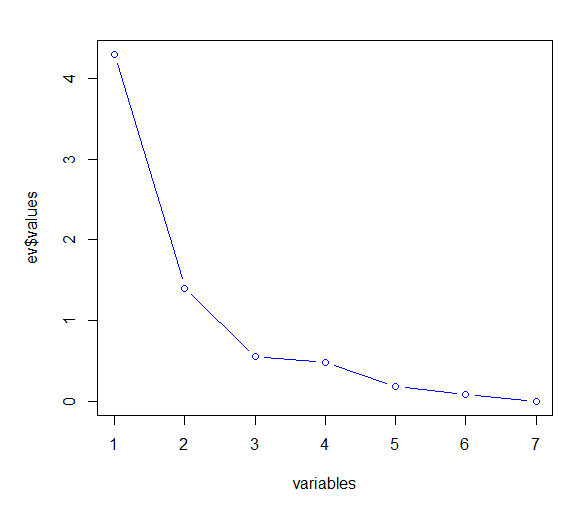
with the above data frame we were able to achieve the correlation matirx as the best combination because of the variable the KMO factor value was way low below the threshold

|  |
| --- |
| library(psych)  KMO(cor(London.dis5))  # Determine Number of Factors to Extract  library(nFactors)  # get eigenvalues: eigen() uses a correlation matrix  ev <- eigen(cor(London.dis5))  ev$values  # plot a scree plot of eigenvalues  plot(ev$values, type="b", col="blue", xlab="variables")  # calculate cumulative proportion of eigenvalue and plot  ev.sum<-0  for(i in 1:length(ev$value)){  ev.sum<-ev.sum+ev$value[i]  }  ev.list1<-1:length(ev$value)  for(i in 1:length(ev$value)){  ev.list1[i]=ev$value[i]/ev.sum  }  ev.list2<-1:length(ev$value)  ev.list2[1]<-ev.list1[1]  for(i in 2:length(ev$value)){  ev.list2[i]=ev.list2[i-1]+ev.list1[i]  }  plot (ev.list2, type="b", col="red", xlab="number of components", ylab ="cumulative proportion") |

above are the codes for performing the kmo test and also dimension reduction method

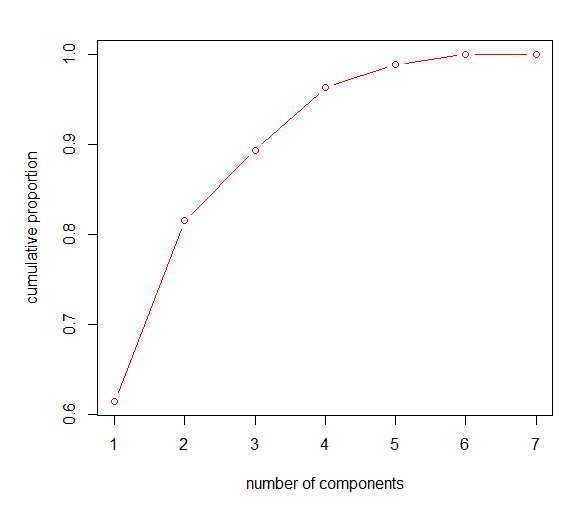
|  |
| --- |
| Kaiser-Meyer-Olkin factor adequacy  Call: KMO(r = cor(London.dis5))  Overall MSA = 0.63  MSA for each item =  London.dis.Children London.dis.Senior  0.67 0.75  London.dis.Good.Health London.dis.Very.good.health  0.57 0.59  London.dis.Bad.health London.dis.one.dimension  0.51 0.79 London.dis.Economically.active..excluding.full.time.students. 0.7 |

with the above results we have proceeded to get the eigen values with the help of graph



|  |
| --- |
| **library(GPArotation)**  **# principal() uses a data frame or matrix of correlations**  **fit <- principal(London.dis2, nfactors=4, rotate="varimax")**  **fit** |

with the help of gp rotation method we can elect the needed factors for further analysis



|  |
| --- |
| # get eigenvalues  ev <- eigen(cor(London.dis5))  ev$values  # plot a scree plot of eigenvalues  plot(ev$values, type="b", col="blue", xlab="variables")  # calculate cumulative proportion of eigenvalue and plot  ev.sum<-0  for(i in 1:length(ev$value)){  ev.sum<-ev.sum+ev$value[i]  }  ev.list1<-1:length(ev$value)  for(i in 1:length(ev$value)){  ev.list1[i]=ev$value[i]/ev.sum  }  ev.list2<-1:length(ev$value)  ev.list2[1]<-ev.list1[1]  for(i in 2:length(ev$value)){  ev.list2[i]=ev.list2[i-1]+ev.list1[i]  }  plot (ev.list2, type="b", col="red", xlab="number of components", ylab ="cumulative proportion")  # Varimax Rotated Principal Components  # retaining 'nFactors' components  fit <- principal(London.dis5, nfactors=4, rotate="varimax")  fit |

the above code is to perform the principal component analysis for the factors that have been decided

|  |
| --- |
| Principal Components Analysis  Call: principal(r = London.dis5, nfactors = 4, rotate = "varimax")  Standardized loadings (pattern matrix) based upon correlation matrix  RC1 RC2 RC4 RC3 h2 u2 com  London.dis.Children -0.09 -0.96 -0.04 -0.21 0.97 0.034 1.1  London.dis.Senior 0.27 0.41 -0.15 0.83 0.95 0.049 1.8  London.dis.Good.Health 0.82 0.02 -0.22 0.48 0.96 0.043 1.8  London.dis.Very.good.health -0.88 -0.05 0.33 -0.31 0.99 0.014 1.6  London.dis.Bad.health 0.80 0.12 -0.52 -0.17 0.95 0.048 1.9  London.dis.one.dimension 0.78 0.51 -0.07 0.26 0.94 0.060 2.0  London.dis.Economically.active..excluding.full.time.students. -0.31 0.05 0.92 -0.19 0.99 0.012 1.3  RC1 RC2 RC4 RC3  SS loadings 2.88 1.36 1.31 1.19  Proportion Var 0.41 0.19 0.19 0.17  Cumulative Var 0.41 0.61 0.79 0.96  Proportion Explained 0.43 0.20 0.19 0.18  Cumulative Proportion 0.43 0.63 0.82 1.00  Mean item complexity = 1.6  Test of the hypothesis that 4 components are sufficient.  The root mean square of the residuals (RMSR) is 0.03  with the empirical chi square 8 with prob < NA  Fit based upon off diagonal values = **1** |

above the results from the pca test

Complexity

* Item complexity reflects the degree to which a variable is associated with multiple components:
  + A low complexity (close to 1) means the variable is predominantly associated with one component.
  + The mean complexity here is 1.6, indicating moderate overlap among components.

Component Statistics

* SS Loadings: Sum of squared loadings for each component, representing the total variance explained by that component:
  + RC1 explains the most variance (2.88), followed by RC2 (1.36), RC4 (1.31), and RC3 (1.19).
* Proportion of Variance: Percentage of total variance explained by each component:
  + RC1 explains 41%, RC2 19%, RC4 19%, and RC3 17%.
* Cumulative Variance: Accumulated variance explained by the components:
  + After all four components, 96% of the variance is explained.

Goodness-of-Fit

* RMSR (Root Mean Square Residual): The average residual between the observed and reproduced correlations. A low RMSR (0.03) indicates a good fit.
* Empirical Chi-Square: Measures the adequacy of the components in reproducing the correlation matrix. Here, the p-value is not provided, but the results suggest a satisfactory fit.

Interpretation of Components

Based on the loadings:

1. RC1: Strongly associated with health-related variables like London.dis.Good.Health, London.dis.Very.good.health, and London.dis.Bad.health.
2. RC2: Predominantly linked to demographics, particularly London.dis.Children.
3. RC4: Related to economic activity, particularly London.dis.Economically.active..excluding.full.time.students..
4. RC3: Tied to senior demographics, with high loadings for London.dis.Senior.

Conclusion

This PCA identifies four primary dimensions that summarize the original dataset:

1. Health-related factors (RC1).
2. Child demographics (RC2).
3. Economic activity (RC4).
4. Senior demographics (RC3).

The components collectively explain 96% of the variance in the dataset, indicating that the analysis successfully captures the key patterns in the data.

|  |
| --- |
| # create four variables to represent the rorated components  fit$scores  fit.data <- data.frame(fit$scores)  # check new variables are uncorrelated  cor.matrix2 <-cor(fit.data, method = "spearman")  #cor.matrix2 <-cor(fit.data, method = "pearson") fro better correlation anaylsis  cor.df2 <- as.data.frame(cor.matrix2)  round(cor.df2, 2) |

with the above result we can understand that

** Diagonal Elements:**

* **The diagonal values are all 1.00 because a component is always perfectly correlated with itself.**

** Off-Diagonal Elements:**

* **These values represent the correlations between the components:**
  + **RC1 and RC2: r=0.06r = 0.06r=0.06 (low correlation, nearly independent).**
  + **RC2 and RC4: r=−0.23r = -0.23r=−0.23 (weak negative correlation).**
  + **RC3 and RC2: r=0.28r = 0.28r=0.28 (moderate positive correlation).**
  + **Other pairs also have low correlations (near zero), indicating weak or no linear relationships.**

|  |
| --- |
| * # Multiple linear Regression * # model with all variables * model2 <- lm(Covid\_Death\_Total ~ Children + Senior + Good.Health + Very.good.health +Bad.health + one.dimension * + Economically.active..excluding.full.time.students. ,data = London.dis)  * summary(model2) * # calculate variance inflation factor * library(car) * vif(model2) * sqrt(vif(model2)) > 2 # if > 2 vif too high * #here we feel that is model is not good bcoz there are lot of true |

from the above we are checking first model what is the best model on the basis of the results

|  |
| --- |
| **>** vif(model2)  Children Senior Good.Health  2.318204 3.259502 3379.490965  Very.good.health Bad.health one.dimension  5832.228280 583.833221 7.416285  Economically.active..excluding.full.time.students.  2.832609  > sqrt(vif(model2)) > 2 # if > 2 vif too high  Children Senior Good.Health  FALSE FALSE TRUE  Very.good.health Bad.health one.dimension  TRUE TRUE TRUE  Economically.active..excluding.full.time.students.  FALSE |

there are many variables with the values as false which indicated this is not the perfect model

4 variables are slected on the basis of dominance in section 9 factor analysis

|  |
| --- |
| **model3 <- lm(Covid\_Death\_Total ~ Children + Senior + Good.Health + Economically.active..excluding.full.time.students.,,data = London.dis)**  **summary(model3)**  **sqrt(vif(model3)) > 2** |

results :

|  |
| --- |
| **> summary(model3)**  **Call:**  **lm(formula = Covid\_Death\_Total ~ Children + Senior + Good.Health +**  **Economically.active..excluding.full.time.students., data = London.dis)**  **Residuals:**  **Min 1Q Median 3Q Max**  **-1.6167 -0.3726 -0.0040 0.3463 3.8506**  **Coefficients:**  **Estimate Std. Error t value Pr(>|t|)**  **(Intercept) 0.8261829 1.2686134 0.651 0.515400**  **Children 0.0078464 0.0020701 3.790 0.000183 \*\*\***  **Senior 0.0003978 0.0010883 0.366 0.714957**  **Good.Health 0.0033812 0.0016262 2.079 0.038472 \***  **Economically.active..excluding.full.time.students. -0.0038124 0.0013271 -2.873 0.004370 \*\***  **---**  **Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1**  **Residual standard error: 0.6144 on 291 degrees of freedom**  **Multiple R-squared: 0.1485, Adjusted R-squared: 0.1368**  **F-statistic: 12.69 on 4 and 291 DF, p-value: 1.568e-09**  **> sqrt(vif(model3)) > 2**  **Children Senior Good.Health**  **FALSE FALSE FALSE**  **Economically.active..excluding.full.time.students.**  **FALSE** |

from the above results we can this is a fit model

|  |
| --- |
| model4 <- lm(Covid\_Death\_Total ~ Children + Senior +Bad.health+ Economically.active..excluding.full.time.students.,,data = London.dis)  summary(model3)  sqrt(vif(model3)) > 2 |

the above medal is replace good health with bad health

results

|  |
| --- |
| **> summary(model3)**  **Call:**  **lm(formula = Covid\_Death\_Total ~ Children + Senior + Good.Health +**  **Economically.active..excluding.full.time.students., data = London.dis)**  **Residuals:**  **Min 1Q Median 3Q Max**  **-1.6167 -0.3726 -0.0040 0.3463 3.8506**  **Coefficients:**  **Estimate Std. Error t value Pr(>|t|)**  **(Intercept) 0.8261829 1.2686134 0.651 0.515400**  **Children 0.0078464 0.0020701 3.790 0.000183 \*\*\***  **Senior 0.0003978 0.0010883 0.366 0.714957**  **Good.Health 0.0033812 0.0016262 2.079 0.038472 \***  **Economically.active..excluding.full.time.students. -0.0038124 0.0013271 -2.873 0.004370 \*\***  **---**  **Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1**  **Residual standard error: 0.6144 on 291 degrees of freedom**  **Multiple R-squared: 0.1485, Adjusted R-squared: 0.1368**  **F-statistic: 12.69 on 4 and 291 DF, p-value: 1.568e-09**  **> sqrt(vif(model3)) > 2**  **Children Senior Good.Health**  **FALSE FALSE FALSE**  **Economically.active..excluding.full.time.students.**  **FALSE** |

from the above results we can this is a fit model

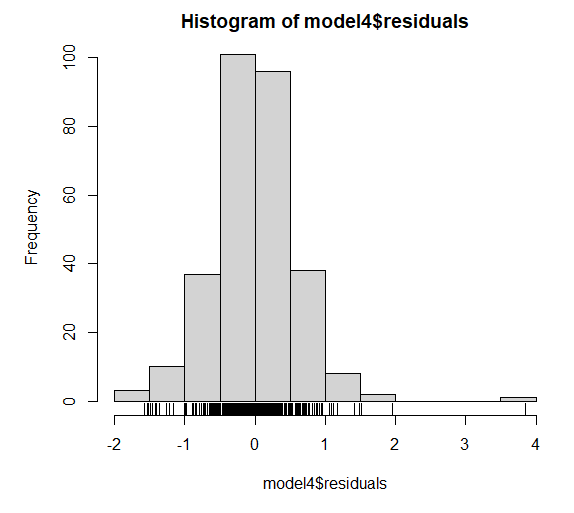
forward stepwise selection

|  |
| --- |
| ibrary(RcmdrMisc)  library(relaimpo)  # forward stepwise selection  model4 <- stepwise(model3, direction = "forward")  summary(model4)  hist(model4$death)  rug(model4$residuals)  plot(model4$residuals ~ model4$fitted.values, xlab = "fitted values", ylab = "residuals")  ks.test(model4$residuals, "pnorm", mean(model4$residuals), sd(model4$residuals))  sqrt(vif(model4)) > 2  calc.relimp(model4, type = c("lmg"), rela = TRUE) |

to find the best model

|  |
| --- |
| **>** summary(model4)  Call:  lm(formula = Covid\_Death\_Total ~ Economically.active..excluding.full.time.students. +  Children + Good.Health, data = London.dis)  Residuals:  Min 1Q Median 3Q Max  -1.5832 -0.3571 -0.0112 0.3491 3.8444  Coefficients:  Estimate Std. Error t value Pr(>|t|)  (Intercept) 0.863427 1.262638 0.684 0.49463  Economically.active..excluding.full.time.students. -0.003859 0.001319 -2.925 0.00371 \*\*  Children 0.007433 0.001732 4.291 2.42e-05 \*\*\*  Good.Health 0.003677 0.001408 2.611 0.00949 \*\*  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  Residual standard error: 0.6134 on 292 degrees of freedom  Multiple R-squared: 0.1481, Adjusted R-squared: 0.1394  F-statistic: 16.92 on 3 and 292 DF, p-value: 3.67e-10 |

Visual representation of the model4



from the code we can understand we have performed many other test methods to analyse

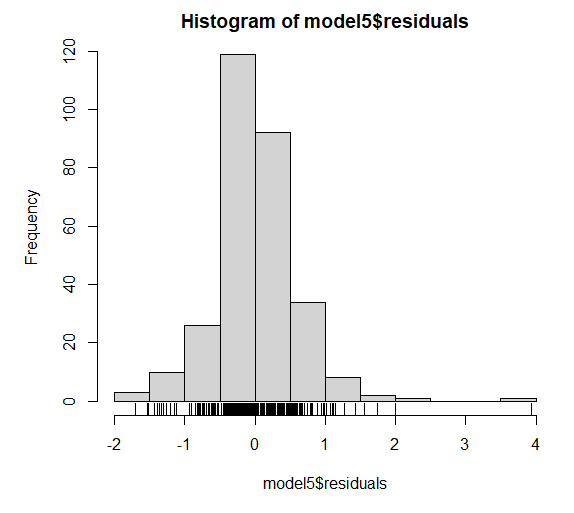
again performing forward stepwise selection

|  |
| --- |
| model5 <- stepwise(model2, direction = "forward")  summary(model5)  hist(model5$residuals)  rug(model5$residuals)  plot(model5$residuals ~ model5$fitted.values, xlab = "fitted values", ylab = "residuals")  ks.test(model5$residuals, "pnorm", mean(model5$residuals), sd(model5$residuals))  sqrt(vif(model5)) > 2  calc.relimp(model5, type = c("lmg"), rela = TRUE) |

to find the best model

|  |
| --- |
| **> summary(model5)**  **Call:**  **lm(formula = Covid\_Death\_Total ~ Bad.health + Children, data = London.dis)**  **Residuals:**  **Min 1Q Median 3Q Max**  **-1.7040 -0.3161 -0.0393 0.3211 3.9424**  **Coefficients:**  **Estimate Std. Error t value Pr(>|t|)**  **(Intercept) -0.356996 0.346032 -1.032 0.303**  **Bad.health 0.026661 0.003643 7.319 2.41e-12 \*\*\***  **Children 0.007636 0.001634 4.674 4.51e-06 \*\*\***  **---**  **Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1**  **Residual standard error: 0.5972 on 293 degrees of freedom**  **Multiple R-squared: 0.1899, Adjusted R-squared: 0.1844**  **F-statistic: 34.34 on 2 and 293 DF, p-value: 3.981e-14** |

Visual representation of the model5



performing annova test

|  |
| --- |
| anova(model2, model5, test = "F") |

results

|  |
| --- |
| **Analysis of Variance Table**  **Model 1: Covid\_Death\_Total ~ Children + Senior + Good.Health + Very.good.health +**  **Bad.health + one.dimension + Economically.active..excluding.full.time.students.**  **Model 2: Covid\_Death\_Total ~ Bad.health + Children**  **Res.Df RSS Df Sum of Sq F Pr(>F)**  **1 288 101.75**  **2 293 104.49 -5 -2.7461 1.5546 0.173** |

Interpretation

**r**esiduals Sum of Squares (RSS):

* Measures the unexplained variance in the dependent variable (Covid\_Death\_Total) after fitting the model.
* Model 1 RSS = 101.75: Slightly lower than Model 2, indicating better fit.
* Model 2 RSS = 104.49: Higher, indicating slightly worse fit.

Data Exploration 1:

|  |
| --- |
| library(ggplot2)  # example: load a dataset (you can replace this with your own data)  # data <- read.csv("your\_dataset.csv")  # function to plot histograms for all numeric variables in a data frame  plot\_histograms <- function(london.dis2) {  numeric\_columns <- sapply(london.dis2, is.numeric) # identify numeric columns  numeric\_data <- london.dis2[, numeric\_columns] # subset only numeric columns    for (col\_name in names(numeric\_data)) {  # create a histogram using ggplot2  p <- ggplot(london.dis2, aes\_string(x = col\_name)) +  geom\_histogram(binwidth = 05, fill = "blue", color = "black", alpha = 0.7) +  labs(title = paste("histogram of", col\_name), x = col\_name, y = "frequency") +  theme\_minimal()    print(p) # display the plot  }  }  # use the function on your dataset  # plot\_histograms(data)  plot\_histograms(london.dis2) |

The above code helps to project all the variables in histogram and understand the plots in a better way to understand the normal distribution

|  |
| --- |
| library(ggplot2)  library(gridExtra) # For arranging multiple plots in a grid  # Function to create histograms for all numeric variables and arrange them in one output  plot\_histograms\_grid <- function(London.dis2) {  numeric\_columns <- sapply(London.dis2, is.numeric) # Identify numeric columns  numeric\_data <- London.dis2[, numeric\_columns] # Subset only numeric columns    plot\_list <- list() # Empty list to store plots    for (col\_name in names(numeric\_data)) {  # Create a histogram using ggplot2  p <- ggplot(London.dis2, aes\_string(x = col\_name)) +  geom\_histogram(binwidth = 0.5, fill = "blue", color = "black", alpha = 0.7) +  labs(title = paste("Histogram of", col\_name), x = col\_name, y = "Frequency") +  theme\_minimal()    plot\_list[[col\_name]] <- p # Add each plot to the list  }    # Arrange all plots in a grid  do.call(grid.arrange, c(plot\_list, ncol = 2)) # Adjust ncol for the number of columns  }  # Use the function on your dataset  # plot\_histograms\_grid(data)  plot\_histograms\_grid(London.dis2) |

result : from the above code we can visualize all the histogram at once

**References :**

**R code :** [https://uelac-my.sharepoint.com /:u:/g/personal/u2761302\_uel\_ac\_uk/EToDb2zgTh1AijJUQuNWlSUBS0eLc9-Ax3kFPWglWAn48w?e=X6Rpse](https://uelac-my.sharepoint.com/:u:/g/personal/u2761302_uel_ac_uk/EToDb2zgTh1AijJUQuNWlSUBS0eLc9-Ax3kFPWglWAn48w?e=X6Rpse)

**Sql Code :** [**SQL databases**](https://uelac-my.sharepoint.com/:f:/g/personal/u2761302_uel_ac_uk/EvWMw_lVn51IiNADrwJi3W0B7AV39JxZpiDw95vDwi_k-A?e=cbBxa6)

**Excel Sheet :** [**https://uelac-my.sharepoint.com/:x:/g/personal/u2761302\_uel\_ac\_uk/EU7515eQ3q9EkG4uSmfeaDgBvzzT541ys3s1kfomn4C0Eg?e=tEeFph**](https://uelac-my.sharepoint.com/:x:/g/personal/u2761302_uel_ac_uk/EU7515eQ3q9EkG4uSmfeaDgBvzzT541ys3s1kfomn4C0Eg?e=tEeFph)