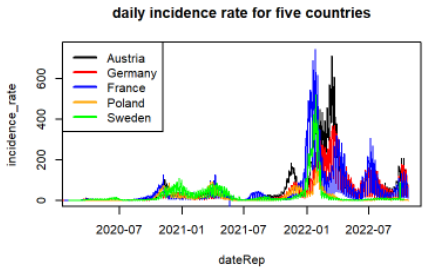
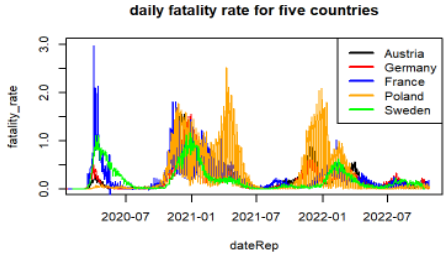
---
title: "COVID 19 Analysis Rerun in Winter 2024"

Analyzing data on new daily covid-19 cases and deaths in European Union (EU) and European Economic Area (EEA) countries. A data file may be downloaded [here](https://www.ecdc.europa.eu/en/publications-data/data-daily-new-cases-covid-19-eueea-country).
I will: (1) perform an Exploratory Data Analysis (EDA), (2) perform some hypothesis testing, (3) perform some correlation testing, and (4) fit and describe a linear regression model.
```
library(ggplot2)
library(gridExtra)
library(lubridate)
library(tidyverse)
library(dplyr)
library(Hmisc)
library(moments)
# The read.csv() below reads a local copy of the data file
data <- read.csv("mydata.csv", na.strings = "", fileEncoding = "UTF-8-BOM")
# The zero-th step in any analysis is to 'sanity check' our data.
glimpse(data)
# Some data processing
data <- data %>%
 select(-c("continentExp")) %>%
 mutate(dateRep = dmy(dateRep),
      countriesAndTerritories = as.factor(countriesAndTerritories),
      geoId = as.factor(geoId),
      countryterritoryCode = as.factor(countryterritoryCode))
```
A data dictionary for the dataset is available [here](https://www.ecdc.europa.eu/sites/default/files/documents/Description-and-disclaimer_daily_reporting.pdf).
Definitions:
* "Incidence rate" is equal to new daily cases per 100K individuals. Country population estimates can be found in 'popData2020.' You will calculate a daily incidence rate in item (1), for each country, that we will explore further in items (2) and (3).
* "Fatality rate" is equal to new daily deaths per 100K individuals. Country population estimates can be found in 'popData2020.' You will calculate a daily fatality rate in item (1), for each country, that we will explore further in items (2) and (3).
---
1. Descriptive Statistics
 Perform an Exploratory Data Analysis (EDA).
```
##Creation of vectors: incidence_rate, the daily new cases per 100K individuals, per country, and fatality_rate, the daily new deaths per 100K individuals, per country.
data$incidence_rate <- data$cases/data$popData2020*100000
data$fatality_rate <- data$deaths/data$popData2020*100000
str(data)
summary(data)
```

```
##Subset data for five countries.
countries <- c("Austria","Germany", "France", "Poland", "Sweden")
dataAT <- data[data$geoId == 'AT',] .....
##Plot incidence rates over time for five countries individually and in a single plot.
par(mfrow = c(3,2))
plot(incidence_rate ~ dateRep, dataAT, type = 'l', col = 'black', main = "daily incidence rate for
Austria") .....
plot(incidence_rate ~ dateRep, dataAT, type = 'l', ylim = c(0, 750), col = 'black', main = "daily incidence rate
for five countries")
lines(incidence_rate ~ dateRep, dataDE, type = 'l', col =
'red') .....
legend( x="topleft",
      legend=countries,
      col=c("black","red", "blue", "orange", "green"), lwd=2, cex = 1)
```




```
##Plot fatality rates over time for five countries individually and in a
single plot.
par(mfrow = c(3,2))
plot(fatality_rate ~ dateRep, dataAT, type = 'l', col = 'black', main = "daily fatality rate for
Austria") .....
plot(fatality_rate ~ dateRep, dataAT, type = 'l', ylim = c(0, 3.0), col
= 'black', main = "daily fatality rate for five countries")
lines(fatality_rate ~ dateRep, dataDE, type = 'l', col = 'red') ...
legend( x="topright",
      legend=countries,
      col=c("black","red", "blue", "orange", "green"), lwd=2, cex = 1)
```

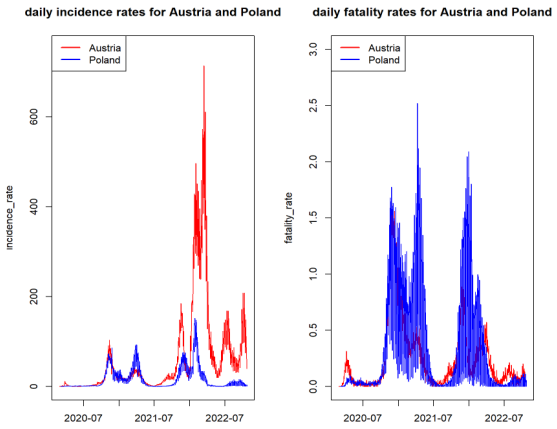



```
##Calculate proportions: prop.Cases and prop.Deaths.
populations <- c(mean(data$popData2020[data$geoId == 'AT']), mean(data$popData2020[data$geoId ==
'DE']), ...)
totalCases <- c(sum(dataAT$cases), sum(dataDE$cases), ...)
prop.Cases <- totalCases/populations
totalDeaths <- c(sum(dataAT$deaths), sum(dataDE$deaths), ...)
prop.Deaths <- totalDeaths/populations

##Calculate case fatality rate = prop.Deaths/prop.Cases = totalDeaths/totalCases.
case_fatality_rate <- totalDeaths/totalCases
##Maximum daily incidence rate and the associated date.
max_incidence_rate <- c(max(dataAT$incidence_rate), max(dataDE$incidence_rate), ...)
date_max_incidence_rate <- c(dataAT$dateRep[which.max(dataAT$incidence_rate)], ...)
##Maximum daily fatality rate and the associated date.
max_fatality_rate <- c(max(dataAT$fatality_rate), max(dataDE$fatality_rate), ...)
date_max_fatality_rate <- c(dataAT$dateRep[which.max(dataAT$fatality_rate)], ...)
##A table or visualization exploring some other aspect of the data
(tbl <- cbind.data.frame(countries, populations, totalCases, prop.Cases, totalDeaths, prop.Deaths,
case_fatality_rate, max_incidence_rate, date_max_incidence_rate, max_fatality_rate,
date_max_fatality_rate))
cat("\n## It is noticeable that out of the five countries, Poland has the highest prop.Deaths and
case_fatality_rate, whereas it has the lowest prop.Cases.\n")
```

cat("\n## For the five countries, the maximum incidence rate occurs all in 2022, and the maximum fatality rate occurred almost all in 2020, except that the maximum fatality rate occurs in 2021 for Poland.\n")
```

#### 2. Inferential Statistics
 Select two (2) countries and compare their incidence or fatality rates using hypothesis testing.
```



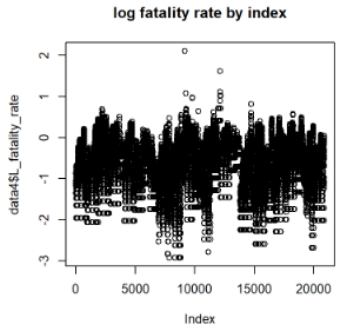
cat("#1. Visualizations comparing the daily incidence or fatality rates of the selected two countries, respectively. \n ")
par(mfrow = c(1,2))
plot(incidence_rate ~ dateRep, dataAT, type = 'l', ylim = c(0, 750), col = 'red', main = "daily incidence rates for Austria and Poland")
lines(incidence_rate ~ dateRep, dataPL, type = 'l', col = 'blue')
legend(x="topleft", legend=c("Austria", "Poland"),
    col=c("red", "blue"), lwd=2, cex = 1)
plot(fatality_rate ~ dateRep, dataAT, type = 'l', ylim = c(0, 3.0), col = 'red', main = "daily fatality rates for Austria and Poland")
lines(fatality_rate ~ dateRep, dataPL, type = 'l', col = 'blue')
legend(x="topleft", legend=c("Austria", "Poland"),
    col=c("red", "blue"), lwd=2, cex = 1)
cat("\n#2. Statement of the null hypotheses: daily incidence rates for Austria and Poland have the same mean.\n")
cat("\n#3. Justification of the statistical test selected: Welch t-test is suitable for comparing two means and it is more reliable than Student's t-test and maintains type I error rates close to normal for two samples with unequal variances and different sizes. Welch's t-test remains robust for skewed distributions and large sample sizes, which is the case for our 'data'.\n")
cat("\n#4. Distributional assumption is normal distribution, with deviations in our data potentially tolerated. Our data deviate from normal distribution in that they display a high-skewness and high-kurtosis distribution with a very large sample size. \n")
#Check about incidence_rate normality.
cat("\n## Check normality of data$incidence_rate by Shapiro-Wilk normality test, histogram, Q-Q plot, and mean & sd. \n")
# Shapiro-Wilk normality test
##with(data, shapiro.test(incidence_rate))
cat("##Shapiro-Wilk normality test for data$incidence_rate cannot be done due to too large a sample size of the 'data'.\n")
layout(matrix(c(1, 2, 3, 4, 5, 6), 3, 2, byrow = TRUE), respect = TRUE)
hist(data$incidence_rate, breaks=5000, xlim = c(-200, 1000))
qqnorm(data$incidence_rate, pch = 1, frame = FALSE, ylim = c(-200, 1000), cex.lab = 1, cex.axis = 1, cex.main = 1, main = "Q-Q plot for incidence_rate")
qqline(data$incidence_rate, col='red', lwd=2)
cat("\n## mean & sd for data$incidence_rate: \n")
(meanIncidence <- mean(data$incidence_rate, na.rm=TRUE))
(sdIncidence <- sd(data$incidence_rate, na.rm=TRUE))
summary(data$incidence_rate)
cat("\n## skewness & kurtosis for data$incidence_rate: \n")

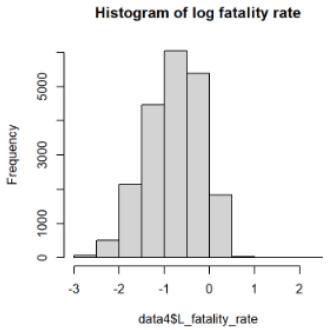
```
skewness(data$incidence_rate, na.rm=TRUE)
kurtosis(data$incidence_rate, na.rm=TRUE)
cat("\n## Check normality of dataAT$incidence_rate by Shapiro-Wilk normality test, histogram, and mean & sd: \n")
# Shapiro-Wilk normality test
with(dataAT, shapiro.test(incidence_rate))
hist(dataAT$incidence_rate, breaks=40, xlim = c(-200, 800))
qqnorm(dataAT$incidence_rate, pch = 1, frame = FALSE, ylim = c(-200, 1000), cex.lab = 1, cex.axis = 1, cex.main = 1, main = "Q-Q plot for incidence_rate in Austria")
qqline(dataAT$incidence_rate, col='red', lwd=2)
cat("\n## mean & sd for dataAT$incidence_rate: \n")
(meanIncidenceAT <- mean(dataAT$incidence_rate, na.rm=TRUE))
(sdIncidenceAT <- sd(dataAT$incidence_rate, na.rm=TRUE))
summary(dataAT$incidence_rate)
cat("\n## skewness & kurtosis for dataAT$incidence_rate: \n")
skewness(dataAT$incidence_rate)
kurtosis(dataAT$incidence_rate)
cat("\n## Check normality of dataPL$incidence_rate by Shapiro-Wilk normality test, histogram, and mean & sd: \n")
# Shapiro-Wilk normality test
with(dataPL, shapiro.test(incidence_rate))
hist(dataPL$incidence_rate, breaks=40, xlim = c(-200, 200))
qqnorm(dataPL$incidence_rate, pch = 1, frame = FALSE, ylim = c(-200, 1000), cex.lab = 1, cex.axis = 1, cex.main = 1, main = "Q-Q plot for incidence_rate in Poland")
qqline(dataPL$incidence_rate, col='red', lwd=2)
cat("\n## mean & sd for dataPL$incidence_rate: \n")
(meanIncidencePL <- mean(dataPL$incidence_rate, na.rm=TRUE))
(sdIncidencePL <- sd(dataPL$incidence_rate, na.rm=TRUE))
summary(dataPL$incidence_rate)
cat("\n## skewness & kurtosis for dataPL$incidence_rate: \n")
skewness(dataPL$incidence_rate)
kurtosis(dataPL$incidence_rate)
cat("\n\n#5. Select alpha = 0.01 for the t-test.\n")
cat("\n\n#6. The test function output; i.e. the R output:\n")
t.test(dataAT$incidence_rate, dataPL$incidence_rate, paired = FALSE, alternative = 'greater', conf.level = 0.99)
cat("#7. The relevant confidence interval is returned by the R test outputs above.\n")
cat("\n#8. The concluding statement on the outcome of the statistical test: Based on the selected alpha = 0.01, with high statistical significance (very low p-value), the null hypotheses is rejected, and thus the daily incidence rate of Austria is greater than that of Poland.\n")
cat("\n#### Alternatively, I also perform the 'Mann-Whitney' test, which is a non-parametric alternative to the t-test for comparing two means. It's particularly recommended in a situation where the data are not normally distributed. It turns out the Wilcox.test (Wilcoxon Rank Sum and Signed Rank Tests) gives the same conclusion in this situation. \n")
wilcox.test(dataAT$incidence_rate, dataPL$incidence_rate, alternative = "greater", conf.int = T, conf.level = 0.99)
```

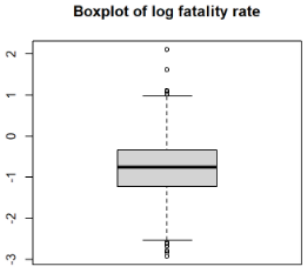
3. Correlation

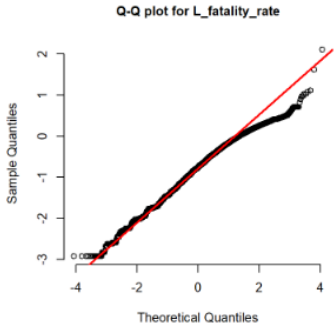
Considering all countries, explore the relationship between incidence rates and fatality rates.

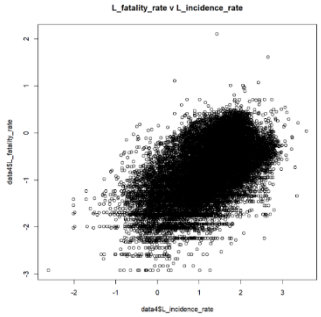
```
cat("#1. Visualizations showing the distributions of daily incidence and fatality rates, regardless of country and date, some plotted as close-ups. \n\n") .....
cat("#2. A short statement identifying the most appropriate correlation coefficient: Since the 'data' distributions for both incidence_rate and fatality_rate largely deviate from normal distribution, kendall's tau rank-based correlation coefficient (non-parametric) is the most appropriate for testing independence between the two variables. Spearman's Rho is another non-parametric rank-based correlation coefficient. Kendall's tau statistic and Spearman's Rho statistic can estimate a rank-based measure of association. These tests do not necessarily require the data to be from a normal distribution. Pearson's correlation (r) statistic measures a linear dependence and requires the data to be from a normal distribution.\n")
cat("\n#3. The calculated rank correlation coefficient or coefficient test output:\n")
cor.test(data$incidence_rate, data$fatality_rate, method = "kendall") ...
cat("## While not appropriate, Pearson product-moment correlation coefficient (for linear correlation) could still be calculated (to be 0.059, very low), showing basically no linear numeric correlation.\n")
cor.test(data$incidence_rate, data$fatality_rate, method = "pearson")
cat("\n## Attempting logarithm with positive incidence rates in the data. \n")
data3 <- data[data$incidence_rate > 0,]
data3$L_incidence_rate <- log10(data3$incidence_rate)
cat("\n## Normality is improved with logarithm for incidence rates as shown in the plots.\n") .....
cat("\n## Trying logarithm with both positive incidence rates and positive fatality rates in the data.\n")
data4 <- data3[data3$fatality_rate > 0,]
data4$L_fatality_rate <- log10(data4$fatality_rate)
cat("\n## Normality and linear correlation improved with logarithm for fatality rates as shown in the plots.\n")
par(mfrow = c(2,2))
plot(data4$L_fatality_rate, main = "log fatality rate by index")
hist(data4$L_fatality_rate, main = "Histogram of log fatality rate")
boxplot(data4$L_fatality_rate, main = "Boxplot of log fatality rate")
qqnorm(data4$L_fatality_rate, pch = 1, frame = FALSE, cex.lab = 1, cex.axis = 1, cex.main = 1, main = "Q-Q plot for L_fatality_rate")
qqline(data4$L_fatality_rate, col='red', lwd=2)
par(mfrow = c(1,1))
plot(data4$L_incidence_rate, data4$L_fatality_rate, main = "L_fatality_rate v L_incidence_rate")
cat("\n## Both linear and rank-based correlation coefficients can be calculated. \n")
cor.test(data4$L_incidence_rate, data4$L_fatality_rate, method = "pearson") .....
```











4. Regression

Here, we will fit a model on data from twenty (20) countries considering total new cases as a function of population, population density and gross domestic product (GDP) per capita. Note that the GDP per capita is given in "purchasing power standard," which considers the costs of goods and services in a country relative to incomes in that country; i.e. we will consider this as appropriately standardized.

Eventually filtering the data frame to include only the remaining 19 countries

Note: Data has changed from 2 years ago. We are selecting the remaining 19 countries to ensure that the analysis includes the most up-to-date data for comparison.

```
# The code below creates a new data frame, 'model_df,' that includes the area,
# GDP per capita, population and population density for the twenty (20)
# countries of interest. All you should need to do is execute this code, as is.
twenty_countries <- c("Austria", ...)
sq_km <- c(83858, ...)
gdp_pps <- c(128, ...)
model_df <- data %>%
 filter(countriesAndTerritories %in% twenty_countries) %>%
 distinct(countriesAndTerritories, .keep_all = TRUE) %>%
 select(countriesAndTerritories, popData2020) %>%
 add_column(sq_km, gdp_pps) %>%
 mutate(pop_dens = popData2020 / sq_km) %>%
 rename(country = countriesAndTerritories, pop = popData2020)
```

``` Next, we add to our 'model_df' data frame -- total new cases for each of the 20 countries.

```
##Calculate the new column - total_cases for each of the twenty countries.
model_df$total_cases <- c(sum(data$cases[data$geoId == 'AT']), sum(data$cases[data$geoId == 'BE']), ...)
```

Now, we fit our model using the data in 'model_df.' We are interested in explaining total cases (response) as a function of population (explanatory), population density (explanatory), and GDP (explanatory).

```
cat("##1. A description - either narrative or using R output - of 'model_df' data frame:\n See the table output returned by 'model_df'.\n\n")
model_df
cat("##2. The *lm()* *summary()* output of the fitted model:\n")
mdl1 <- lm(formula = total_cases ~ pop + pop_dens + gdp_pps, data = model_df)
summary(mdl1)
cat("##3. A short statement on the fit of the model: \n# Which, if any, of our coefficients are statistically significant?\n## Answer: The coefficient for 'pop' is statistically significant (p-value < 0.001. \n\n# What is the R^2 of our model? \n## Answer: Multiple R-squared is 0.9049, and adjusted R-squared is 0.887. \n\n# Should we consider a reduced model; i.e. one with fewer parameters? \n## Answer: Yes. The current model's parameters make sense but it may still work equally well or better after removing one parameter. It may be improved by including additional parameters other than current ones. There is a little improvement in one of the three reduced models shown below using two parameters. In the model (reduced2) using only 'pop' and 'gdp_pps' as the two parameters (the 2nd in the three), the residual standard error (2653000) is slightly less than that of the three-parameter model (2733000), and the adjusted R-squared (0.8935) is slightly greater than that of the three-parameter model (0.887). There's basically no
```
```

```
improvement when examining predictions and residual distribution (in next section regression_d),
though.\n")
reduced1 <- lm(total_cases ~ pop + pop_dens, data = model_df)
summary(reduced1)
reduced2 <- lm(total_cases ~ pop + gdp_pps, data = model_df)
summary(reduced2)
reduced3 <- lm(total_cases ~ pop_dens + gdp_pps, data = model_df)
summary(reduced3)
```
The last thing we will do is use our model to predict the total new cases of two (2) new countries
```
# Defines our 'newdata' data frame.
newdata <- data.frame(country = c("Luxembourg", "Netherlands"),
                      pop = c(626108, 17407585),
                      gdp_pps = c(261, 130),
                      pop_dens = c(626108, 17407585) / c(2586, 41540))
newdata$total_cases <- c(sum(data$cases[data$geoId == 'LU']), sum(data$cases[data$geoId == 'NL']))
# Predicted by our model
newdata$pdtTtlCs <- round(-2.364*10^06 + newdata$pop*3.317*10^-01 +
newdata$pop_dens*2.728*10^02 + newdata$gdp_pps*1.934*10^04)
newdata$residual <- newdata$total_cases - newdata$pdtTtlCs
cat("\n## 'newdata' returns the table that contains the predicted total new cases and the actual total new
cases for both countries (as well as the residuals).\n")
newdata
cat("\n## Check residuals of fitted model: heteroscedaticity.\n")
## Residuals of fitted model .....................................
# There were issues in subsequent steps unlike two years ago.
# So check data lengths and identify the data issue for one country.
# Remove the country "Denmark" based on new data
model_df <- model_df %>%
  filter(country != "Denmark")
##Include residuals from fitted model in the data.
data2 <- model_df
data2$pdtTtlCs <- round(fitted.values(mdl1))
data2$residual <- round(resid(mdl1))
data2
cat("\n## Check reduced model for predictions: very similar to mdl1.\n")
## Predictions by the best of the reduced models.
newdata2 <- newdata
newdata2$pdt2 <- round(-2.316*10^06 + newdata2$pop*3.313*10^-01 + newdata2$pop_dens*0 +
newdata2$gdp_pps*1.941*10^04)
newdata2$res2 <- newdata2$total_cases - newdata2$pdt2
newdata2
cat("\n## Check residuals of reduced model: heteroscedaticity.\n")
## Residuals of reduced model ...................
cat("## Check residuals of the two new countries predicted with mdl1.\n")
## Find how residuals of the two new countries compare with fitted data in mdl1.
data5 <- data2
```

```
data5$sq_km <- NULL
(data5 <- rbind.data.frame(data5, newdata))
## Attempt modeling including two new countries and plots.
mdl2 <- lm(formula = total_cases ~ pop + pop_dens + gdp_pps, data = data5)
summary(mdl2)
par(mfrow = c(1,1))
plot(mdl2$fitted, rstudent(mdl2),
 main="Studentized Residuals (mdl2)",
 xlab="Fitted",ylab="Studentized Resid")
abline(h=0, lty=2)
par(mfrow = c(2,2))
plot(data5$country, mdl2$resid,
 main="Residuals by Country (mdl2)",
 xlab="Country",ylab="Residuals", cex.axis = 0.4)
abline(h=0,lty=2)
hist(mdl2$resid,main="Histogram of Residuals")
boxplot(mdl2$resid, main="Boxplot of Residuals")
qqnorm(mdl2$resid, main="Q-Q plot of
Residuals")
qqline(mdl2$resid)
cat("\n## Check relative positions of two new residuals as compared with countries in mdl1 (and mdl2):
\n## While the models behave normally, the residuals predicted for the two countries have large errors
compared with the actual values, just like other countries in the models, possibly due to observed
heteroscedaticity with the models.\n")
par(mfrow = c(2,1))
plot(model_df$country, mdl1$resid,
 main="Residuals by Country (mdl1)",
 xlab="Country",ylab="Residuals", cex.axis = 0.4)
abline(h=0,lty=2)
plot(data5$country, mdl2$resid,
 main="Residuals by Country (mdl2)",
 xlab="Country",ylab="Residuals", cex.axis = 0.4)
abline(h=0,lty=2)
text(20, -5e+6, 'LU', cex = 0.5, col = 'red')
text(22, -5e+6, 'NL', cex = 0.5, col = 'red')
```