Covid positive rate and Google searches

Introduction and scope

The objective of this mini project is to undertake a statistical study of the relation between the number of Google searches for "covid symptoms" in the United Kingdom and the rate of positive coronavirus tests at a given date. As we will see, there seems to exist a linear relation between these two variables, and we will construct a linear model that attempts to predict the positive rate from the relative amount of Google searches for "covid symptoms." Despite the model offering a satisfactory approximation for the UK data (which is the data used to train it), several factors prevent me from being optimistic about its application to other regions, and even possibly to the UK in the future:

1. The sample size, as we will see, might be too small to extract significant conclusions.
2. The model is sensitive to a change of conditions. For instance, different regions/countries may have different COVID-19 testing strategies, distinct COVID-19 spread prevention policies, their population may have more or less access to the Internet, etc. All of these factors may weigh on the reliability of the model when applied to the specific case of one of these regions.
3. In a similar vein, this model may not be able to accurately predict the evolution of the disease in the future even in the UK: indeed, a change of policies on how to combat the virus may require some future fine tuning.
4. I am looking for the relation between searches for the English term "covid symptoms" and the covid positive rate. However, a change on the term searched, even within the English language, will lead to a different model. It is not a trivial matter to find what expression one should be looking for in a different language.

There is, however, a continually changing variable other than the number of Googl searches for "covid symptoms" that I chose to include when building the model that substantially improves its accuracy: the vaccination rate. Unfortunately, it is just one of many factors that influence the quality of the predictions.

This mini project forms part of the Codecademy online course "Analyze Data with R" that I have been following, and should be regarded as an educational exercise rather than as a thorough statistical analysis. Serious research has already been undertaken on this topic: see for instance (Cinarka et al. 2021).

The data used in this project originates from two sources: the data relating to the Google searches is obtained via the package `gtrendsR`; the source of the COVID-19 related data is the Our World in Data database (Hannah Ritchie and Roser 2020).

Expectation

There are two reasons why an increase on the number of Google searches for "covid symptoms" should more or less coincide with periods with higher positive rates:

1. If a person recognises some symptoms that might be covid-related, it is natural to assume that they will check online whether this is the case or not.
2. Every time there is a surge in the number of cases, the media coverage on the virus will increase, which should lead to an increase of the number of Google searches for these terms.

Some notation I will use in this section:

$P(t)$: proportion of tests made that were positive (positive rate) at time t.

$I(t)$: number of infected people at time t.

$S(t)$: fraction of the population that is susceptible at time t.

$V(t)$: fraction of the population that has been fully vaccinated at time t.

$R(t)$: effective reproduction number at time t.

R_0: basic reproduction number.

$h(t)$: *hits*, a unit proportional to the amount of Google searches for "covid symptoms."

Δt: average amount of time during which a person remains infected.

We have the equality $R(t) = R_0 \cdot S(t)$. I am going to make the simplifying assumption that the population that cannot be infected coincides with that that has been fully vaccinated. The equality above then becomes $R(t) = R_0 \cdot (1 - V(t))$. Assuming that the positive rate at a time t is proportional to the proportion of infected population at some fixed time in the past, we have

$$P(t + \Delta t) \approx R_0 \cdot (1 - V(t)) \cdot P(t).$$

What we will see is that there seems to be a linear relation between $P(t + \varepsilon)$, for some fixed ε, and $h(t)$, which leads to a linear relation between $P(t + \varepsilon)$ and both $V(t)$ and $(1 - V(t)) \cdot h(t)$. Possibly due to the fact that the proportion of susceptible population is not exactly $(1 - V(t))$, or perhaps due to the fact that the vaccination is not uniform across different demographics, it will actually turn out that a linear model relating $P(t + \varepsilon)$ and $h(t)$ and $(1 - V(t)) \cdot h(t)$ is actually a better fit.

Statistical study

```
# Libraries that will be used.
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(modelr)
library(tidyr)
library(ggplot2)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(gtrendsR)
```

This is the Our World in Data (OWID) covid data set. We will later restrict to the UK data.

```
covid <- read.csv("owid-covid-data.csv")
```

```
covid <- covid %>%
  select(location, date, positive_rate, people_fully_vaccinated_per_hundred) %>%
  mutate(date = as.Date(date))
```

If one tried to use data from earlier in the pandemic, one would find that the amount of Google searches for "covid symptoms" were disproportionately higher than the amount of Google searches later in the pandemic for periods of time with the same incidence rate. This phenomenon is easily explained by the novelty factor associated to the virus at that point in time. For this reason, I will restrict myself to data relating to the dates from 01/12/2020 (12/01/2020 in American format) onward.

```
covid <- covid %>%
  filter(date >= "2020-12-01")
```

For periods of time between 8 months and 5 years, the dataframes provided the package **gtrendsR** only have weekly data. For that reason, instead of working with the daily data in the OWID data set, I will take the average positive rates and (full) vaccination rate for each week.

```
covid <- covid %>%
  group_by(location) %>%
  mutate(week = c(1:n())) %>%
  mutate(week = floor((week  + 6)/7)) %>%
  ungroup()
```

```
covid <- covid %>%
  filter(date <= "2021-08-11") %>%
  mutate(people_fully_vaccinated_per_hundred = replace_na(people_fully_vaccinated_per_hundred, 0)) %>%
  group_by(week, location) %>%
  summarize(avg_positive = mean(positive_rate), vaccination = mean(people_fully_vaccinated_per_hundred)
  ungroup()
```

```
## `summarise()` has grouped output by 'week'. You can override using the `.groups` argument.
```

```
covid_UK <- covid %>%
  filter(location == "United Kingdom")
```

This is now the **gtrendsR** data set.

```
gcovid_UK <- gtrends(
  keyword = "covid symptoms",
  geo = "GB",
  time = "2020-11-15 2021-08-15"
)$interest_over_time
summary(gcovid_UK)
```

```
##       date                          hits          keyword
##  Min.   :2020-11-15 00:00:00   Min.   : 19.00   Length:39
##  1st Qu.:2021-01-20 12:00:00   1st Qu.: 25.50   Class :character
##  Median :2021-03-28 00:00:00   Median : 44.00   Mode  :character
##  Mean   :2021-03-28 00:00:00   Mean   : 45.82
##  3rd Qu.:2021-06-02 12:00:00   3rd Qu.: 57.50
##  Max.   :2021-08-08 00:00:00   Max.   :100.00
##      geo                time              gprop              category
##  Length:39          Length:39          Length:39          Min.   :0
##  Class :character   Class :character   Class :character   1st Qu.:0
##  Mode  :character   Mode  :character   Mode  :character   Median :0
##                                                            Mean   :0
##                                                            3rd Qu.:0
##                                                            Max.   :0
```

We need to adjust the dates so that they coincide with these in the `covid` dataframe.

```
gcovid_UK <- gcovid_UK %>%
  select(date, hits) %>%
  mutate(date = as.Date(date) + 2)
```

```
gcovid_UK <- gcovid_UK %>%
  filter(date >= "2020-12-01")
gcovid_UK <- gcovid_UK %>%
  mutate(week = c(1:nrow(gcovid_UK)))
```

```
merged_UK <- covid_UK %>%
  inner_join(gcovid_UK)
```

```
## Joining, by = "week"
```

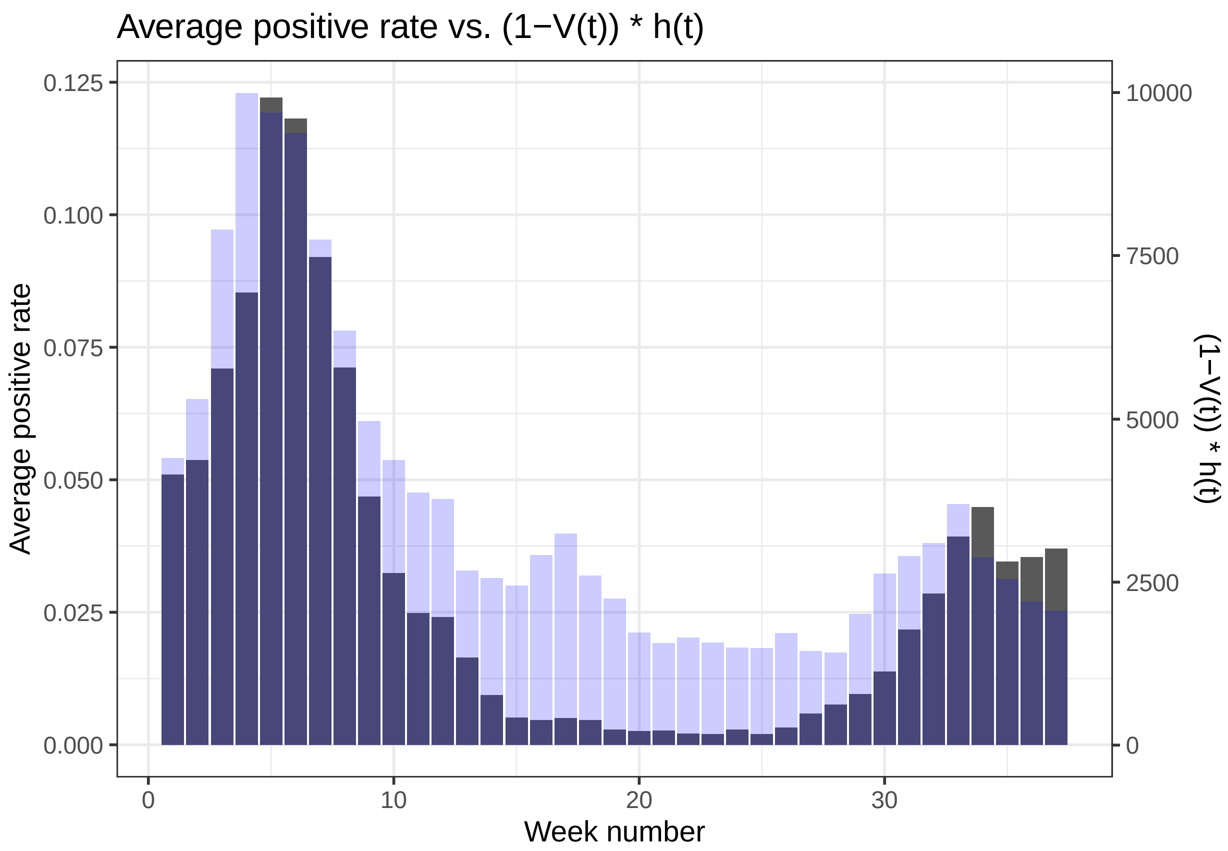
```
merged_UK <- merged_UK %>%
  mutate(hit_vac = (100-vaccination) * hits)
```

```
summary(merged_UK)
```

```
##       week      location          avg_positive       vaccination
##  Min.   : 1   Length:37          Min.   :0.002000   Min.   : 0.0000
##  1st Qu.:10   Class :character   1st Qu.:0.004714   1st Qu.: 0.7486
##  Median :19   Mode  :character   Median :0.021714   Median :10.1614
##  Mean   :19                      Mean   :0.030726   Mean   :20.7564
##  3rd Qu.:28                      3rd Qu.:0.044857   3rd Qu.:43.3514
##  Max.   :37                      Max.   :0.122143   Max.   :58.8150
##       date                 hits           hit_vac
##  Min.   :2020-12-01   Min.   : 19.00   Min.   : 1416
##  1st Qu.:2021-02-02   1st Qu.: 25.00   1st Qu.: 2007
##  Median :2021-04-06   Median : 39.00   Median : 2670
##  Mean   :2021-04-06   Mean   : 45.49   Mean   : 3634
##  3rd Qu.:2021-06-08   3rd Qu.: 58.00   3rd Qu.: 4367
##  Max.   :2021-08-10   Max.   :100.00   Max.   :10000
```

Let us visualise some plots in order to see how `hit_vac` relates to `avg_positive`. In the following graph, the blue bars represent the values of $(1 - V(t)) \cdot h(t)$, while the gray bars (dark blue when under the blue ones) represent the average positive rate for a given week.

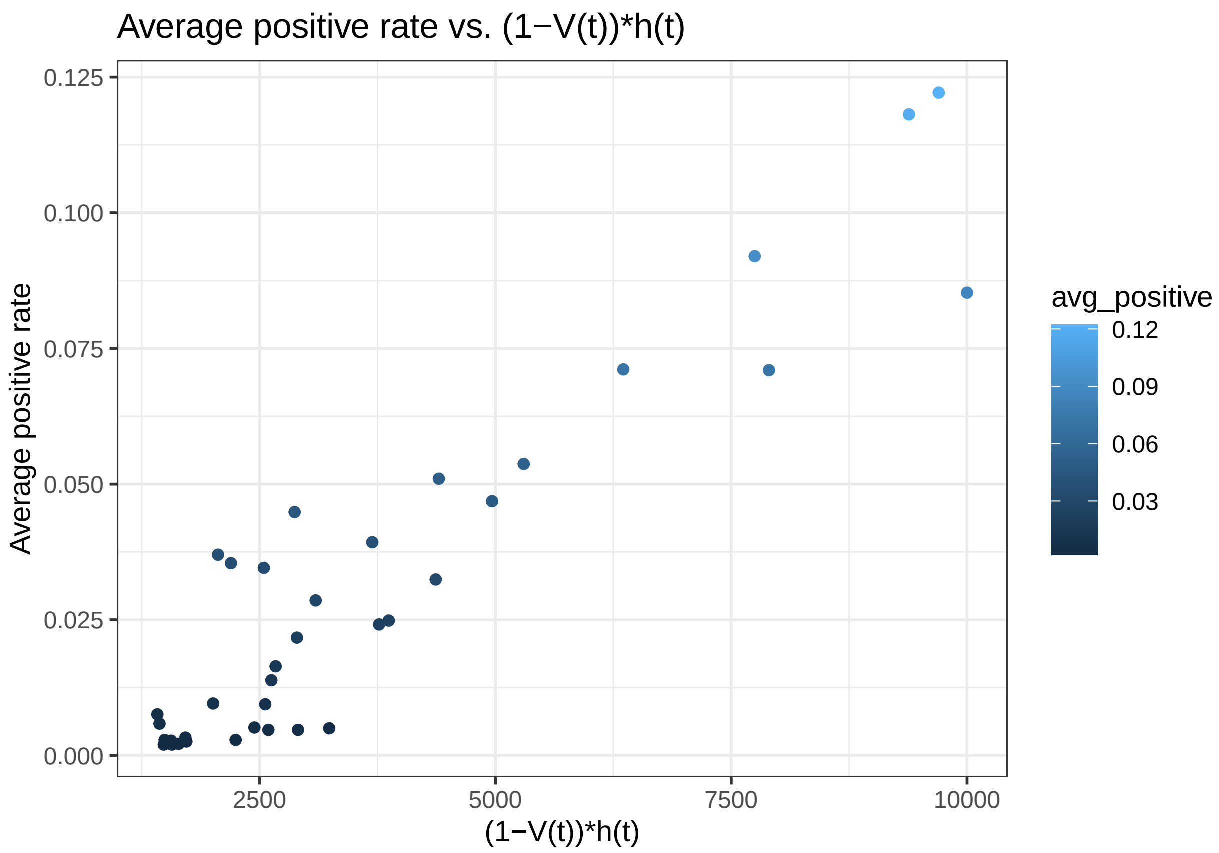
```
coeff = 10000/0.123
compare_plot <- ggplot(
  merged_UK,
  aes(x = week)
) +
  geom_bar(stat = "identity", aes(y = avg_positive)) +
  geom_bar(stat = "identity", fill = "blue", aes(y = hit_vac/coeff), alpha = 0.2) +
  scale_y_continuous(name = "Average positive rate",
                     sec.axis = sec_axis(~.*coeff, name = "(1-V(t)) * h(t)")) +

  labs(title = "Average positive rate vs. (1-V(t)) * h(t)", x = "Week number") +
  theme_bw()
compare_plot
```



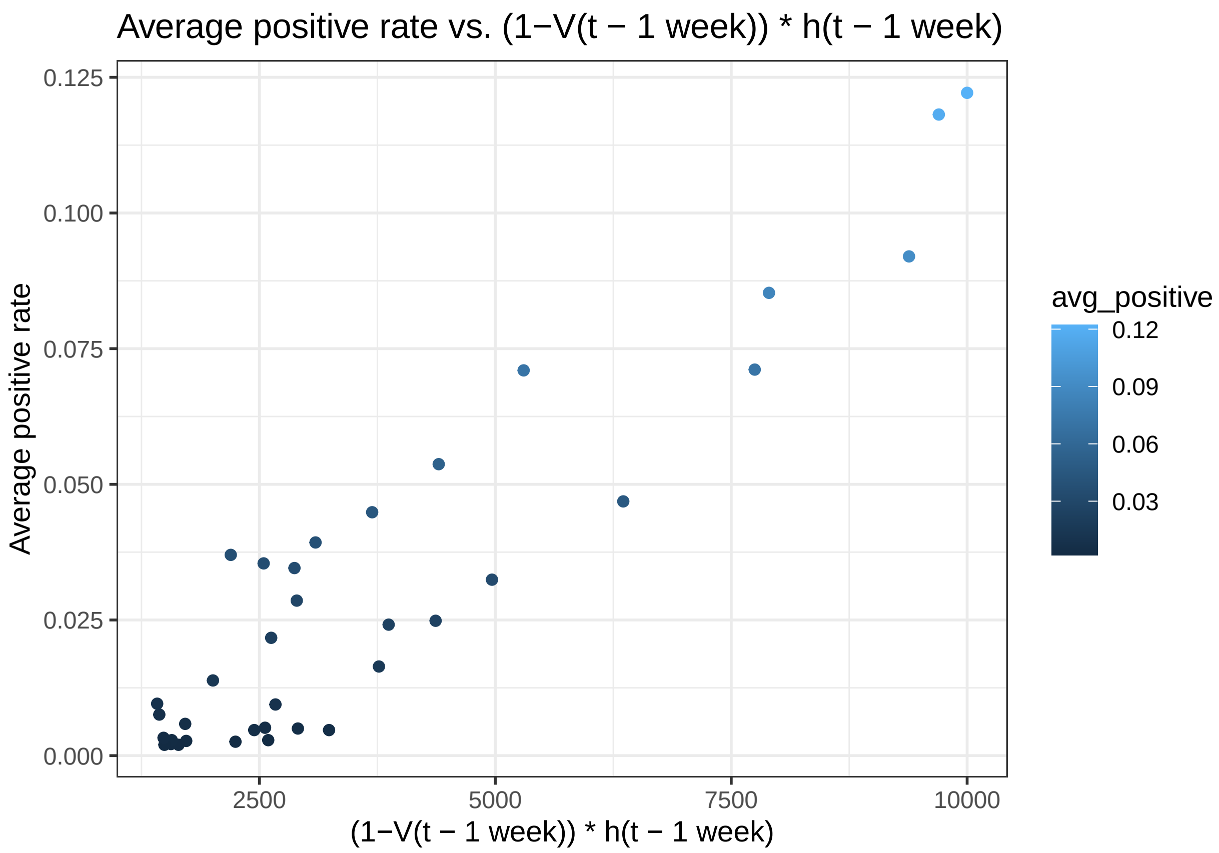
I will compare the current situation with the one obtained by "moving" the blue plot to the right, by not by means of a graphic plot. Instead, I will calculate the correlation coefficient.

```
merged_UK_lag <- merged_UK %>%
  mutate(hit_vac_lag = lag(hit_vac), hits_lag = lag(hits), vac_lag = lag(vaccination)) %>%
  filter(!is.na(hit_vac_lag))
```

```
scatter <- ggplot(merged_UK, aes(x = hit_vac, y = avg_positive)) +
  geom_point(aes(color = avg_positive)) +
  labs(title = "Average positive rate vs. (1-V(t))*h(t)", y = "Average positive rate", x = "(1-V(t))*h(
  theme_bw()
scatter
```



```
scatter_lag <- ggplot(merged_UK_lag, aes(x= hit_vac_lag, y = avg_positive)) +
  geom_point(aes(color = avg_positive)) +
  labs(title = "Average positive rate vs. (1-V(t - 1 week)) * h(t - 1 week)", y = "Average positive rat
  theme_bw()
scatter_lag
```



```
cor(merged_UK$hit_vac, merged_UK$avg_positive)
```

```
## [1] 0.9319822
```

```
cor(merged_UK_lag$hit_vac_lag, merged_UK_lag$avg_positive)
```

```
## [1] 0.9299234
```

Above are the correlation coefficients. Both `hit_vac` and `hit_vac_lag` appear to have a strong linear relation with `avg_positive`, something which is confirmed by the two scatter plots shown before.

Let us now try to build some models and compare them. Since the data set is very small, I will not split it into a training and a testing set.

```
model1 <- lm(avg_positive ~ hit_vac, merged_UK)
summary(model1)
```

```
##
## Call:
## lm(formula = avg_positive ~ hit_vac, data = merged_UK)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.025535 -0.007781 -0.001682  0.006169  0.026083
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.499e-02  3.608e-03  -4.155 0.000199 ***
## hit_vac      1.258e-05  8.272e-07  15.210  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01214 on 35 degrees of freedom
## Multiple R-squared:  0.8686, Adjusted R-squared:  0.8648
## F-statistic: 231.3 on 1 and 35 DF,  p-value: < 2.2e-16
```

```
model2 <- lm(avg_positive ~ hit_vac_lag, merged_UK_lag)
summary(model2)
```

```
##
## Call:
## lm(formula = avg_positive ~ hit_vac_lag, data = merged_UK_lag)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0199267 -0.0100166 -0.0007865  0.0092557  0.0254429
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.603e-02  3.756e-03  -4.267  0.00015 ***
## hit_vac_lag  1.256e-05  8.518e-07  14.745 2.46e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01243 on 34 degrees of freedom
## Multiple R-squared:  0.8648, Adjusted R-squared:  0.8608
## F-statistic: 217.4 on 1 and 34 DF,  p-value: 2.464e-16
```

It appears to be the case that, both in terms of the residual standard error and the R-squared value, the model involving the `hit_vac` is slightly better than the one using `hit_vac_lag`. But these models can still be improved by adding extra variables.

```
model3 <- lm(avg_positive ~ hit_vac + vaccination, merged_UK_lag)
summary(model3)
```

```
##
## Call:
## lm(formula = avg_positive ~ hit_vac + vaccination, data = merged_UK_lag)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0286899 -0.0053551  0.0000186  0.0063032  0.0136262
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.129e-02  4.227e-03  -7.403 1.67e-08 ***
## hit_vac      1.453e-05  7.456e-07  19.482  < 2e-16 ***
## vaccination  4.208e-04  8.337e-05   5.048 1.60e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.009287 on 33 degrees of freedom
## Multiple R-squared:  0.9267, Adjusted R-squared:  0.9223
## F-statistic: 208.7 on 2 and 33 DF,  p-value: < 2.2e-16
```

```
model4 <- lm(avg_positive ~ hit_vac_lag + vac_lag, merged_UK_lag)
summary(model4)
```

```
##
## Call:
## lm(formula = avg_positive ~ hit_vac_lag + vac_lag, data = merged_UK_lag)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0135835 -0.0052959  0.0000427  0.0026299  0.0265245
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.396e-02  3.861e-03  -8.795 3.64e-10 ***
## hit_vac_lag  1.480e-05  6.864e-07  21.559  < 2e-16 ***
## vac_lag      4.923e-04  7.884e-05   6.245 4.71e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.008542 on 33 degrees of freedom
## Multiple R-squared:  0.938,  Adjusted R-squared:  0.9343
## F-statistic: 249.7 on 2 and 33 DF,  p-value: < 2.2e-16
```

There is almost no difference between `model3` and `model4`.

```
model5 <- lm(avg_positive ~ hit_vac_lag + hits, merged_UK_lag)
summary(model5)
```

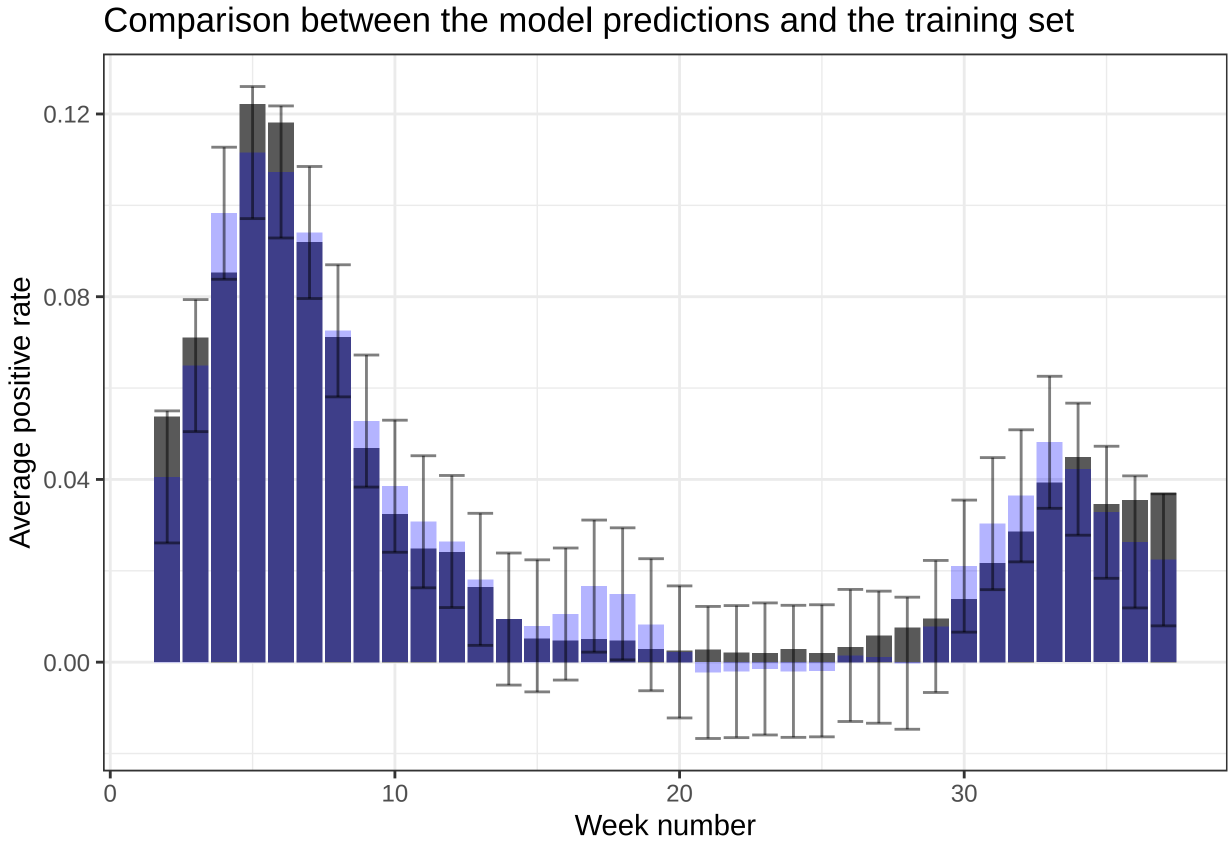
```
##
## Call:
## lm(formula = avg_positive ~ hit_vac_lag + hits, data = merged_UK_lag)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.012986 -0.005886  0.000159  0.004792  0.014592
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.785e-02  2.676e-03 -10.405 5.93e-12 ***
## hit_vac_lag  7.301e-06  8.308e-07   8.788 3.70e-10 ***
## hits         6.844e-04  8.582e-05   7.975 3.37e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.007375 on 33 degrees of freedom
## Multiple R-squared:  0.9538, Adjusted R-squared:  0.951
## F-statistic: 340.6 on 2 and 33 DF,  p-value: < 2.2e-16
```

Of all the linear models that I looked at, the one that seems to achieve the highest degree of accuracy and whose coefficient estimates maintain a high level of significance is `model5`. This model explains about 95% of the variance in the sample and has a residual standard error of approximately 0.007. Given that the average positive rate varies between 0.002 and 0.122 (approximately), the 95% confidence intervals are going to be wide relatively to the range of values of `avg_positive`.

Before moving on to trying to apply the model to other countries, let us have a look at a plot comparing the predictions given by our model and our training set. In blue, we can see the predictions of our model. In gray, the actual values of avg_positive. The 95% confidence intervals are represented by the black bars.

```
merged_UK_lag <- add_predictions(merged_UK_lag, model5)
```

```
ci = 1.96 * sigma(model5)
pred_plot <- ggplot(
  merged_UK_lag,
  aes(x = week)
) +
  geom_bar(stat = "identity", aes(y = avg_positive)) +
  geom_bar(stat = "identity", aes(y = pred), fill = "blue", alpha = 0.3) +
  labs(title = "Comparison between the model predictions and the training set",
       x = "Week number",
       y = "Average positive rate") +
  geom_errorbar(aes(ymin = pred - ci, ymax = pred + ci), color = "black", alpha = 0.5) +
  theme_bw()
pred_plot
```



Applying the model to the United States

```
covid_US <- covid %>%
  filter(location == "United States")
```

```
gcovid_US <- gtrends(
  keyword = "covid symptoms",
  geo = "US",
  time = "2020-11-15 2021-08-15"
)$interest_over_time
summary(gcovid_US)
```

```
##       date                          hits          keyword
##  Min.   :2020-11-15 00:00:00   Min.   : 16.00   Length:39
##  1st Qu.:2021-01-20 12:00:00   1st Qu.: 26.00   Class :character
##  Median :2021-03-28 00:00:00   Median : 34.00   Mode  :character
##  Mean   :2021-03-28 00:00:00   Mean   : 45.69
##  3rd Qu.:2021-06-02 12:00:00   3rd Qu.: 68.00
##  Max.   :2021-08-08 00:00:00   Max.   :100.00
##      geo                time              gprop              category
##  Length:39          Length:39          Length:39          Min.   :0
##  Class :character   Class :character   Class :character   1st Qu.:0
##  Mode  :character   Mode  :character   Mode  :character   Median :0
##                                                           Mean   :0
##                                                           3rd Qu.:0
##                                                           Max.   :0
```

```
gcovid_US <- gcovid_US %>%
  select(date, hits) %>%
  mutate(date = as.Date(date) + 2)
```

```
gcovid_US <- gcovid_US %>%
  filter(date >= "2020-12-01")
gcovid_US <- gcovid_US %>%
  mutate(week = c(1:nrow(gcovid_US)))
```

```
merged_US <- covid_US %>%
  inner_join(gcovid_US)
```

```
## Joining, by = "week"
```

```
merged_US <- merged_US %>%
  mutate(hit_vac = (100-vaccination) * hits)
```

```
summary(merged_US)
```

```
##       week      location          avg_positive      vaccination
##  Min.   : 1   Length:37          Min.   :0.01843   Min.   : 0.000
##  1st Qu.:10   Class :character   1st Qu.:0.03782   1st Qu.: 2.303
##  Median :19   Mode  :character   Median :0.05321   Median :20.471
##  Mean   :19                      Mean   :0.06540   Mean   :22.507
##  3rd Qu.:28                      3rd Qu.:0.09929   3rd Qu.:41.134
##  Max.   :37                      Max.   :0.13914   Max.   :49.930
##                                  NA's   :1
##       date                 hits          hit_vac
##  Min.   :2020-12-01   Min.   :16.00   Min.   : 891.1
##  1st Qu.:2021-02-02   1st Qu.:26.00   1st Qu.:1472.7
##  Median :2021-04-06   Median :34.00   Median :2810.7
##  Mean   :2021-04-06   Mean   :42.97   Mean   :3582.1
##  3rd Qu.:2021-06-08   3rd Qu.:57.00   3rd Qu.:4396.4
##  Max.   :2021-08-10   Max.   :92.00   Max.   :9200.0
##
```

```
merged_US <- merged_US %>%
```

```r
  mutate(hit_vac_lag = lag(hit_vac))
merged_US <- merged_US %>%
  filter(!is.na(hit_vac_lag) & !is.na(avg_positive))
summary(merged_US)
```

```
##       week        location          avg_positive      vaccination    
##  Min.   : 2.0   Length:35          Min.   :0.01843   Min.   : 0.000  
##  1st Qu.:10.5   Class :character   1st Qu.:0.03650   1st Qu.: 2.662  
##  Median :19.0   Mode  :character   Median :0.05314   Median :20.471  
##  Mean   :19.0                      Mean   :0.06416   Mean   :22.367  
##  3rd Qu.:27.5                      3rd Qu.:0.09071   3rd Qu.:40.548  
##  Max.   :36.0                      Max.   :0.13914   Max.   :49.607  
##       date                 hits          hit_vac        hit_vac_lag    
##  Min.   :2020-12-08   Min.   :16.00   Min.   : 891.1   Min.   : 891.1  
##  1st Qu.:2021-02-05   1st Qu.:24.50   1st Qu.:1411.8   1st Qu.:1411.8  
##  Median :2021-04-06   Median :34.00   Median :2710.6   Median :2710.6  
##  Mean   :2021-04-06   Mean   :40.89   Mean   :3432.4   Mean   :3588.8  
##  3rd Qu.:2021-06-04   3rd Qu.:53.50   3rd Qu.:4137.8   3rd Qu.:4860.0  
##  Max.   :2021-08-03   Max.   :92.00   Max.   :9200.0   Max.   :9200.0  
```

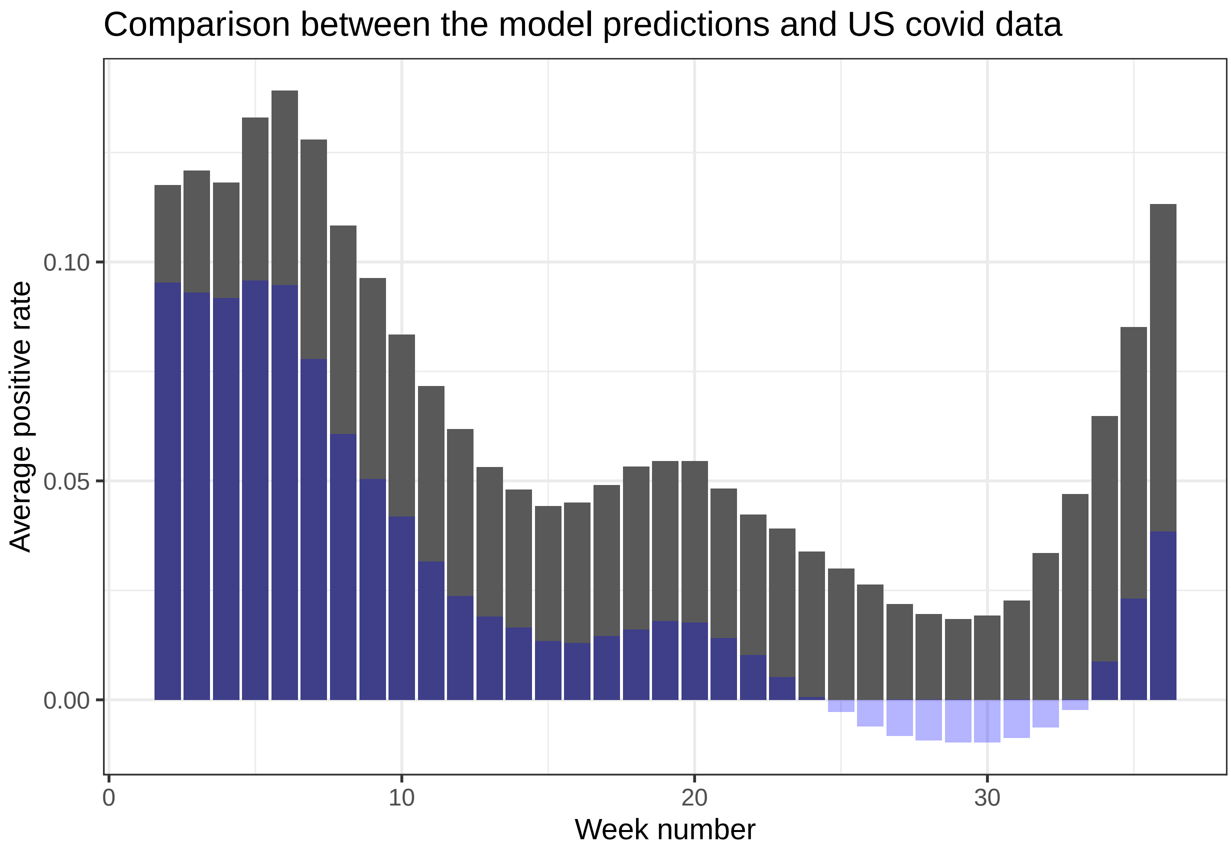
```r
merged_US <- add_predictions(merged_US, model5)
```

```r
summary(merged_US)
```

```
##       week        location          avg_positive      vaccination    
##  Min.   : 2.0   Length:35          Min.   :0.01843   Min.   : 0.000  
##  1st Qu.:10.5   Class :character   1st Qu.:0.03650   1st Qu.: 2.662  
##  Median :19.0   Mode  :character   Median :0.05314   Median :20.471  
##  Mean   :19.0                      Mean   :0.06416   Mean   :22.367  
##  3rd Qu.:27.5                      3rd Qu.:0.09071   3rd Qu.:40.548  
##  Max.   :36.0                      Max.   :0.13914   Max.   :49.607  
##       date                 hits          hit_vac        hit_vac_lag    
##  Min.   :2020-12-08   Min.   :16.00   Min.   : 891.1   Min.   : 891.1  
##  1st Qu.:2021-02-05   1st Qu.:24.50   1st Qu.:1411.8   1st Qu.:1411.8  
##  Median :2021-04-06   Median :34.00   Median :2710.6   Median :2710.6  
##  Mean   :2021-04-06   Mean   :40.89   Mean   :3432.4   Mean   :3588.8  
##  3rd Qu.:2021-06-04   3rd Qu.:53.50   3rd Qu.:4137.8   3rd Qu.:4860.0  
##  Max.   :2021-08-03   Max.   :92.00   Max.   :9200.0   Max.   :9200.0  
##       pred          
##  Min.   :-0.0097691  
##  1st Qu.:-0.0008927  
##  Median : 0.0159944  
##  Mean   : 0.0263386  
##  3rd Qu.: 0.0401079  
##  Max.   : 0.0957175  
```

In the plot below, blue represents the model predictions and gray stands for the average positive rate in the US.

```r
pred_US <- ggplot(
  merged_US,
  aes(x = week)
) +
  geom_bar(stat = "identity", aes(y = avg_positive)) +
  geom_bar(stat = "identity", aes(y = pred), fill = "blue", alpha = .3) +
  labs(title = "Comparison between the model predictions and US covid data", x = "Week number", y = "Ave
  theme_bw()
pred_US
```

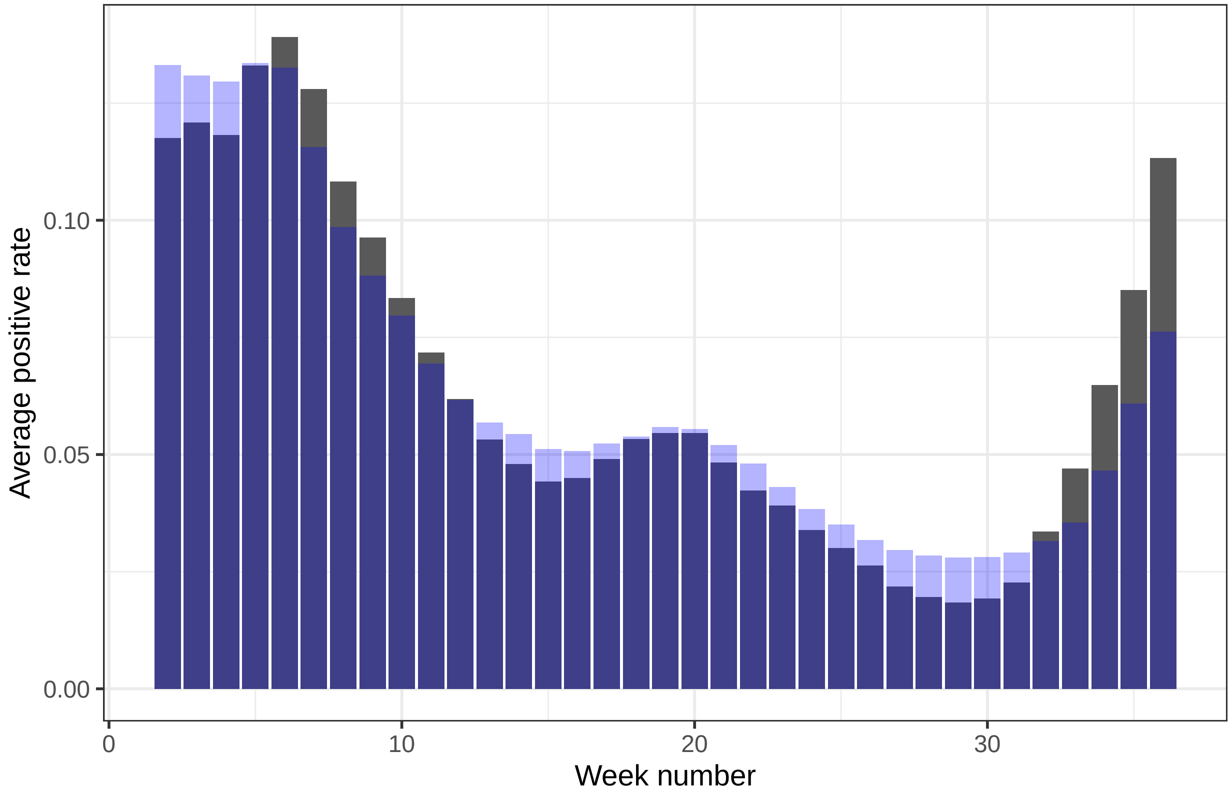


This is clearly unsatisfactory. However, we see that our model correctly predicts the shape of the curve. Perhaps we can try to tweak the model by adding to it the average amount of the absolute value of the error.

```
merged_US <- merged_US %>%
  mutate(error = avg_positive - pred)
avg_error <- mean(merged_US$error)
merged_US <- merged_US %>%
  mutate(pred2 = pred + avg_error)
```

```
pred_US <- ggplot(
  merged_US,
  aes(x = week)
) +
  geom_bar(stat = "identity", aes(y = avg_positive)) +
  geom_bar(stat = "identity", aes(y = pred2), fill = "blue", alpha = .3) +
  labs(title = "Comparison between the model predictions and US covid data", x = "Week number", y = "Ave
  theme_bw()
pred_US
```

Comparison between the model predictions and US covid data



This seems to be much better. In terms of the model, adding `avg_error` to the predictions amounts to changing the intercept value, but keeping the other coefficients fixed. Let's see what the residual standard error of this model with respect to the US covid data, and then let us compare it with the residual standard data of the linear model calculated by R for this data set.

```
merged_US <- merged_US %>%
  mutate(res2 = (pred2 - avg_positive)^2)
```

```
rse <- sqrt(sum(merged_US$res2)/(nrow(merged_US)-3))
rse
```

```
## [1] 0.0110695
```

```
model_US <- lm(avg_positive ~ hit_vac_lag + hits, merged_US)
summary(model_US)
```

```
##
## Call:
## lm(formula = avg_positive ~ hit_vac_lag + hits, data = merged_US)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.017289 -0.004214 -0.001970  0.006960  0.018322
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 8.468e-04  3.227e-03   0.262    0.795
## hit_vac_lag 1.654e-06  1.362e-06   1.214    0.233
## hits        1.403e-03  1.602e-04   8.760 5.21e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.008671 on 32 degrees of freedom
## Multiple R-squared:  0.9489, Adjusted R-squared:  0.9457
## F-statistic:   297 on 2 and 32 DF,  p-value: < 2.2e-16
```

The residual standard error of our model when applied to US data is relatively much higher than the one associated to the model calculated by R for the US data. It is also interesting to see that neither the intercept value nor the coefficient associated to `hit_vac_lag` are significant in the US model, contrasting with the UK situation. But even the estimated value of the `hits` coefficient, which carries a very high significance, is very different from the ones obtained in `model5`.

Conclusion

As mentioned in the introduction, the sample size I worked with is too small to be able to extract any relevant conclusions from the study undertaken above. In particular, I did not split the data is test and train sets. It would be interesting to use the model in a few months time to compare its predictions with new UK covid data. One should not be over confident though because, as I have also mentioned, there is the potential for any change of policies or otherwise to affect the reliability of this model.

References

Cinarka, Halit, Mehmet Atilla Uysal, Atilla Cifter, Elif Yelda Niksarlioglu, and Aslı Çarkoğlu. 2021. "The Relationship Between Google Search Interes for Pulmonary Symptoms and COVID-19 Cases Using Dynamic Conditional Correlation Analysis." *Scientific Reports* 11. https://doi.org/10.1038/s41598-021-93836-y.

Hannah Ritchie, Diana Beltekian, Esteban Ortiz-Ospina, and Max Roser. 2020. "Coronavirus Pandemic (COVID-19)." *Our World in Data*.