Vaccination in the UK

Elena Basargina

Contents

1 Step 1 Find dataset

- Create a list of metrics for each dataset

- Look at the metrics

New people receiving 2nd dose
New people vaccinated with a booster dose by publish date
New people vaccinated complete by publish date
New people fully vaccinated by vaccination date
New people vaccinated 1st dose by publish date

New people vaccinated with a first dose by vaccination date
New people vaccinated 2nd dose by publish date
New people vaccinated with a second dose by vaccination date
New people vaccinated with a third dose by publish date
New people vaccinated with a booster dose plus new people vaccinated with a third dose by publish date

New people vaccinated with a booster or third dose by vaccination date
New vaccines given by publish date

Table 1: Lower Tier Local Authority (LTLA)

New people receiving 1st dose
New people receiving 2nd dose
New people vaccinated with a booster dose by publish date
New people vaccinated complete by publish date
New people fully vaccinated by vaccination date

New people vaccinated 1st dose by publish date
New people vaccinated with a first dose by vaccination date
New people vaccinated 2nd dose by publish date
New people vaccinated with a second dose by vaccination date
New people vaccinated with a third dose by publish date

New people vaccinated with a booster dose plus new people vaccinated with a third dose by publish date
New people vaccinated with a booster or third dose by vaccination date
New vaccines given by publish date

Table 2: Nation

So, as we can see, **some metrics are common**. I suggest finding out which metrics are the same for all datasets.

- Add new metrics in a common list

- Build zero-matrix, which dimension is the count of metrics x the count of area types

- Show links

- Look at the result

	ltla	msoa	nation	nhsRegion	nhsTrust	overview	region	utla
New people receiving 2nd dose	1	0	1	0	0	0	1	0
New people vaccinated with a booster dose by publish date	1	0	1	0	0	0	1	0
New people vaccinated complete by publish date	1	0	1	0	0	0	1	0
New people fully vaccinated by vaccination date	1	0	1	0	0	0	1	0
New people vaccinated 1st dose by publish date	1	0	1	0	0	0	1	0
New people vaccinated with a first dose by vaccination date	1	0	1	0	0	0	1	0
New people vaccinated 2nd dose by publish date	1	0	1	0	0	0	1	0
New people vaccinated with a second dose by vaccination date	1	0	1	0	0	0	1	0
New people vaccinated with a third dose by publish date	1	0	1	0	0	0	1	0
New people vaccinated with a booster dose plus new people vaccinated with a third dose by publish date	1	0	1	0	0	0	1	0
New people vaccinated with a booster or third dose by vaccination date	1	0	1	0	0	0	1	0
New vaccines given by publish date	1	0	1	0	0	0	1	0
New people receiving 1st dose	0	0	1	0	0	0	1	0

First of all, I am interested in data about the **first jab**. **So, I need to look at the datasets:**

Lower Tier Local Authority (LTLA)
Nation
Region

2 Step 2 Ask something

I live in Bristol. What do I know about Bristol?

- This city is a part of the UK, England, and South West.
- There are two universities.
- The city rests every summer when students come back to their homes and works hardly elsewhen.

So, I am interested in data about the UK, England, South West, and Bristol.

Question 0: Are there dependencies between academic year events and vaccination waves? Does the vaccination depend on holidays?

I was vaccinated by

- the first dose on 8 August 2021,
- the second dose on 3 October 2021,
- the booster dose on 8 January 2022.

Question 1: How many people got their jabs with me?

I got the first and the second jabs on Sunday. There were fewer people in the vaccination centre. When I got the third jab on Saturday, there was a big queue.

Question 2: When do people prefer to get a jab: weekdays or weekends/Saturdays or Sundays?

Question 3: Is there something illogical in data?

3 Step 3 Look at the datasets

3.1 Region

As we can see on the website, Region metrics are available for regions of England. I am interested in the South West and metrics that start with "New":

areaCode
areaName
areaType
date
newPeopleVaccinatedFirstDoseByVaccinationDate
newPeopleVaccinatedSecondDoseByVaccinationDate
newPeopleVaccinatedThirdInjectionByVaccinationDate

We have additional columns. Let's look at them.

- For **areaCode** unique value is $E12000009$,

- for **areaName** unique value is $SouthWest$,

- for **areaType** unique value is $region$.

So, we do not need to look at them in the future because these columns are used for filtering that we have already done on the website.

Let's prepare data for the plotting.

- Rename columns and columns

- Add the column MonthYear

- Create long table

date	MonthYear	Dose	Count
2022-01-26	1.2022	First	986
2022-01-25	1.2022	First	899
2022-01-24	1.2022	First	723
2022-01-23	1.2022	First	1035
2022-01-22	1.2022	First	1822
2022-01-21	1.2022	First	1085

Let's plot something.

3.1.1 Question 0

Are there dependencies between academic year events and vaccination waves? Does the vaccination depend on holidays?
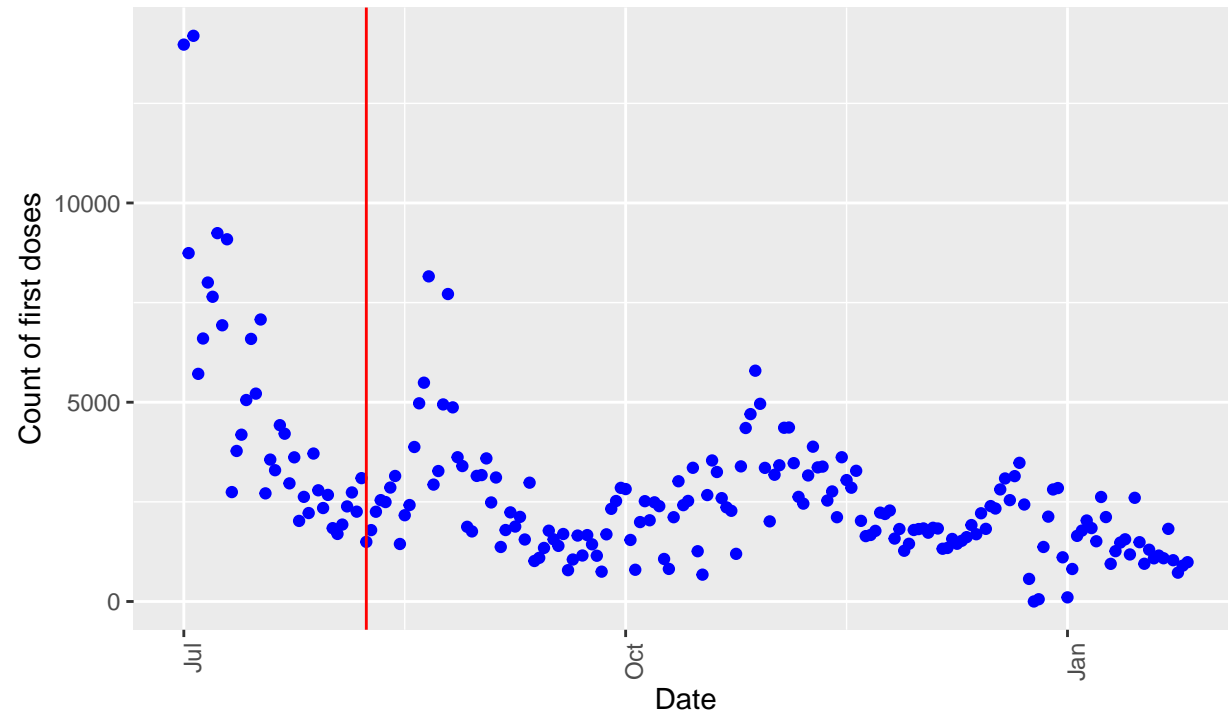
Figure 1: My caption
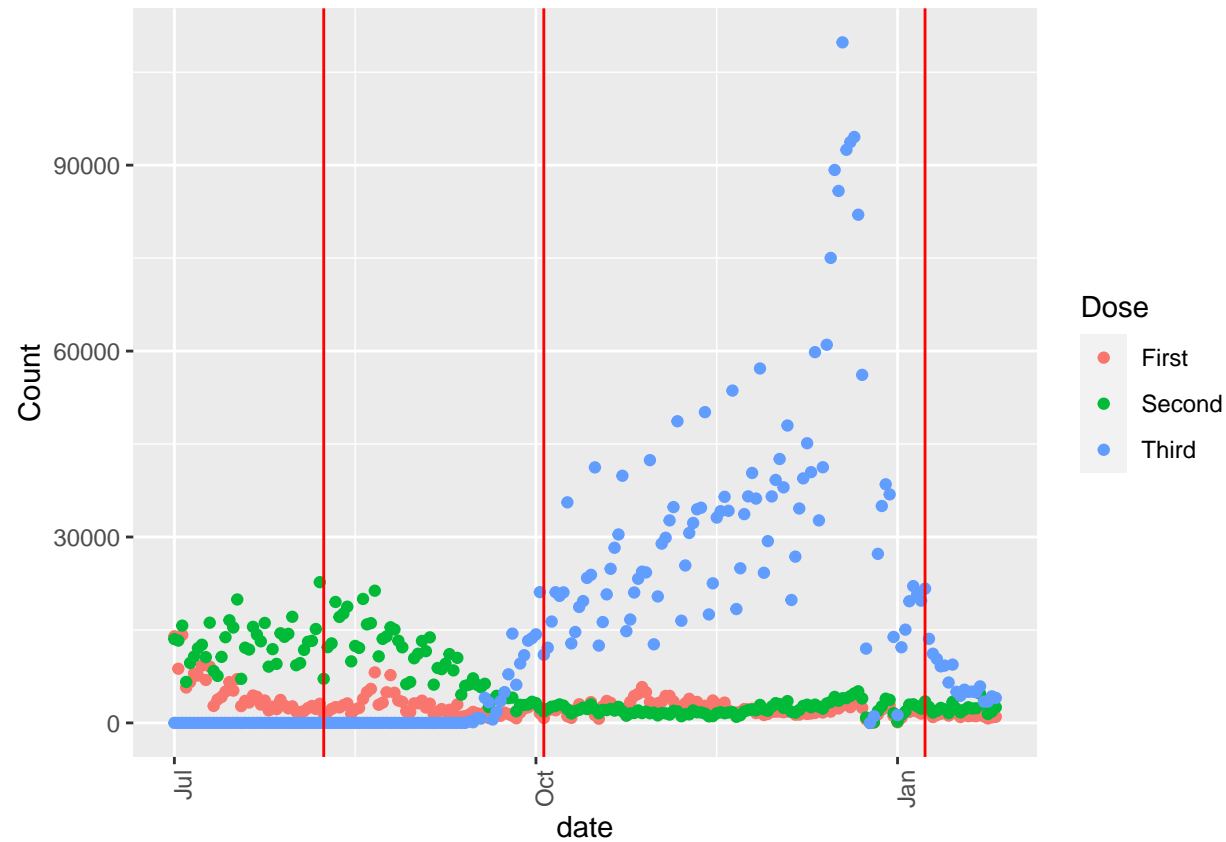
The result is not beautiful because of the active growth of the third jabs count at the end of 2021.

Let's plot them separately.
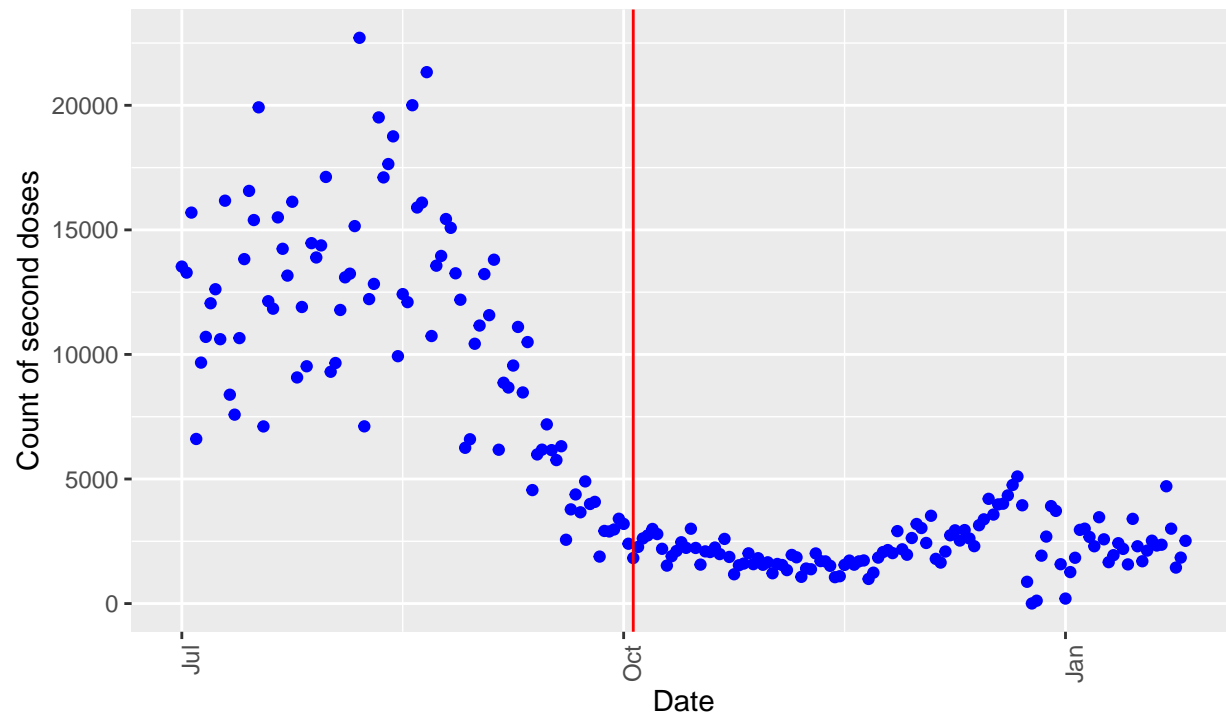
Vaccination in South West

The first dose

More information https://coronavirus.data.gov.uk/details/about−data

It is so interesting why the graph is wavy. 1496 people got their first jabs with me.

Vaccination in South West

The second dose

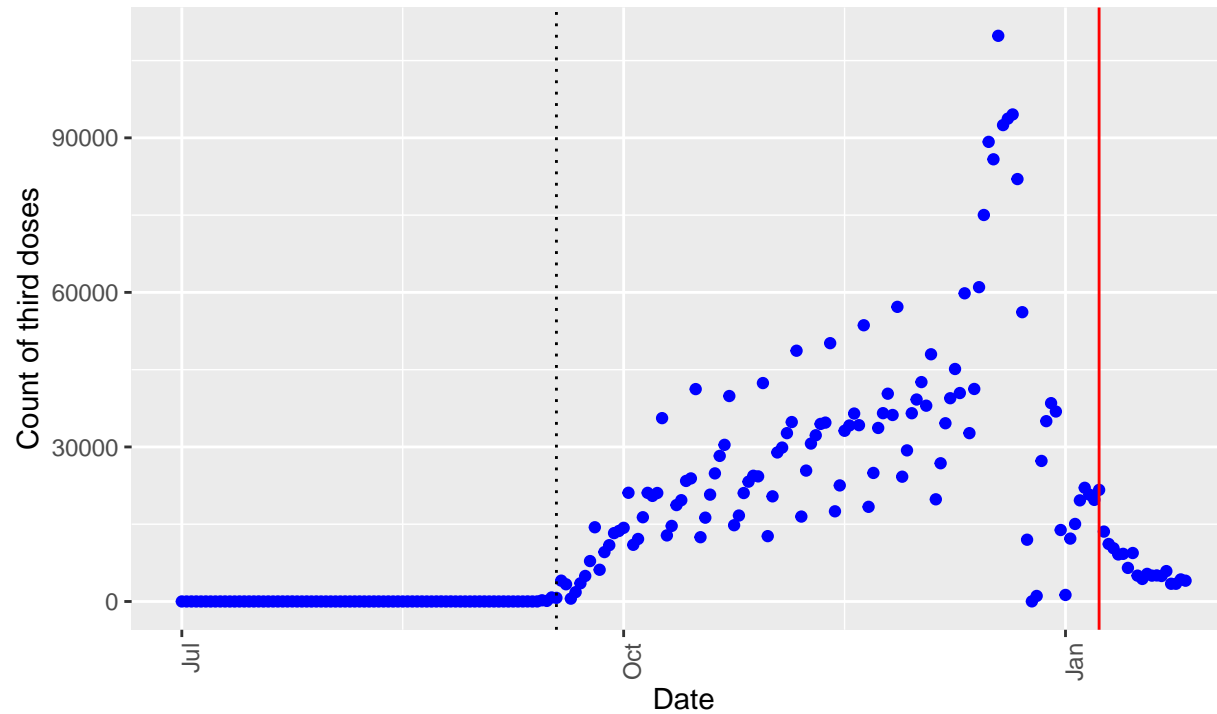More information https://coronavirus.data.gov.uk/details/about−data

1828 people got their second jabs with me.

Vaccination in South West
The third dose
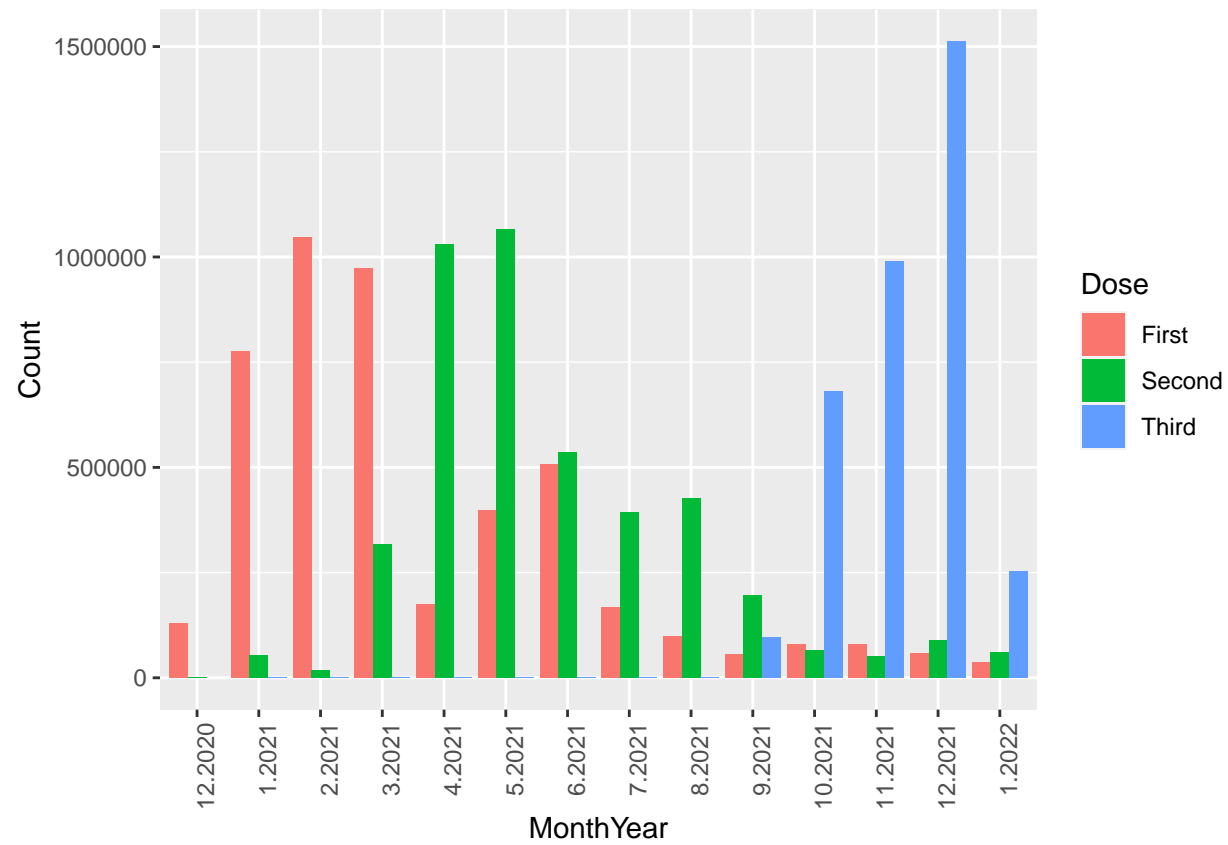
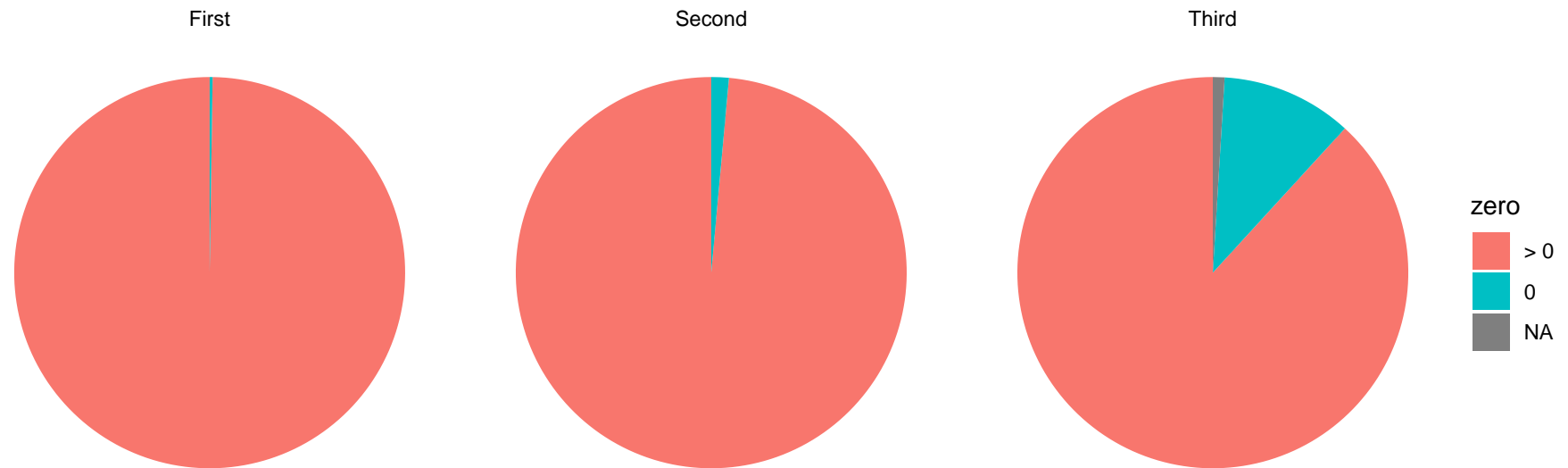21664 people got their third jabs with me. We can see when the active phase of vaccination by the third dose started.

Let's calculate the date. ("2021-09-17")

```
## Warning: Removed 1 rows containing missing values (geom_col).
```

Figure 2:   My caption

3.1.2 Zeroes

The column "Third" has more zero values than "First" and "Second; but, I think, it won't influence models' accuracy. Also, we can see missing values for the column"Third"; in our case, missing values mean that nobody got the third jab. I suggest replacing them with zeroes.

Replace missing values.

3.1.3 Data description

Median, percentiles and mean

	mean	median	Q0.25	Q0.75	Q0.9
First	11037.614	4210	2072.0	14213.0	31224.6
Second	10367.352	3998	1659.5	13708.5	28700.6
Third	8589.925	6	2.0	10629.5	33689.0

What can I say?

- Mean and median have a visible difference. So, there are large extreme values.

- For the Third dose, half of the values are below 6. That is not surprised. In the beginning, people needed to get two jabs.

- If we look at "Q0.25", "Q0.75", "Q0.90", we find out that the Third dose's wave caught up with other doses' waves quickly. We already saw this fact on the plot 1.
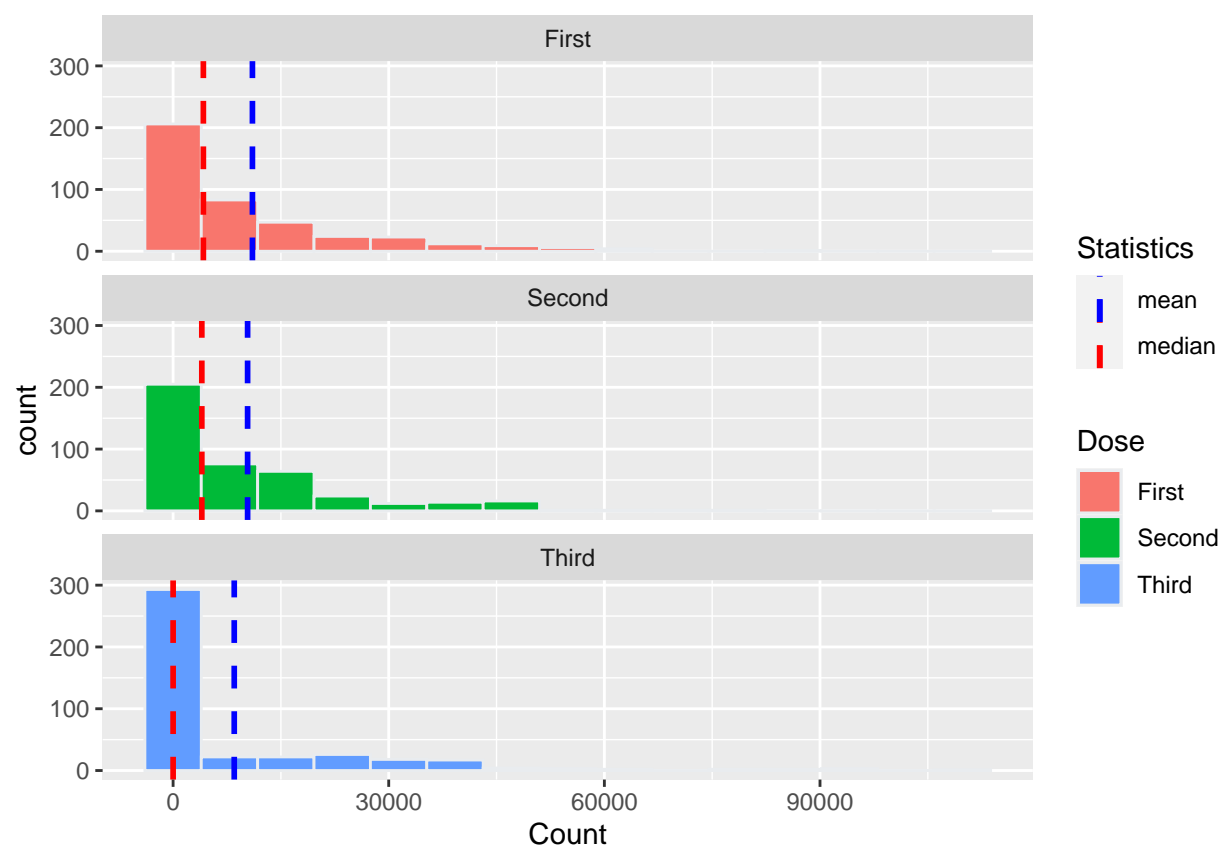
Standard deviation (sd), IQR and range

	sd	range	IQR
First	13971.58	84537	12141.0
Second	13477.22	78425	12049.0
Third	17512.10	109810	10627.5

IQR and standard deviation for each dose are big, consequently, the data spread out. Also, we can see the difference between largest and smallest values in the column "range".

3.2 Bristol

The dataset's columns:

areaCode
areaName
areaType
date
age
VaccineRegisterPopulationByVaccinationDate
cumPeopleVaccinatedCompleteByVaccinationDate
newPeopleVaccinatedCompleteByVaccinationDate
cumPeopleVaccinatedFirstDoseByVaccinationDate
newPeopleVaccinatedFirstDoseByVaccinationDate
cumPeopleVaccinatedSecondDoseByVaccinationDate
newPeopleVaccinatedSecondDoseByVaccinationDate
cumPeopleVaccinatedThirdInjectionByVaccinationDate
newPeopleVaccinatedThirdInjectionByVaccinationDate
cumVaccinationFirstDoseUptakeByVaccinationDatePercentage
cumVaccinationSecondDoseUptakeByVaccinationDatePercentage
cumVaccinationThirdInjectionUptakeByVaccinationDatePercentage
cumVaccinationCompleteCoverageByVaccinationDatePercentage

We have additional columns. Let's look at them.

There are 1 unique value for **areaCode**, 1 unique value for **areaName**, and 1 unique value for **areaType** as well. So, we do not need to look at them in the future because these columns are used for filtering that we have already done on the website.

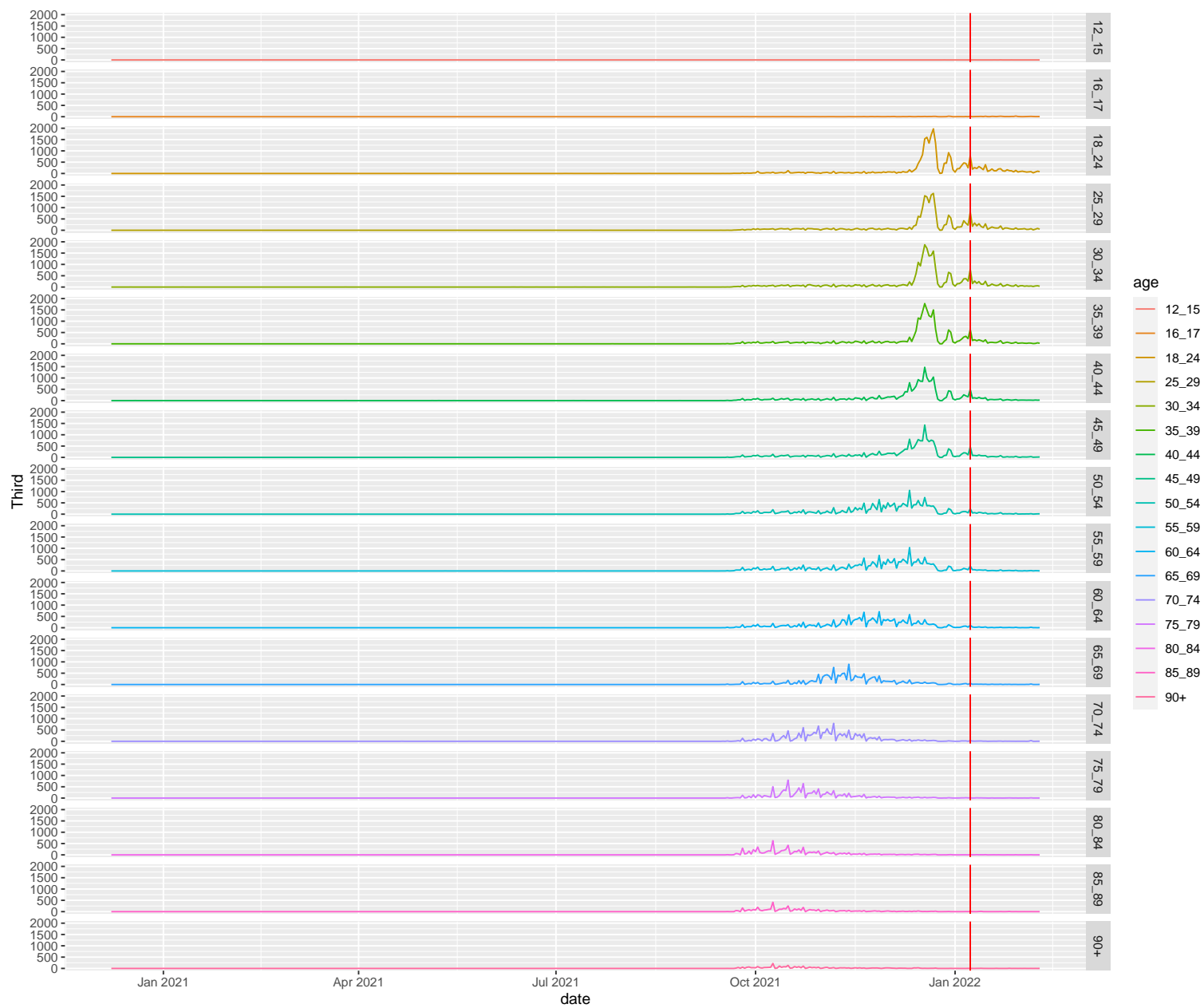Just rename columns and we will move on to answer the questions.

age	date	First	Second	Third
12_15	2022-02-09	28	45	0
16_17	2022-02-09	4	8	6
18_24	2022-02-09	25	18	83
25_29	2022-02-09	11	8	48
30_34	2022-02-09	4	10	37
35_39	2022-02-09	3	8	26

16 people in my age group got their First jabs with me in Bristol.
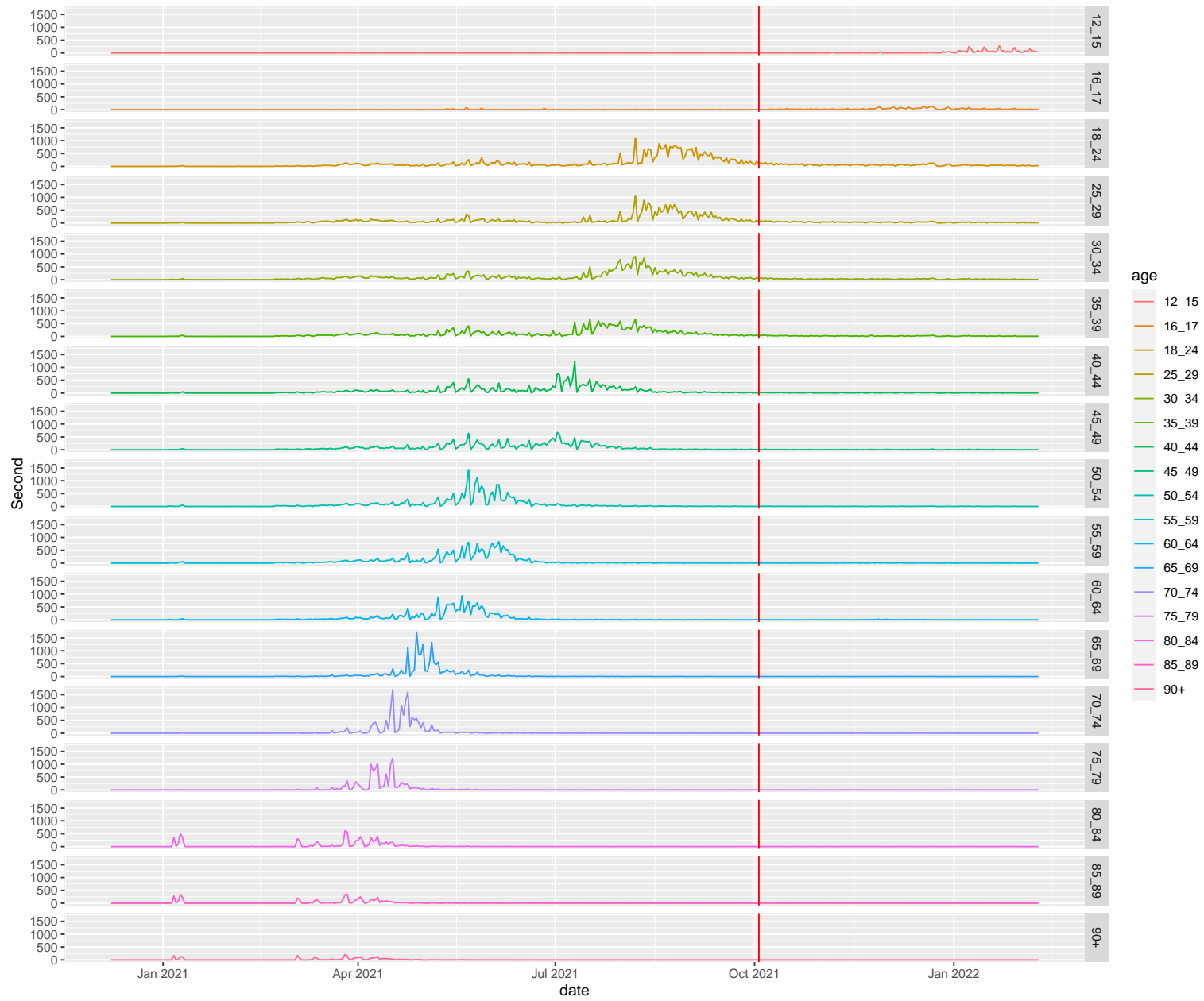
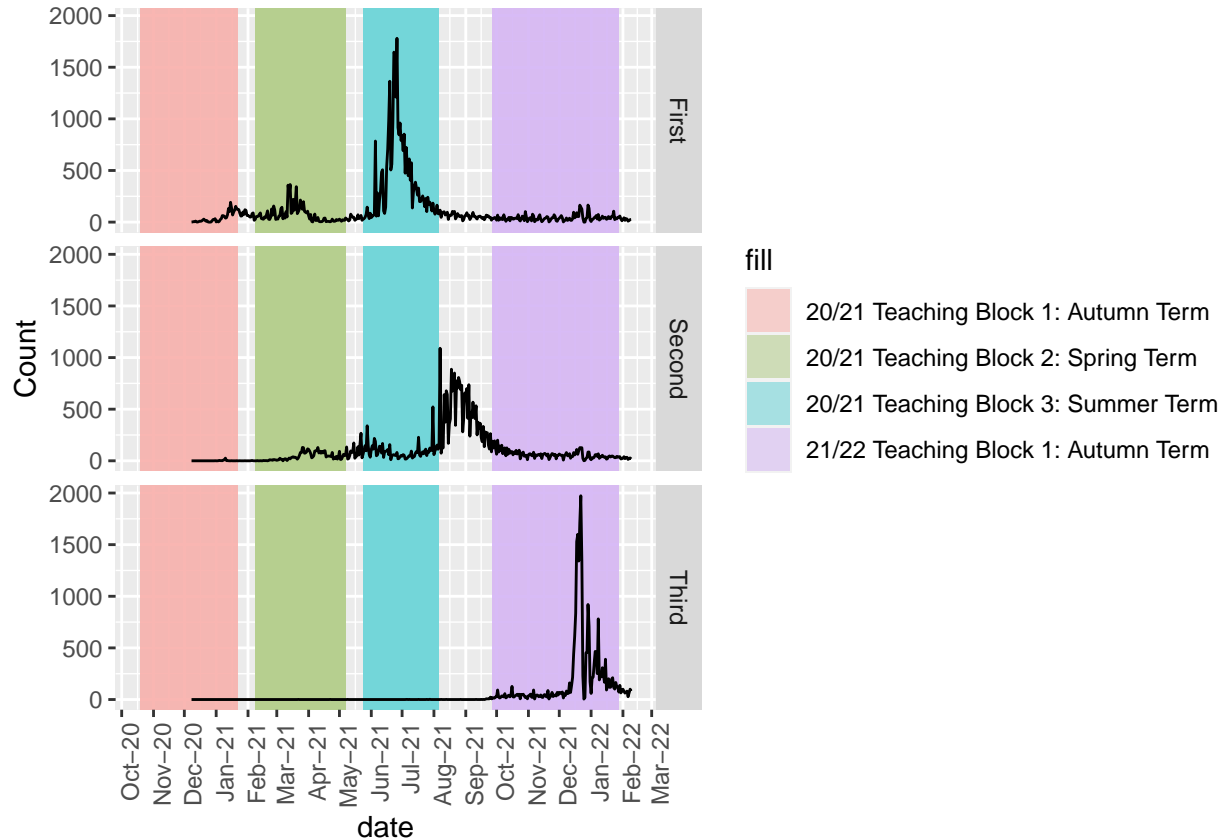37 people in my age group got their Second jabs with me in Bristol.

806 people in my age group got their Third jabs with me in Bristol.

Students

https://www.uwe.ac.uk/study/term-dates/2020-21-term-dates

https://www.uwe.ac.uk/study/term-dates/2021-22-term-dates

https://www.uwe.ac.uk/study/term-dates/2022-23-term-dates

3.3 England

Look at the dataset's columns.

areaCode
areaName
areaType
date
newPeopleVaccinatedFirstDoseByVaccinationDate
newPeopleVaccinatedSecondDoseByVaccinationDate
newPeopleVaccinatedThirdInjectionByVaccinationDate

We are not going to look at the columns areaCode, areaName, areaType, because these columns have one unique value ($E92000001$ for areaCode, *England* for areaName, *nation* for areaType), they are used for filtering on the website.

Rename columns and exclude unnecessary columns.

|	Date	First	Second	Third
457	2020-12-08	5370	146	NA
456	2020-12-09	9648	137	2
455	2020-12-10	11888	119	1
454	2020-12-11	12516	82	1
453	2020-12-12	10565	23	3
452	2020-12-13	6134	30	0

I want to be sure that the file has statistics for every day.

- Add the column with difference between the date and the previous date within the table.

The difference between dates is 1. So, we have full statistics by date.

Do we have missing values?

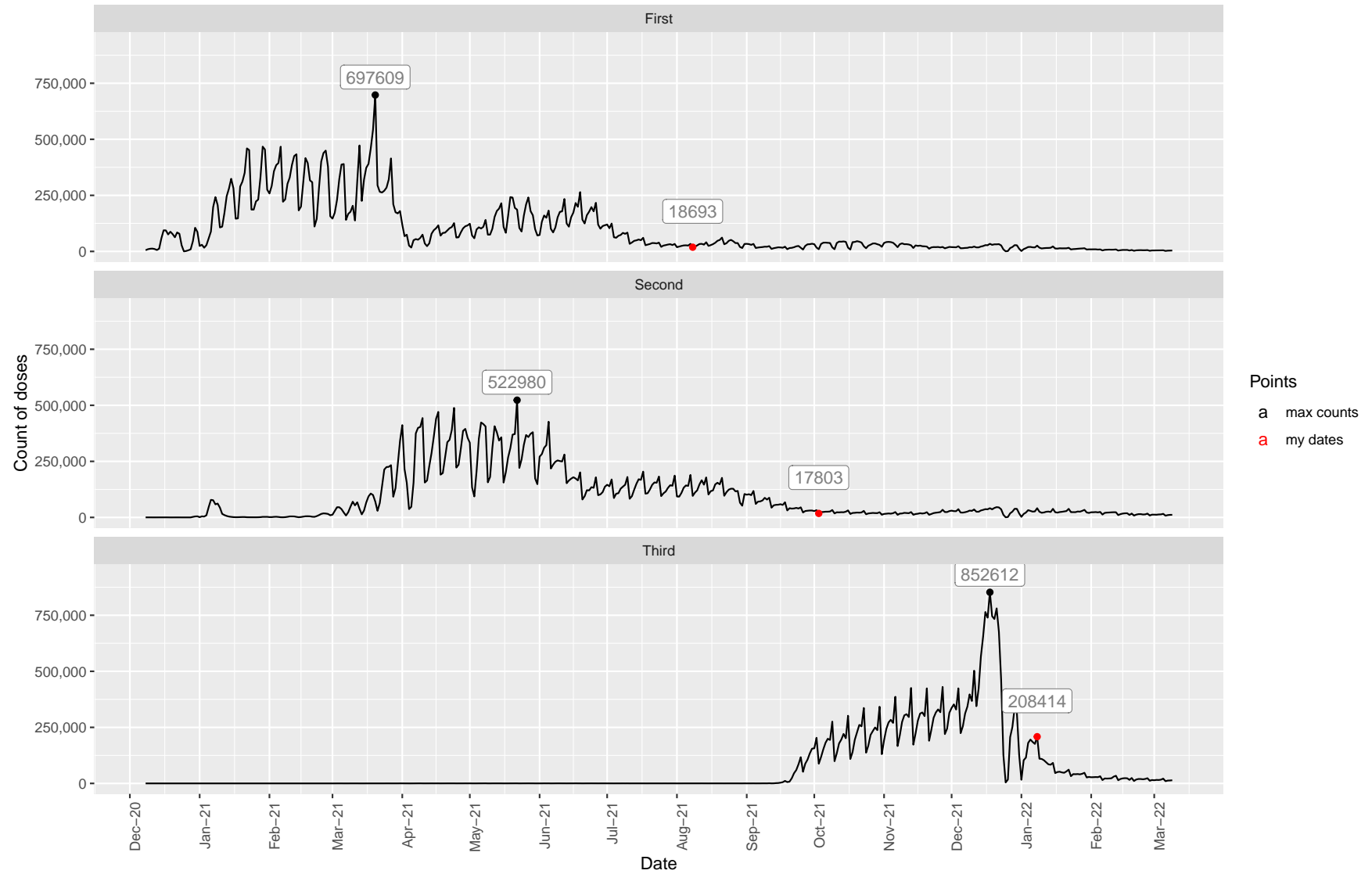|	
Date	0
First	0
Second	0
Third	1
Diffdate	0
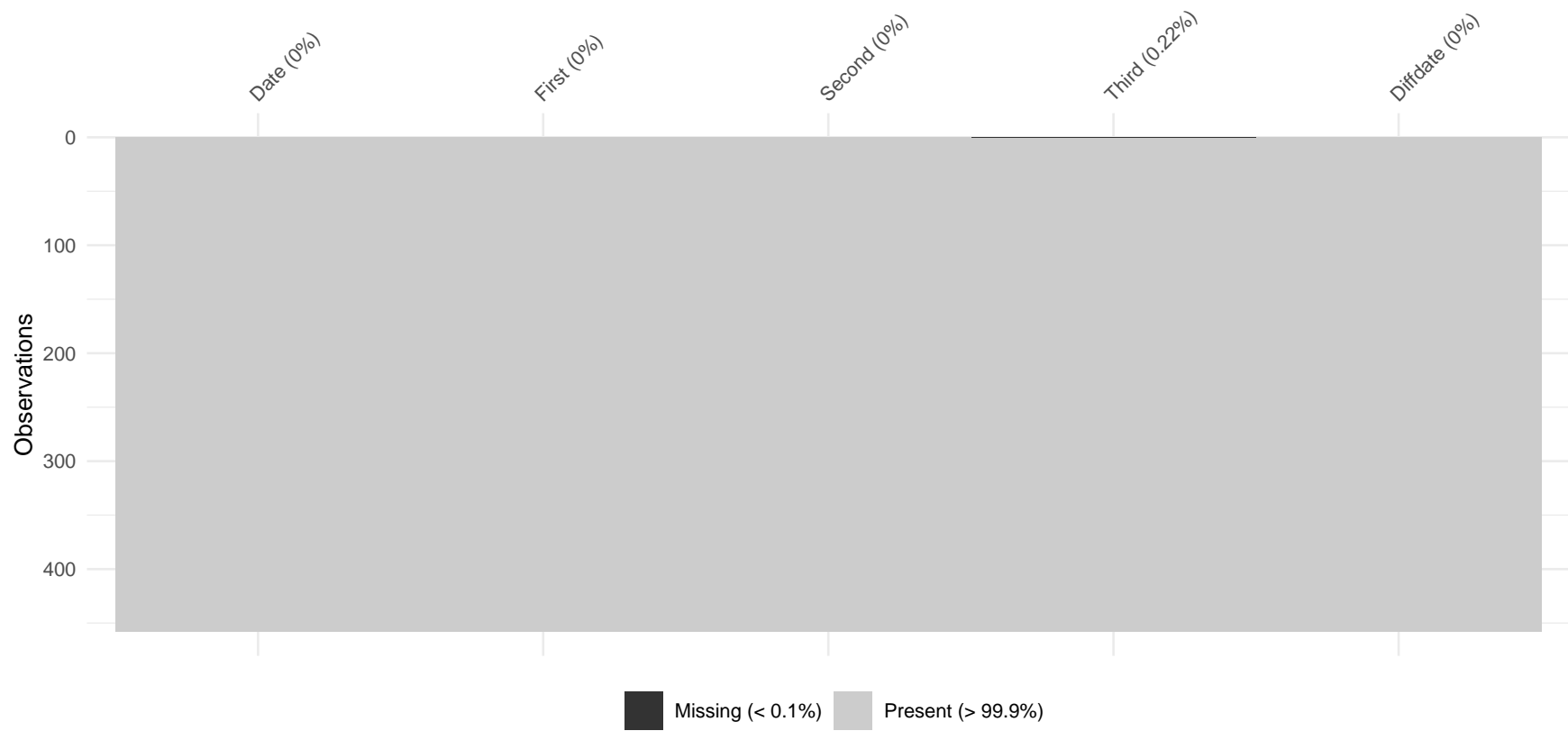
Yes, we have missing values in the column "Third". In this case, missing values and zeroes are equivalent.

Replace by zero.

I have a question.

How many people got their jabs with me?

Vaccination in England

4 Step 4 Machine learning

4.1 South West

4.1.1 Look at and Modify the dataset

So, I am curious. Can I predict vaccination data?

I will work with the South West's vaccination data.

	First	Second	Third
2022-01-26	986	2520	4034
2022-01-25	899	1845	4283
2022-01-24	723	1445	3441
2022-01-23	1035	3007	3439
2022-01-22	1822	4709	5896
2022-01-21	1085	2362	4944
2022-01-20	1152	2330	5058
2022-01-19	1083	2524	5017
2022-01-18	1298	2126	5359
2022-01-17	946	1699	4374
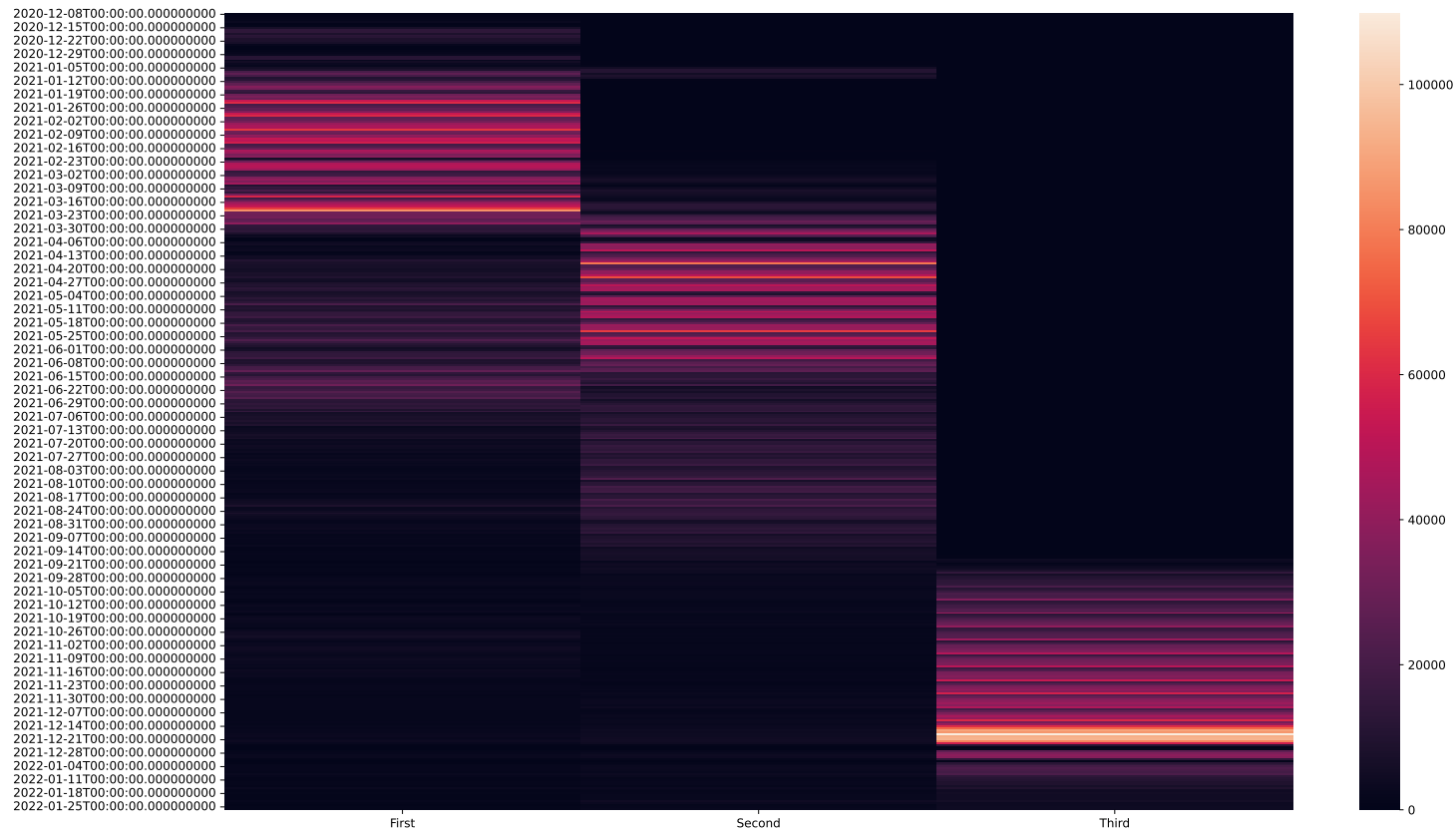
As we discuss earlier 2, there are waves. So, the count of jabs depends on dates.

Let's get features: 1) Year 2) Month 3) Day etc.

	First	Second	Third	Year	Month	Day	DayOfYear	Weekday	Quarter	IsMonthStart	IsMonthEnd
2022-01-26	986	2520	4034	2022	1	26	26	2	1	FALSE	FALSE
2022-01-25	899	1845	4283	2022	1	25	25	1	1	FALSE	FALSE
2022-01-24	723	1445	3441	2022	1	24	24	0	1	FALSE	FALSE
2022-01-23	1035	3007	3439	2022	1	23	23	6	1	FALSE	FALSE
2022-01-22	1822	4709	5896	2022	1	22	22	5	1	FALSE	FALSE
2022-01-21	1085	2362	4944	2022	1	21	21	4	1	FALSE	FALSE
2022-01-20	1152	2330	5058	2022	1	20	20	3	1	FALSE	FALSE
2022-01-19	1083	2524	5017	2022	1	19	19	2	1	FALSE	FALSE
2022-01-18	1298	2126	5359	2022	1	18	18	1	1	FALSE	FALSE
2022-01-17	946	1699	4374	2022	1	17	17	0	1	FALSE	FALSE

First of all, I am going to use Regression Machine Learning models:

- Decision Tree

- Random Forest.

What is my plan?

1. Read data

I already did this step.

2. Understand statistics about the data

It will be helpful to choose the right features for better results.

- Work with missing data and categorical variables
- Work with outliers or not completed data.

5. Store prediction target (y) in a Series, selecting multiple features by providing a list of column names inside brackets, define X (subset with features), check the X summary.

6. Choose the library

7. Build and use the model What type of model will it be? Capture patterns from provided data. Predict Evaluate = Determine how accurate the model's predictions are

Let's look at the dataset carefully.

4.1.2 Explore the dataset

In the previous chapter 3.1, we already looked at the South West's data. Do we need to know something else? Yes.

4.1.2.1 Data types It is important to know which types of data columns have. Sometimes we don't realise what we see: the string or the number.

First	double
Second	double
Third	double
Year	double
Month	double
Day	double
DayOfYear	double
Weekday	double
Quarter	double
IsMonthStart	logical
IsMonthEnd	logical

The good news is I don't need to convert my variables because they fit into Regression Machine Learning models.

We will move on to correlations.

4.1.2.2 Correlations What do we need to remember? Correlation does not imply causation. So, the columns that have a strong relationship may show low accuracy in the model.
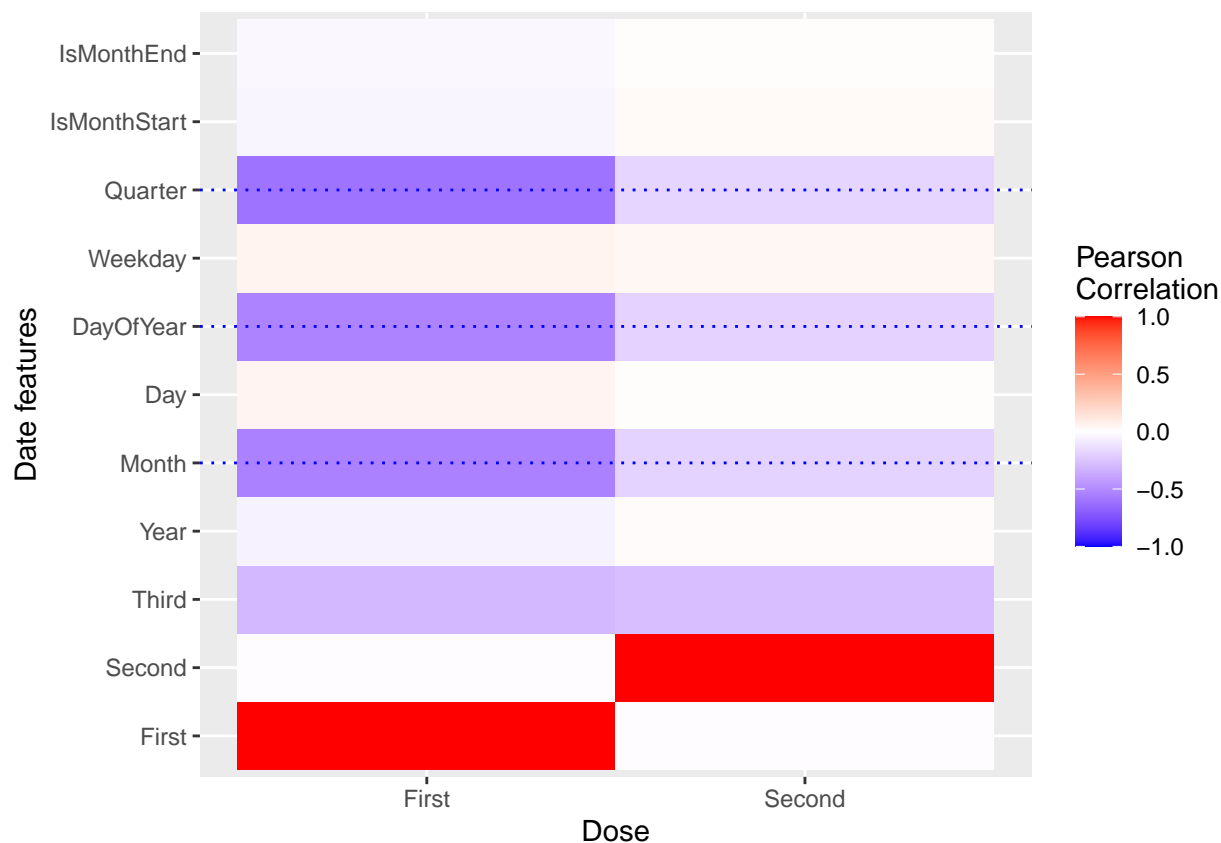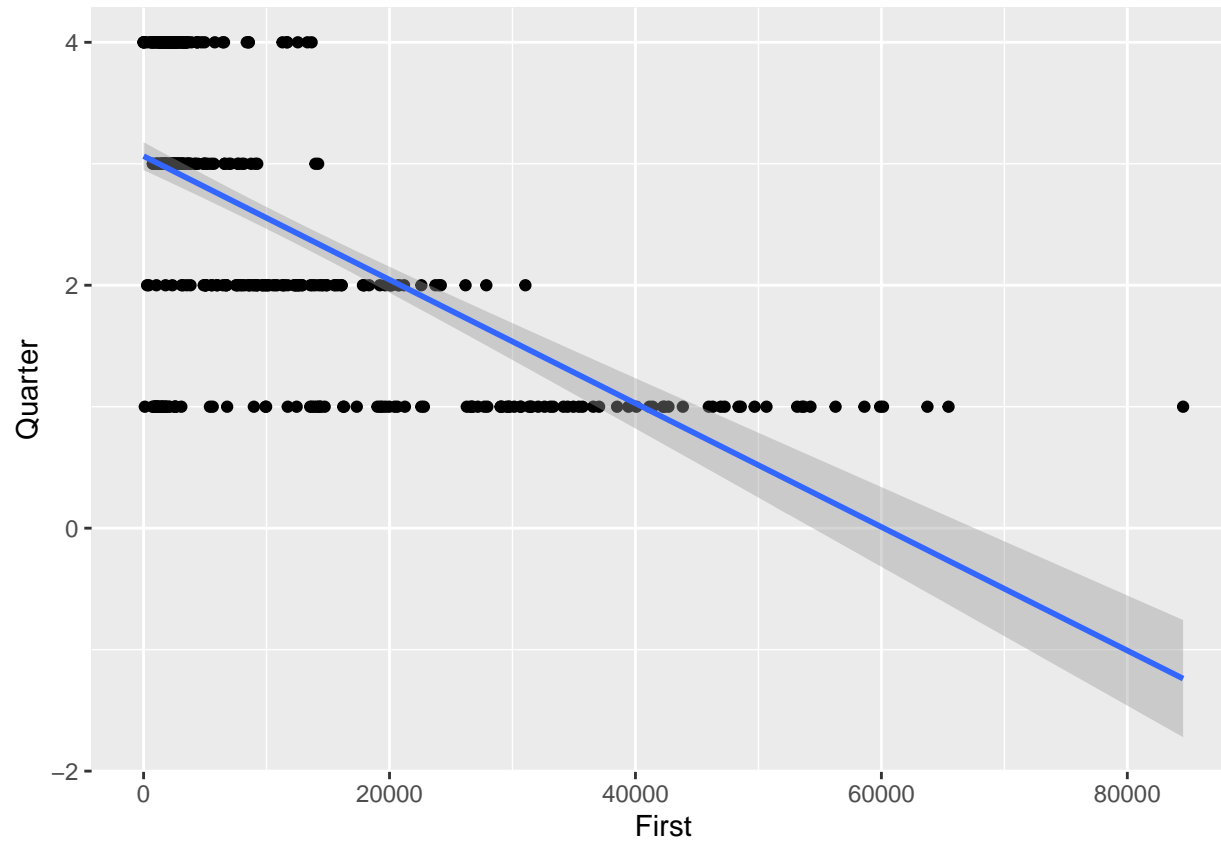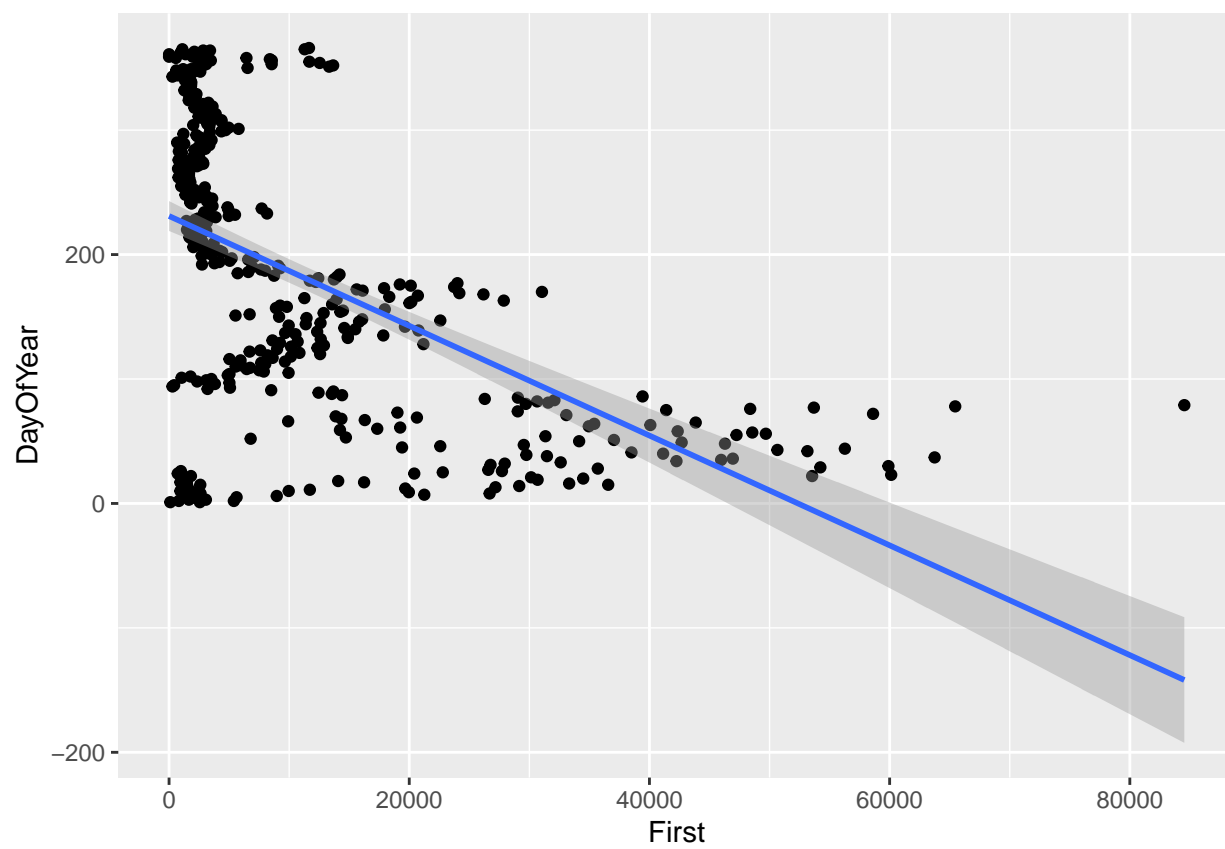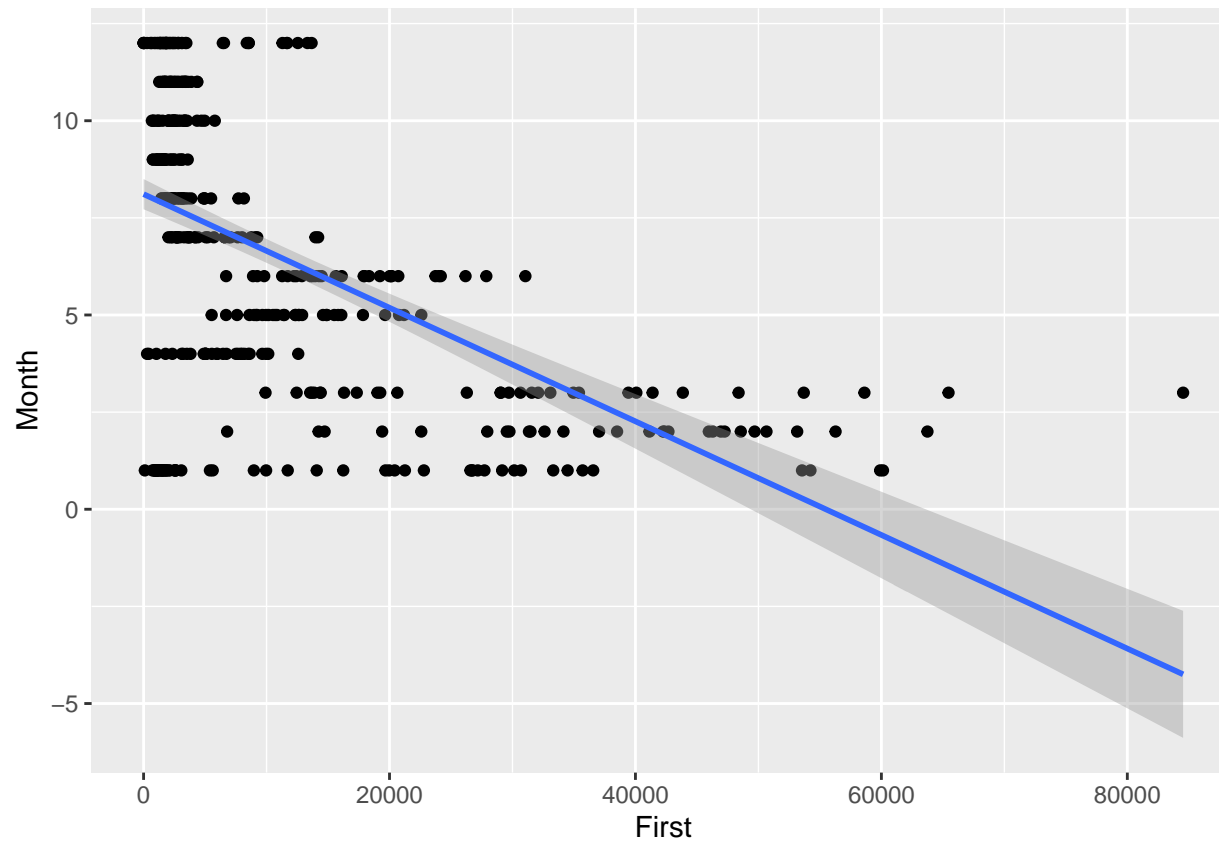

In the table below, we can see the numeric values.

First	Month	-0.5432969
Second	Month	-0.1888013
First	DayOfYear	-0.5343244
Second	DayOfYear	-0.1901012
First	Quarter	-0.6070344
Second	Quarter	-0.1799906

As we can see, the column "First" has a strong relationship with

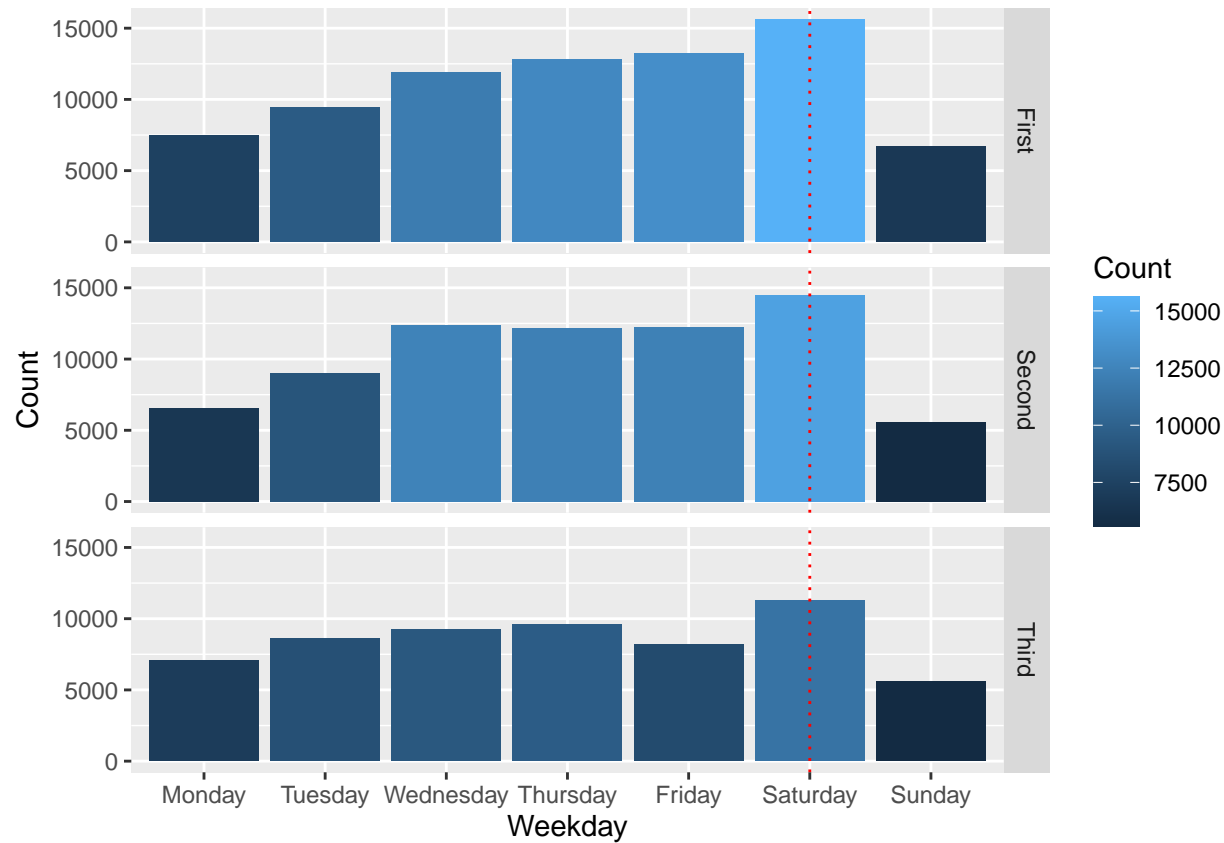- "Quarter",

- "DayOfYear",

- "Month".

At the same time, the column "Second" doesn't have strong relationships; but we can use the same columns.

4.1.2.3   Weekdays   As you remember, I have a question. When do people prefer to get a jab: weekdays or weekends/Saturdays or Sundays? It may be helpful to choose the right features.

Let's answer.

So, most of South West's people prefer to get a jab on Saturdays. That is not illogical because, for example, for me, the side effects go away during the weekend.

4.1.2.4 Missing values As we already saw in the previous chapter 3.1, the column "Third" has missing values, but we can replace them with zeroes. Do we have the dates when nobody got the jab?

Calculate a count of dates in the dataset.

```
## 415
```

Calculate a count of dates between maximum and minimum dates.

```
## 415
```

There are no missing dates.

So, we have finished the dataset exploring. The next steps are about the models.

4.1.3   Split sets, train a Machine Learning Model and Evaluate performance

Define necessary variables

First of all, I will use all columns that I have.

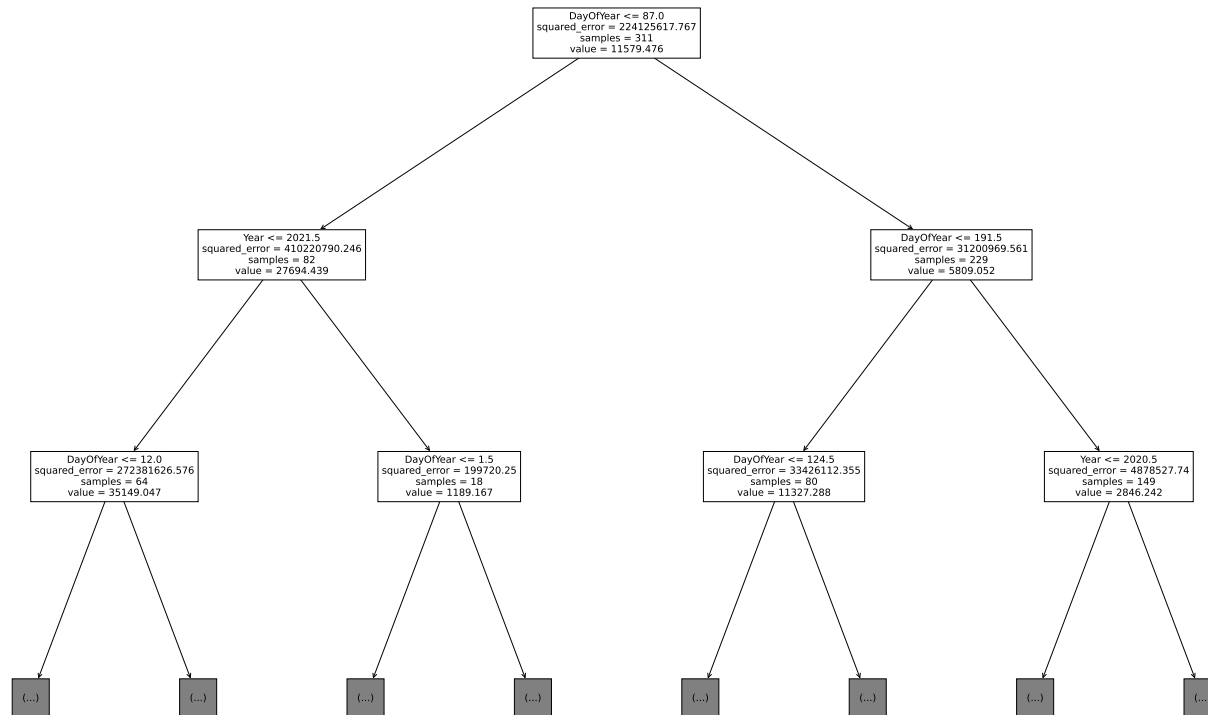| |
| --- |
| Year |
| Month |
| Day |
| DayOfYear |
| Weekday |
| Quarter |
| IsMonthStart |
| IsMonthEnd |

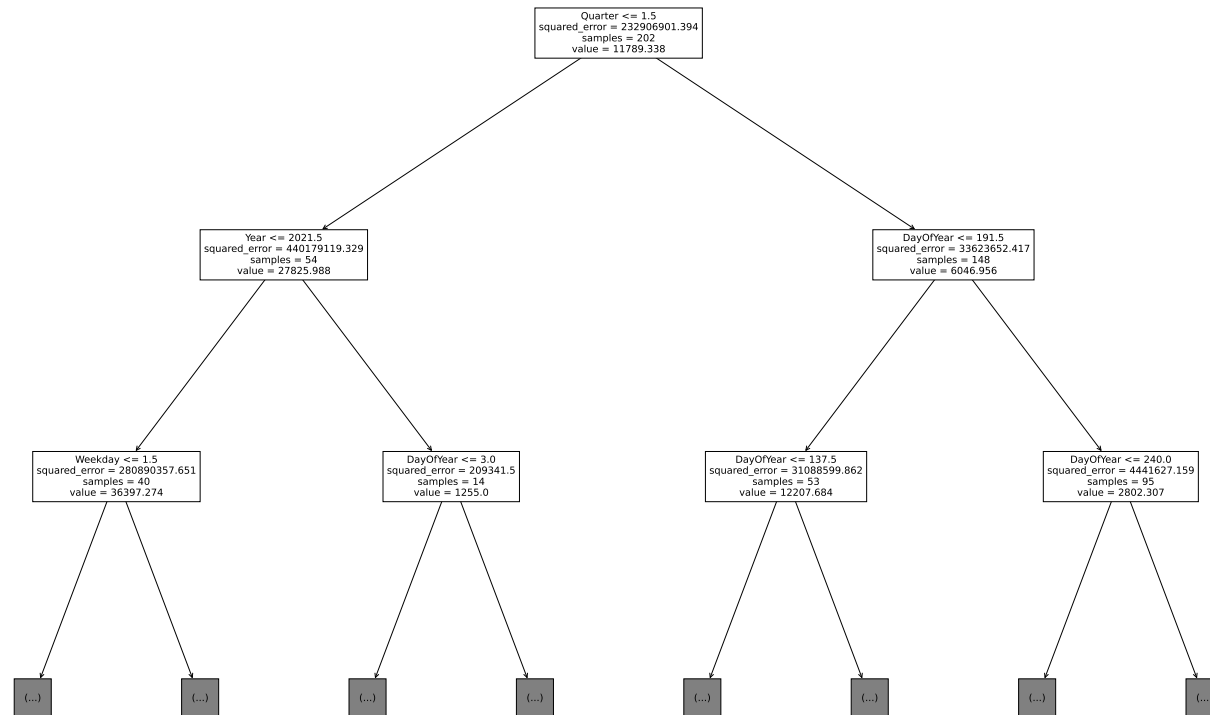**Prepare sets and train models using parameters.**

The first column that I will predict is "First".

```
## DecisionTree:   0.719657929335243
```
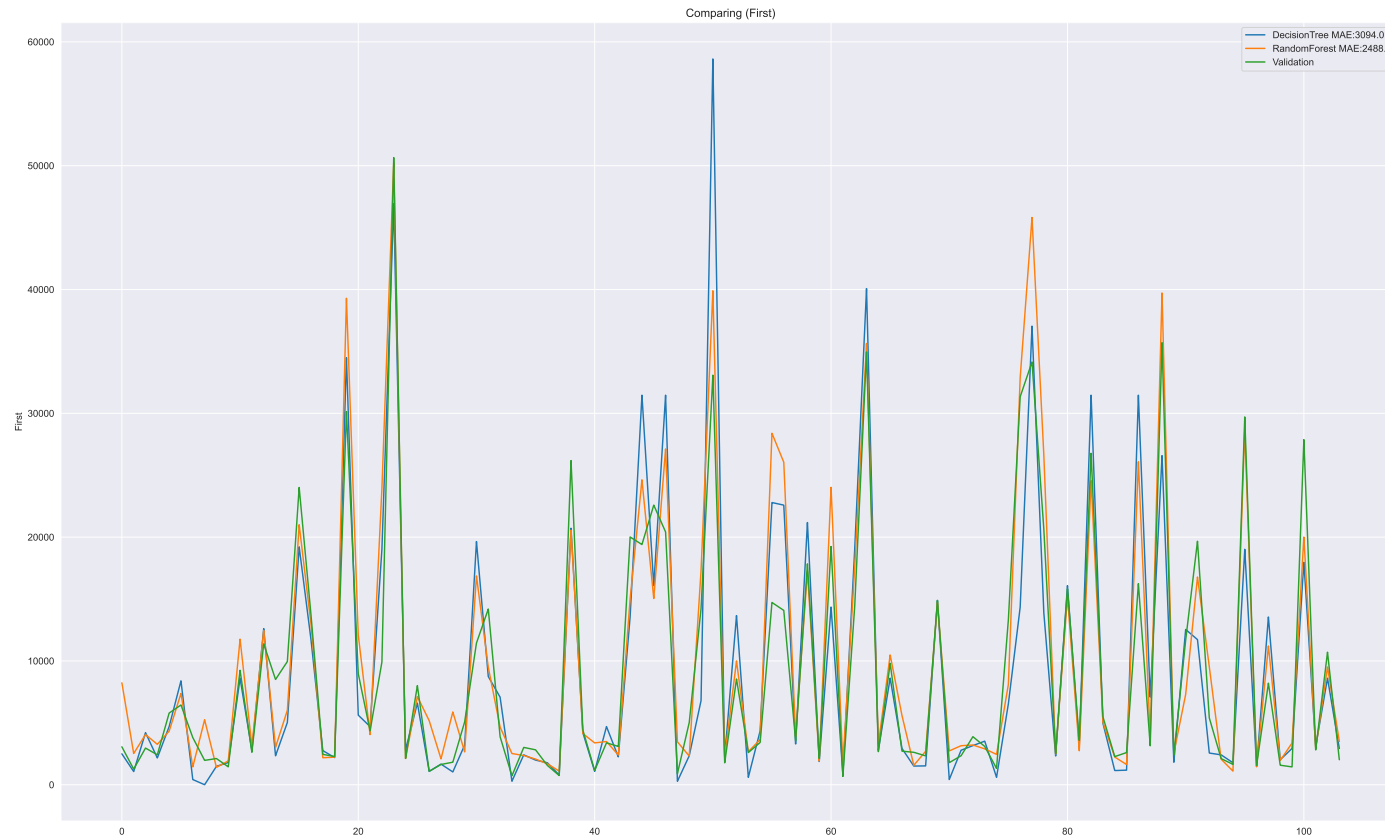
```
## RandomForest:   0.774580856609961
```

Look at the tree

DayOfYear <= 87.0
squared_error = 224125617.767
samples = 311
value = 11579.476

Year <= 2021.5
squared_error = 410220790.246
samples = 82
value = 27694.439

DayOfYear <= 191.5
squared_error = 31200969.561
samples = 229
value = 5809.052

DayOfYear <= 12.0
squared_error = 272381626.576
samples = 64
value = 35149.047

DayOfYear <= 1.5
squared_error = 199720.25
samples = 18
value = 1189.167

DayOfYear <= 124.5
squared_error = 33426112.355
samples = 80
value = 11327.288

Year <= 2020.5
squared_error = 4878527.74
samples = 149
value = 2846.242

(...) (...) (...) (...) (...) (...) (...) (...)

```
                        Quarter <= 1.5
                  squared_error = 232906901.394
                          samples = 202
                         value = 11789.338
```

```
        Year <= 2021.5                              DayOfYear <= 191.5
 squared_error = 440179119.329            squared_error = 33623652.417
        samples = 54                              samples = 148
      value = 27825.988                          value = 6046.956
```

```
  Weekday <= 1.5              DayOfYear <= 3.0        DayOfYear <= 137.5          DayOfYear <= 240.0
squared_error = 280890357.651   squared_error = 209341.5   squared_error = 31088599.862   squared_error = 4441627.159
  samples = 40                samples = 14              samples = 53               samples = 95
 value = 36397.274           value = 1255.0          value = 12207.684           value = 2802.307
```

(...) (...)   (...) (...)   (...) (...)   (...) (...)

What can we see?

38

Finally, look at the result.

Comparing (First)

In my opinion, the result is good.

- The waves were recognized.

- The extreme values are bigger than in real data.

Let's work with the columns that I chose during the dataset exploring.

- "Weekday" that we discussed in this chapter influences the wave during the week.

- "Year" is the logical key because of the vaccination steps.

- DayOfYear was chosen because of the dependency on dates.

```
## DecisionTree:  0.7248630326024768
```

```
## RandomForest:  0.7837038702898657
```
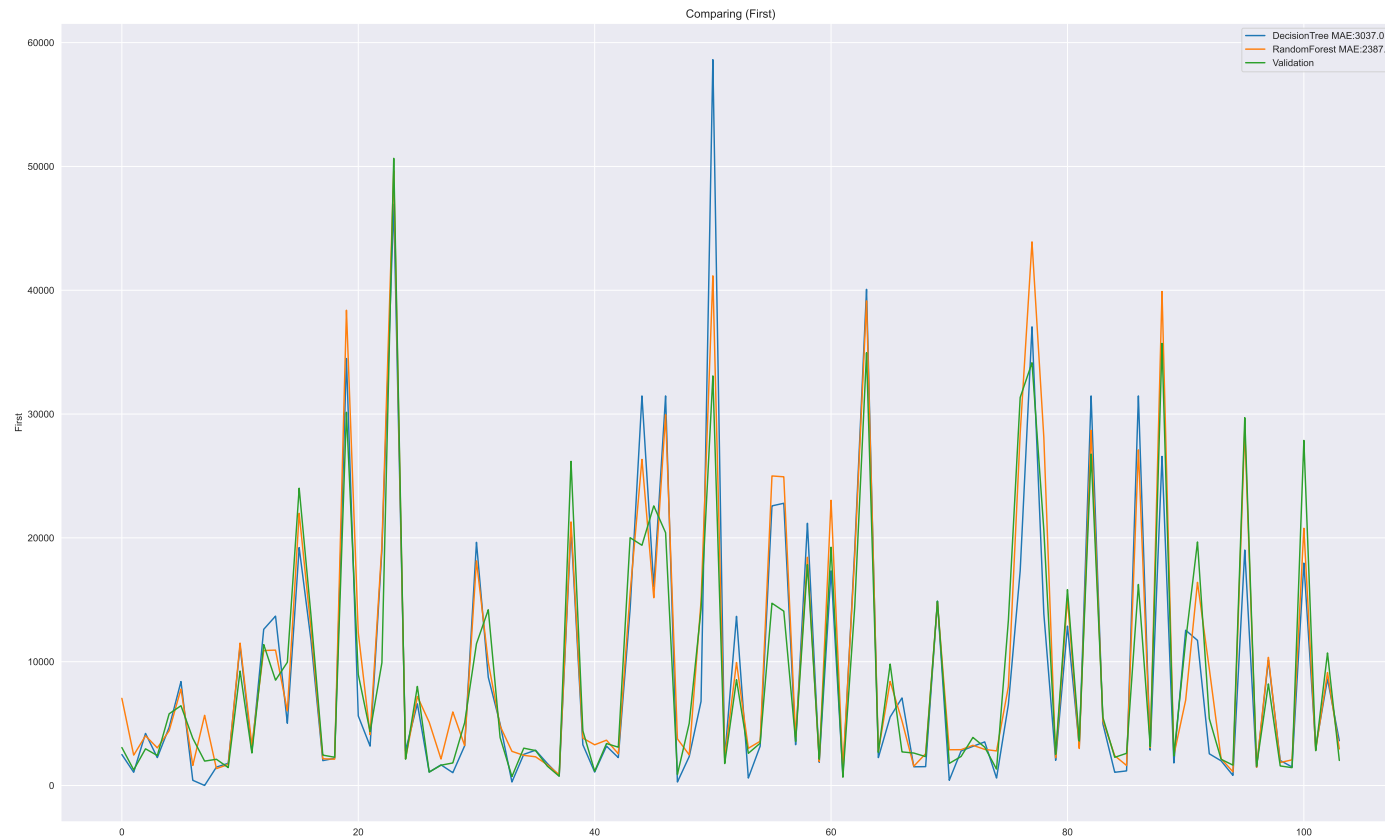
The result is better a little, but extreme values are disappointing.

Also, I suggest checking the model with columns that we discussed during the correlations search.

```
## DecisionTree:   0.338881322155111
```

```
## RandomForest:   0.47495276411406473
```

Comparing (First)
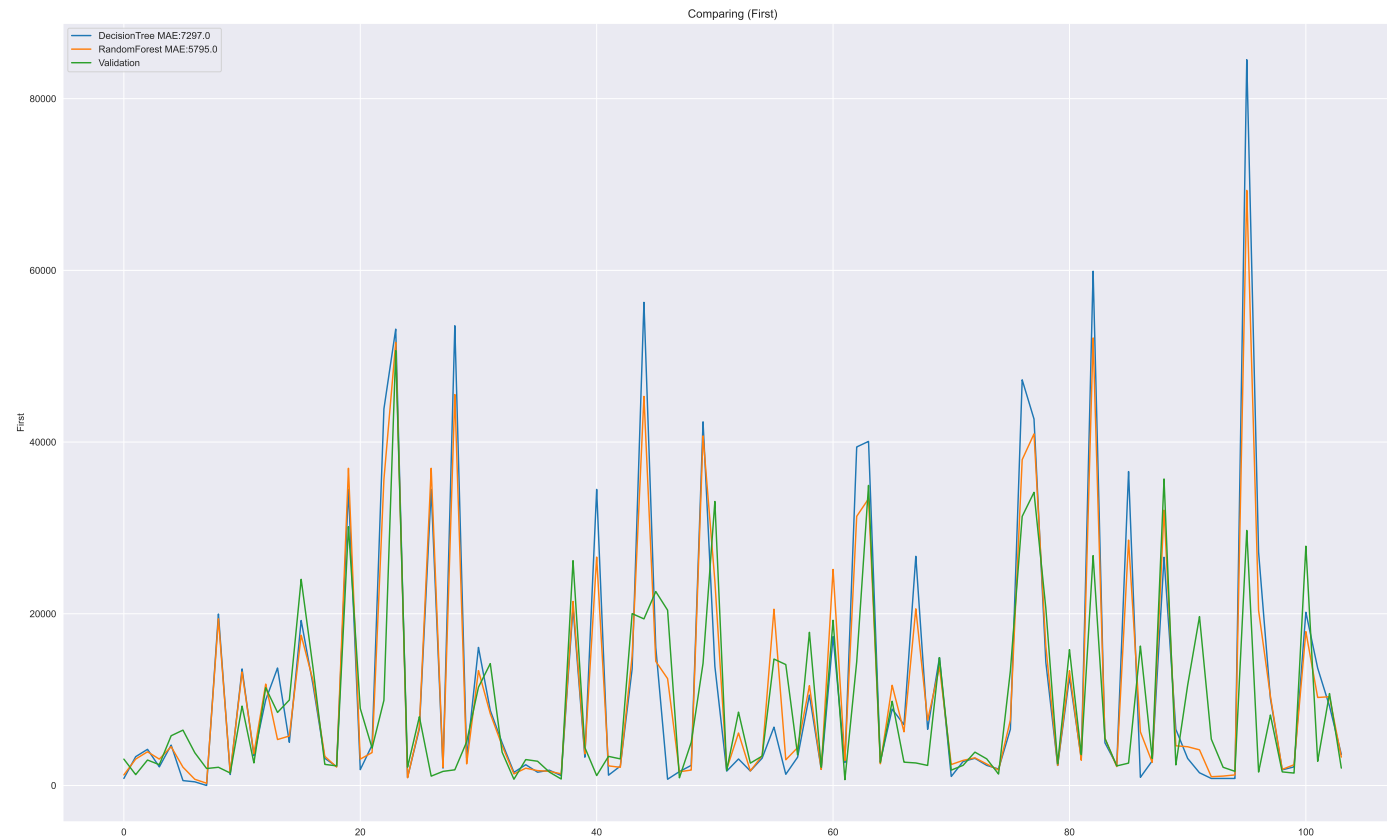
Not so good.

Repeat for the Second

```
## DecisionTree:  0.7445280765098062

## RandomForest:  0.8144473412230163
```

Comparing (Second)
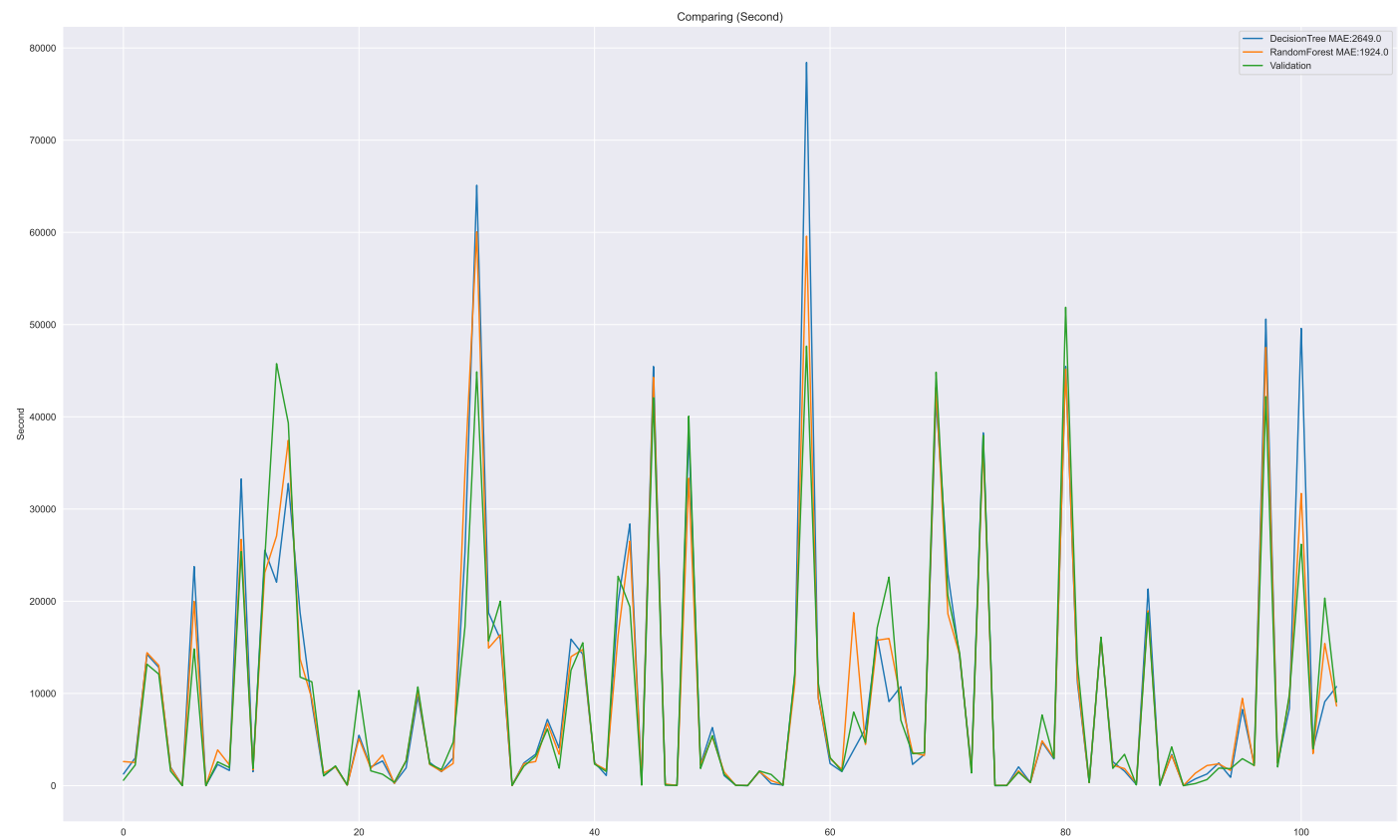
DecisionTree: 0.7692636418171874

RandomForest: 0.8318561653086721

Comparing (Second)
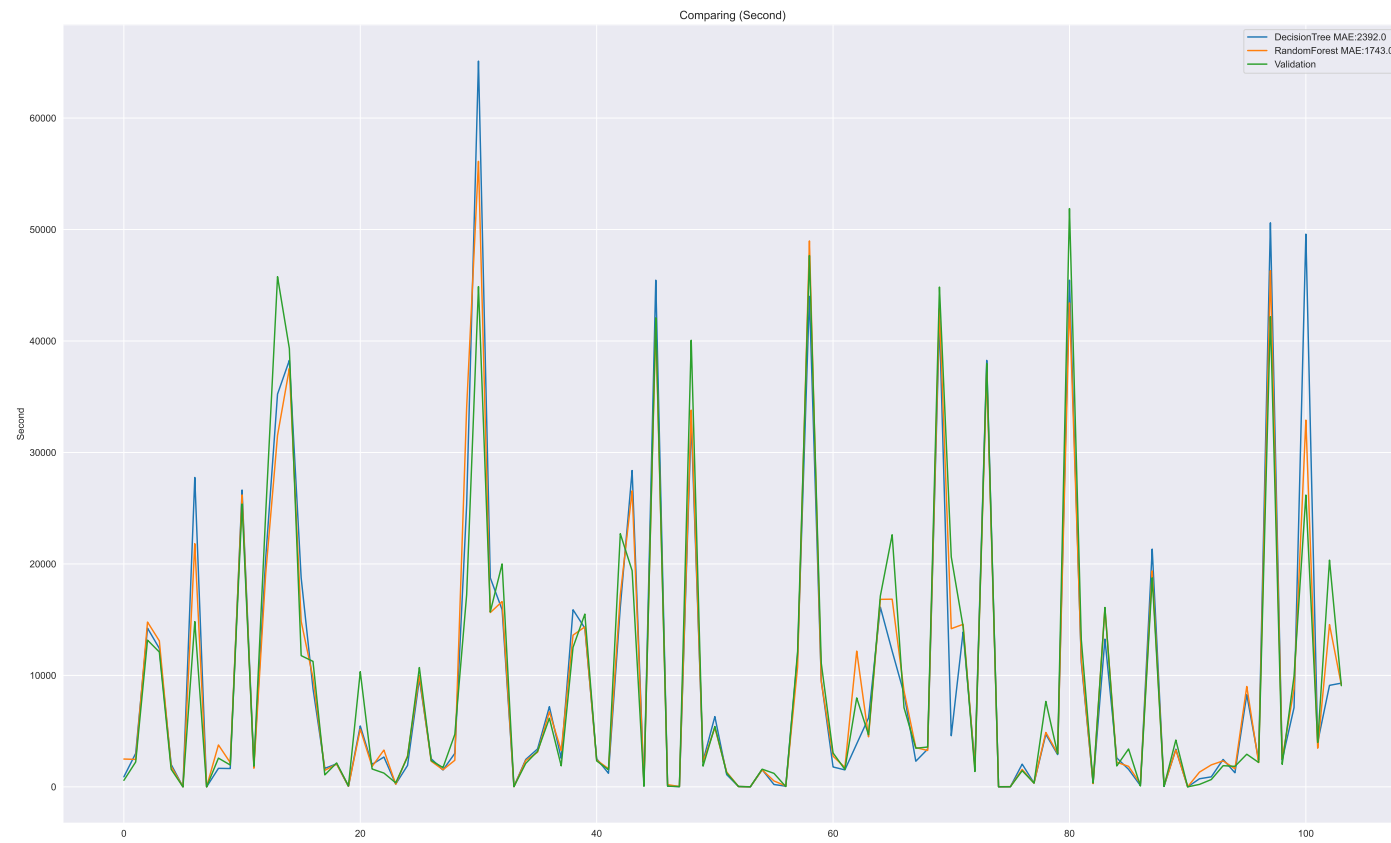
DecisionTree: 0.6985989452917025

RandomForest: 0.7239113203537064

Comparing (Second)
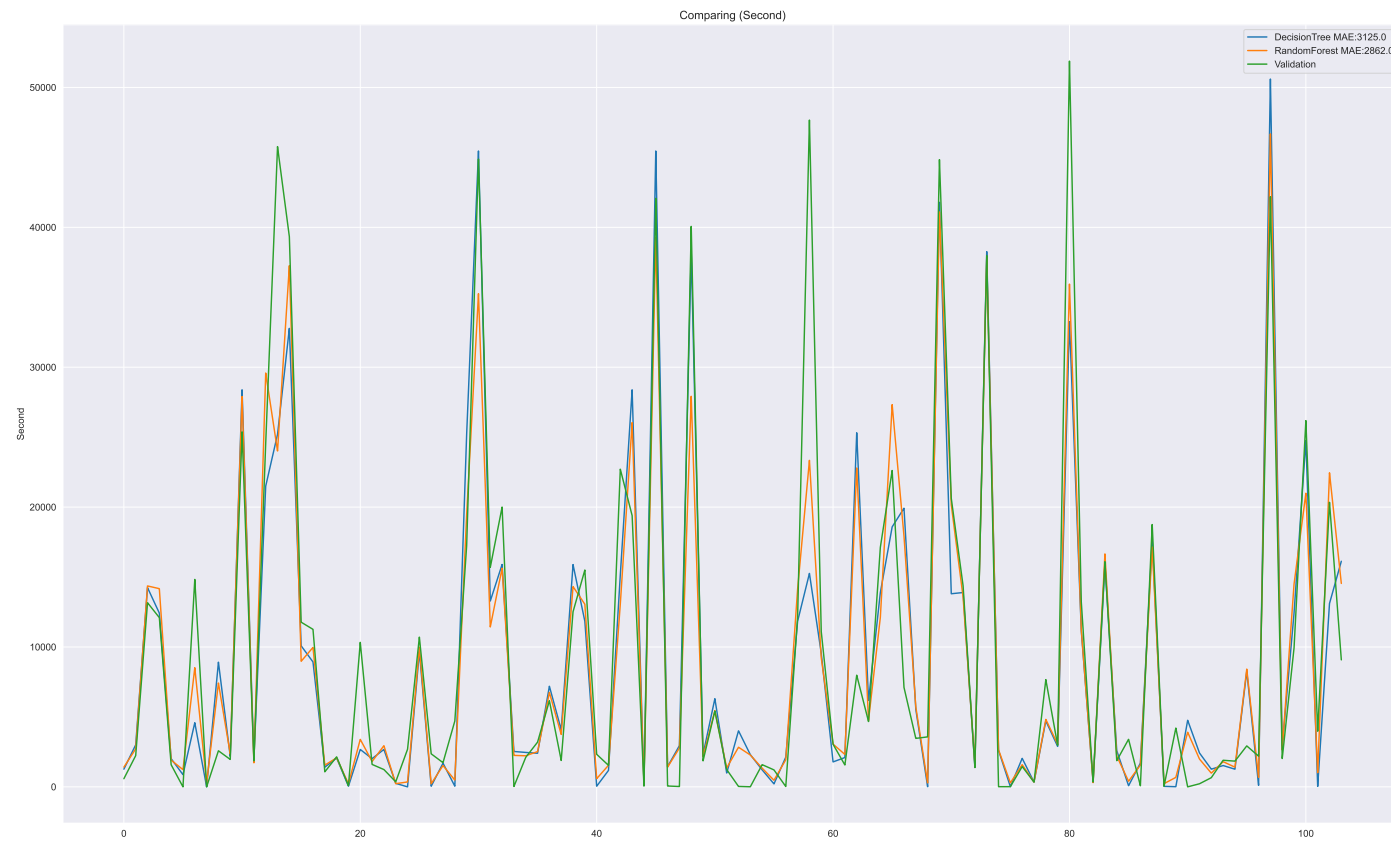
Repeat for Third

DecisionTree: 0.8323340834065006

RandomForest: 0.8304365732995669
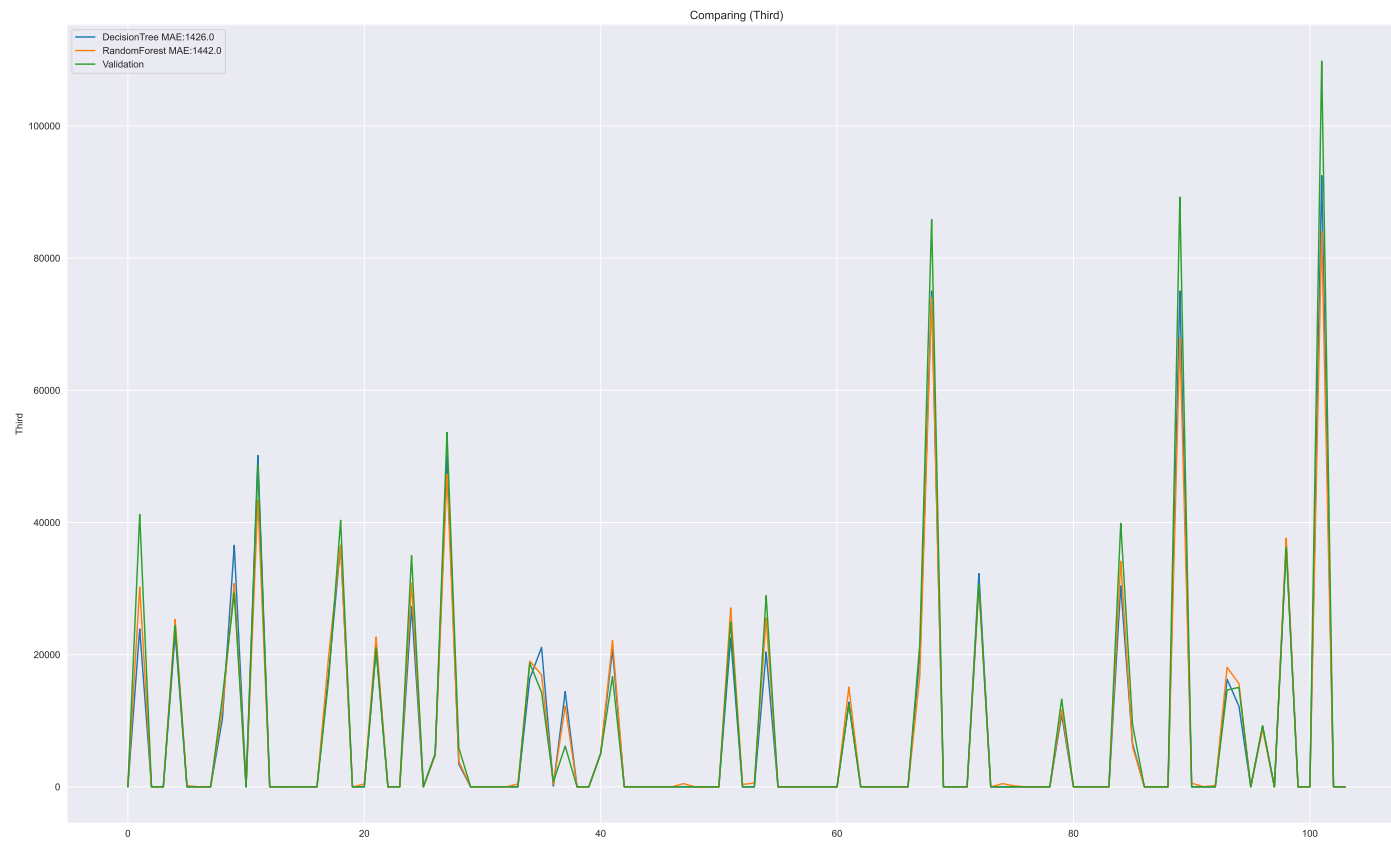
DecisionTree: 0.8664423499345815

RandomForest: 0.840745098066024

Comparing (Third)
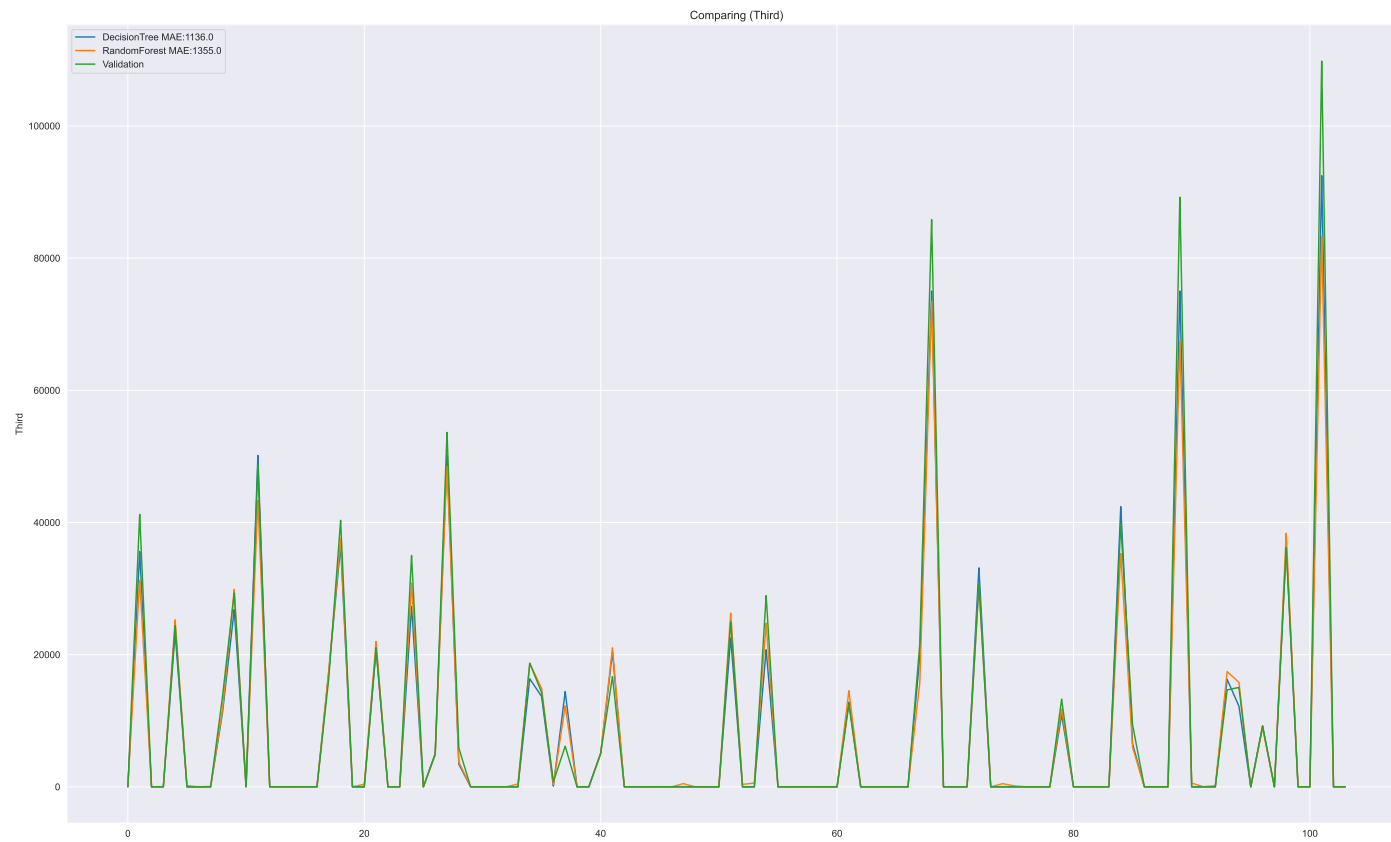
DecisionTree: -0.11041243558502623

RandomForest: 0.028430358547872348

Comparing (Third)
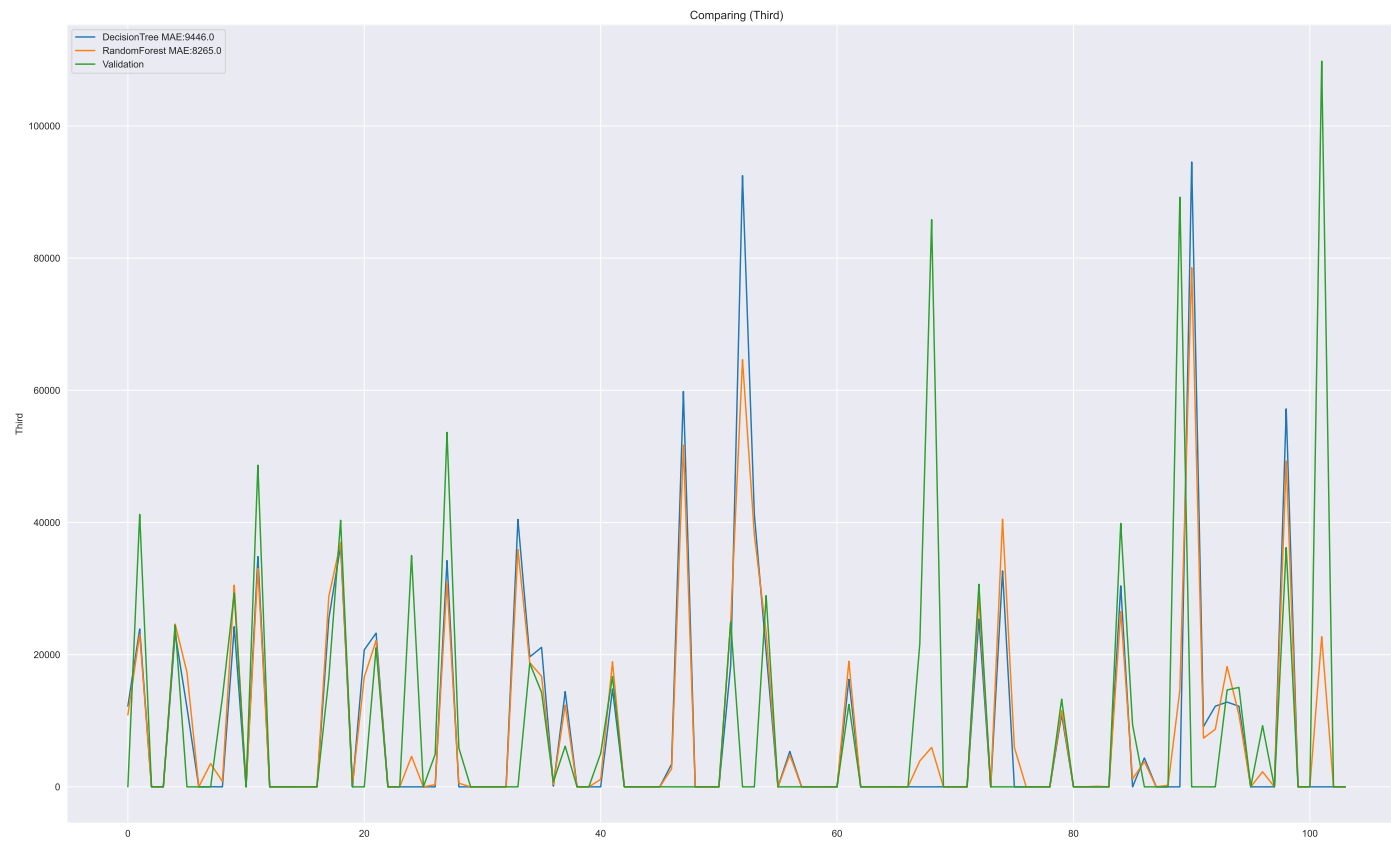
A combination of the following features give us the best result: Weekday, Year, DayOfYear.

4.2 Bristol

Look at the dataset.

age	date	First	Second	Third	Age
12_15	2022-02-09	28	45	0	1
16_17	2022-02-09	4	8	6	2
18_24	2022-02-09	25	18	83	3
25_29	2022-02-09	11	8	48	4
30_34	2022-02-09	4	10	37	5
35_39	2022-02-09	3	8	26	6

Using the k-means clustering model, we find out that sometimes a lot of people in the same age group got their jabs together. And the group 18-24 is not an exception.
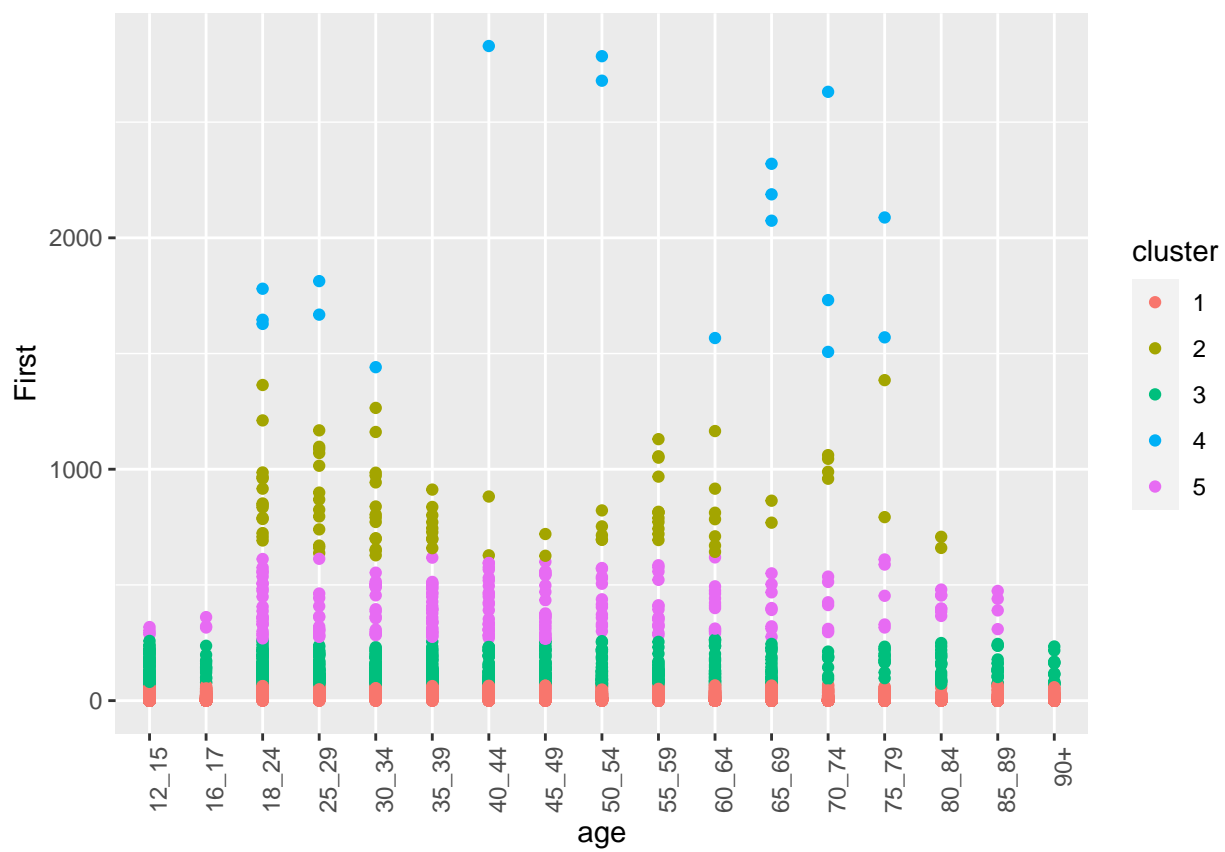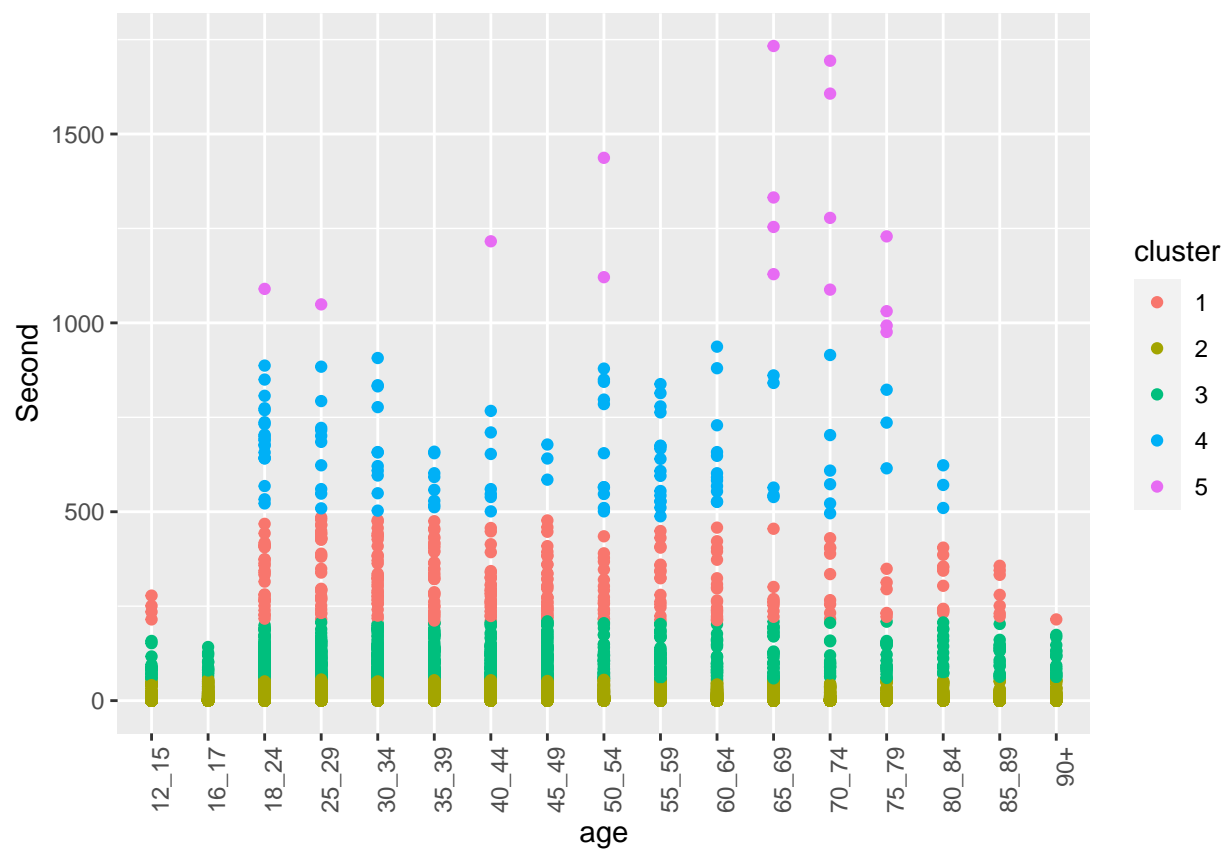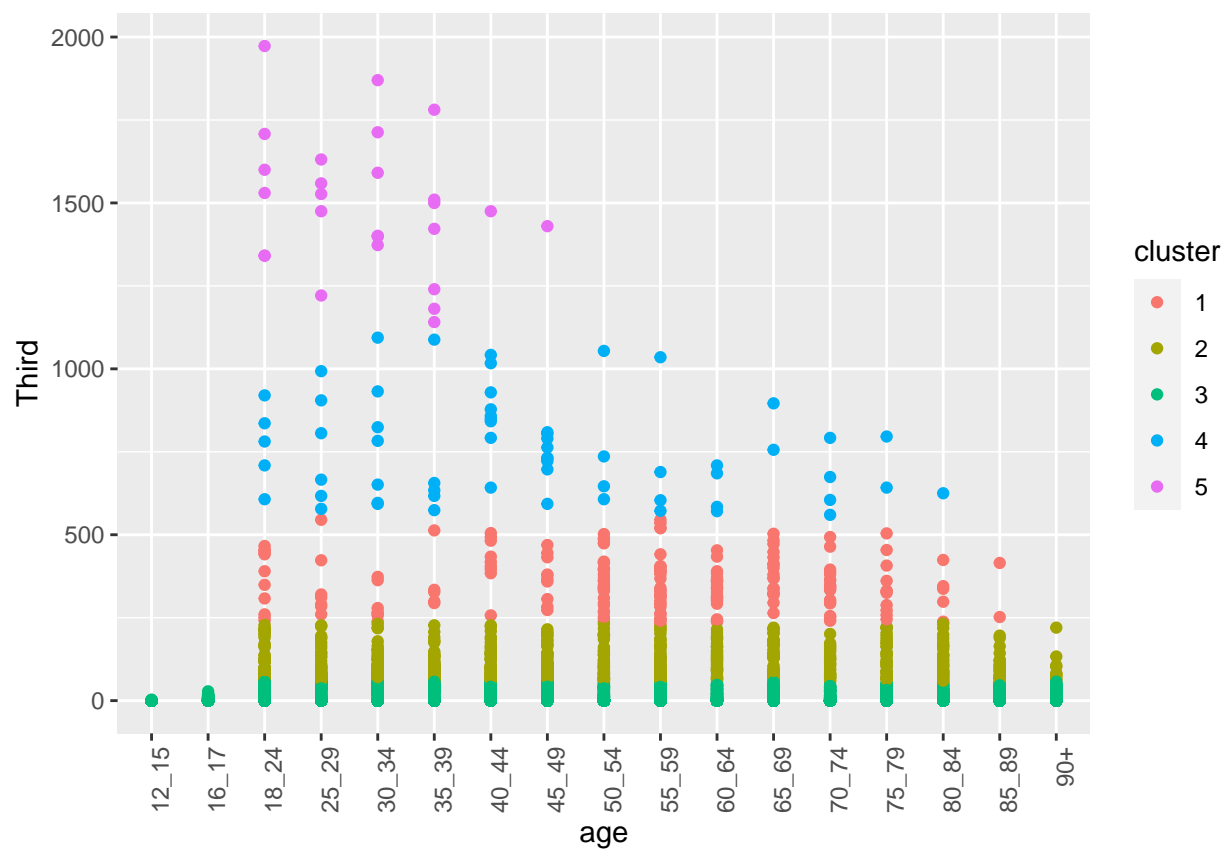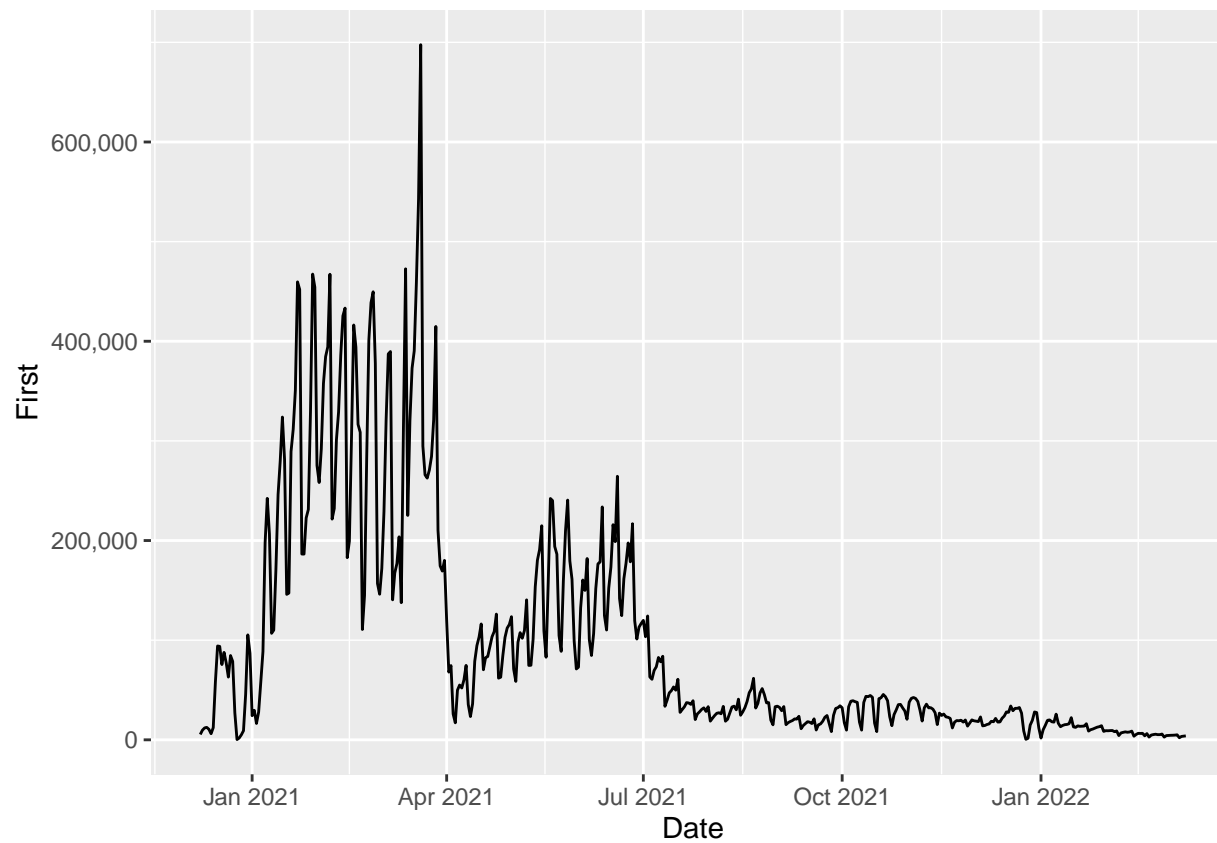
4.3 Model for England

Look at the dataset.

	Date	First	Second	Third
457	2020-12-08	5370	146	0
456	2020-12-09	9648	137	2
455	2020-12-10	11888	119	1
454	2020-12-11	12516	82	1
453	2020-12-12	10565	23	3
452	2020-12-13	6134	30	0

What can we say?

There is a not stationary time series, as the series wanders up and down for long periods.

```
## Series: englandFit
## ARIMA(5,1,3)
##
## Coefficients:
##           ar1      ar2      ar3      ar4      ar5      ma1     ma2      ma3
##        0.1844  -0.8187  -0.1423  -0.2771  -0.5162  -0.4280  0.7674  -0.1984
## s.e.   0.0609   0.0472   0.0701   0.0406   0.0472   0.0646  0.0465   0.0534
##
## sigma^2 = 1.382e+09:  log likelihood = -5443.42
## AIC=10904.84   AICc=10905.24   BIC=10941.94
##
## Training set error measures:
##                     ME      RMSE      MAE       MPE      MAPE      MASE
## Training set -18.47453 36805.03 19210.84 -40.98583 58.06987 0.7904012
##                    ACF1
## Training set -0.03255101
```

ARIMA(5, 1, 3) ARIMA(p,d,q)

- p - is the order of Auto-regressive or linear model
- q – is the order of Moving Average/ number of lagged values
- d- difference value to make the time series stationary from non-stationary. If the data is stationary, then d=0. So, I was right earlier.

First
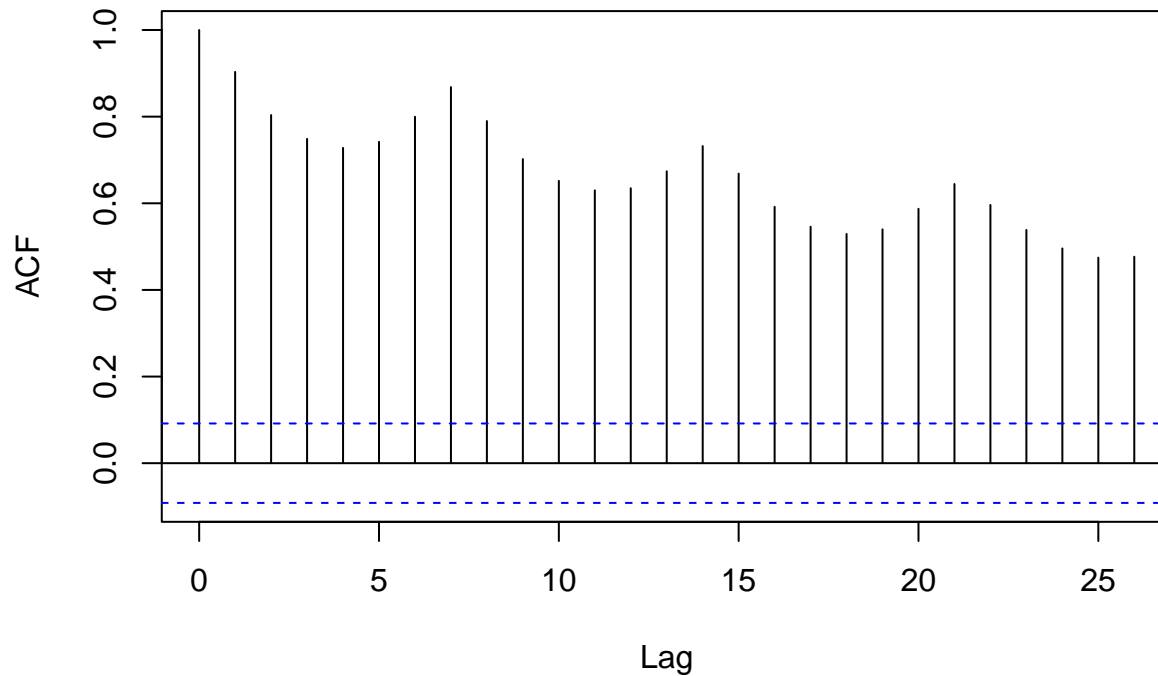
As we know, the autocorrelation function (ACF) assesses the correlation between observations in a time series for a set of lags. In an ACF plot, each bar represents the size and direction of the correlation. Bars that extend across the blue line are statistically significant.

So,

- this ACF plot indicates that these time series data are not random.
- the autocorrelations decline slowly.
- When a time series has both a trend and seasonality, the ACF plot displays a mixture of both effects. Notice how you can see the wavy correlations for the seasonal pattern and the slowly diminishing lags of a trend.

The residuals in ARIMA models tell a story about the performance of your model and should be taken into consideration when evaluating them. The functions checkresiduals, ACF and PACF make it easy to keep track of the information left behind in the residuals by your model. https://towardsdatascience.com/time-series-analysis-with-auto-arima-in-r-2b220b20e8ab

Residuals from ARIMA(5,1,3)
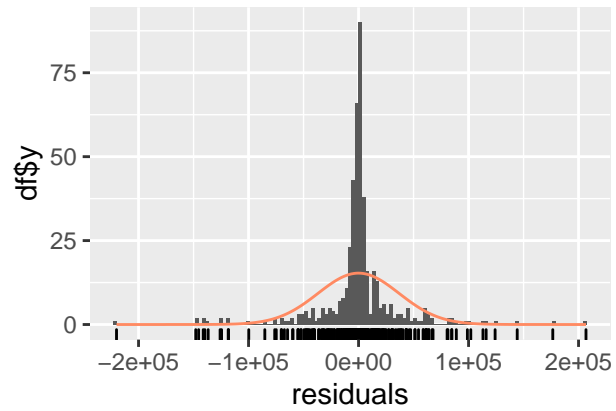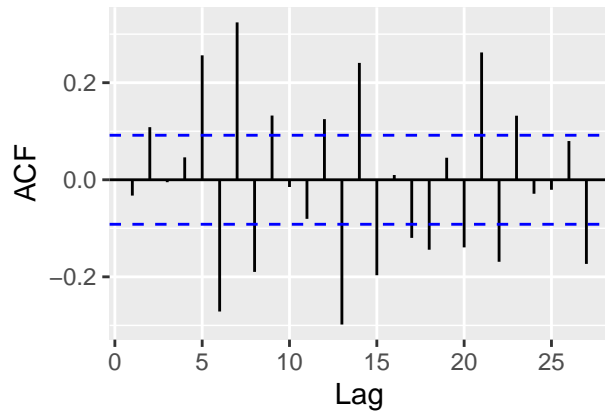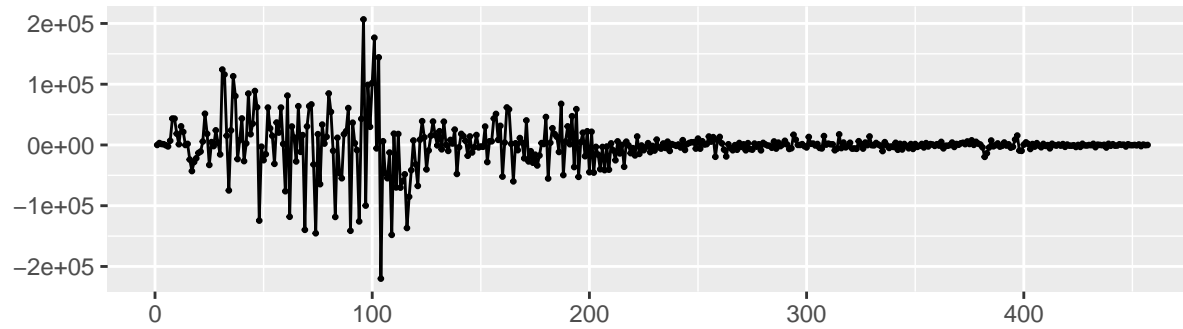

```
## 
##  Ljung-Box test
## 
## data:  Residuals from ARIMA(5,1,3)
## Q* = 148.81, df = 3, p-value < 2.2e-16
## 
## Model df: 8.    Total lags used: 11
```

Forecasts from ARIMA(5,1,3)
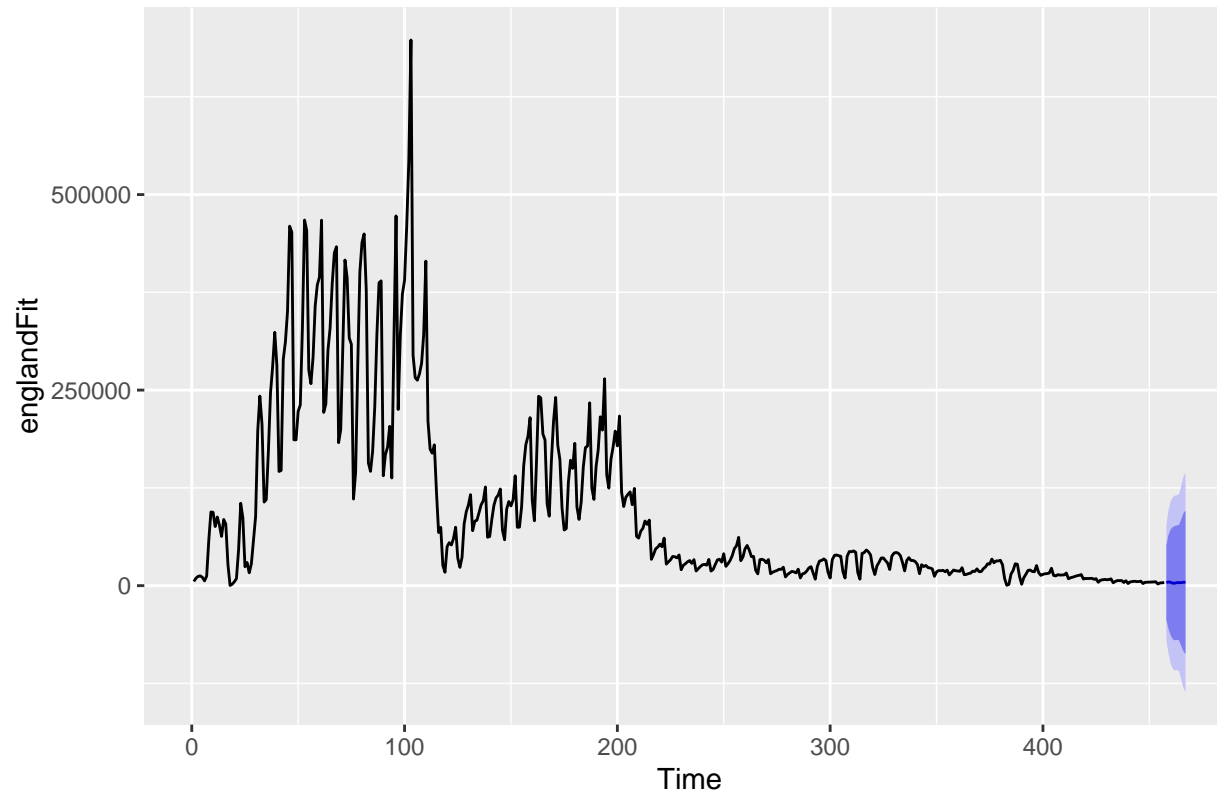

```
forecast(fit)$lower
```

```
## Time Series:
## Start = 458
## End = 467
## Frequency = 1
##             80%        95%
## 458 -43782.97  -69001.52
## 459 -54927.77  -86548.71
## 460 -63249.94  -98986.75
## 461 -68432.48 -106338.20
```

```
## 462 -69923.65 -108617.74
## 463 -69190.35 -107892.99
## 464 -70108.61 -109273.86
## 465 -77076.64 -119910.27
## 466 -83815.86 -130472.92
## 467 -87430.74 -135959.13
```

```
forecast(fit)$upper[1,]
```

```
##      80%      95%
## 51494.96 76713.51
```