Contents

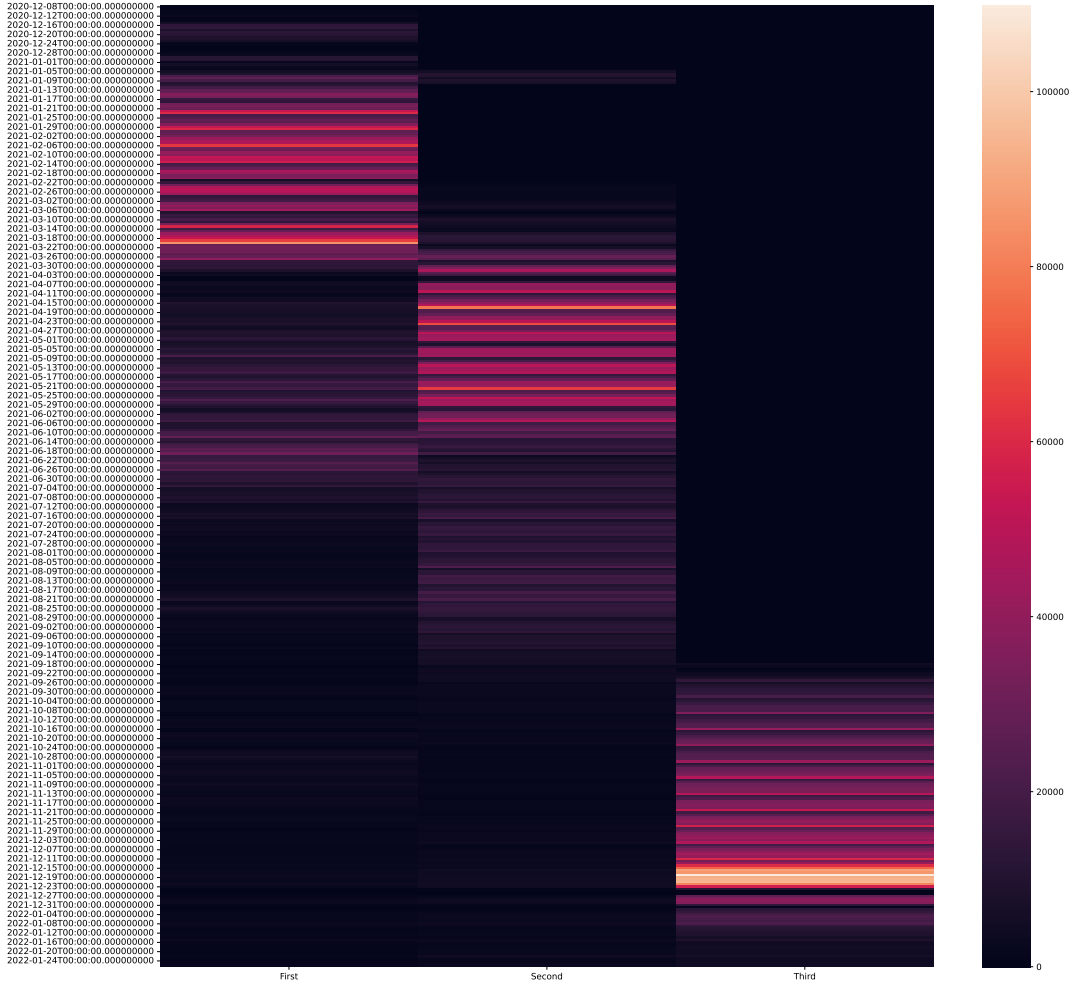
1 Step 4 Machine learning

1.1 Step 0: 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


2020-12-08T00:00:00.000000000
2020-12-12T00:00:00.000000000
2020-12-16T00:00:00.000000000
2020-12-20T00:00:00.000000000
2020-12-24T00:00:00.000000000
2020-12-28T00:00:00.000000000
2021-01-01T00:00:00.000000000
2021-01-05T00:00:00.000000000
2021-01-09T00:00:00.000000000
2021-01-13T00:00:00.000000000
2021-01-17T00:00:00.000000000
2021-01-21T00:00:00.000000000
2021-01-25T00:00:00.000000000
2021-01-29T00:00:00.000000000
2021-02-02T00:00:00.000000000
2021-02-06T00:00:00.000000000
2021-02-10T00:00:00.000000000
2021-02-14T00:00:00.000000000
2021-02-18T00:00:00.000000000
2021-02-22T00:00:00.000000000
2021-02-26T00:00:00.000000000
2021-03-02T00:00:00.000000000
2021-03-06T00:00:00.000000000
2021-03-10T00:00:00.000000000
2021-03-14T00:00:00.000000000
2021-03-18T00:00:00.000000000
2021-03-22T00:00:00.000000000
2021-03-26T00:00:00.000000000
2021-03-30T00:00:00.000000000
2021-04-03T00:00:00.000000000
2021-04-07T00:00:00.000000000
2021-04-11T00:00:00.000000000
2021-04-15T00:00:00.000000000
2021-04-19T00:00:00.000000000
2021-04-23T00:00:00.000000000
2021-04-27T00:00:00.000000000
2021-05-01T00:00:00.000000000
2021-05-05T00:00:00.000000000
2021-05-09T00:00:00.000000000
2021-05-13T00:00:00.000000000
2021-05-17T00:00:00.000000000
2021-05-21T00:00:00.000000000
2021-05-25T00:00:00.000000000
2021-05-29T00:00:00.000000000
2021-06-02T00:00:00.000000000
2021-06-06T00:00:00.000000000
2021-06-10T00:00:00.000000000
2021-06-14T00:00:00.000000000
2021-06-18T00:00:00.000000000
2021-06-22T00:00:00.000000000
2021-06-26T00:00:00.000000000
2021-06-30T00:00:00.000000000
2021-07-04T00:00:00.000000000
2021-07-08T00:00:00.000000000
2021-07-12T00:00:00.000000000
2021-07-16T00:00:00.000000000
2021-07-20T00:00:00.000000000
2021-07-24T00:00:00.000000000
2021-07-28T00:00:00.000000000
2021-08-01T00:00:00.000000000
2021-08-05T00:00:00.000000000
2021-08-09T00:00:00.000000000
2021-08-13T00:00:00.000000000
2021-08-17T00:00:00.000000000
2021-08-21T00:00:00.000000000
2021-08-25T00:00:00.000000000
2021-08-29T00:00:00.000000000
2021-09-02T00:00:00.000000000
2021-09-06T00:00:00.000000000
2021-09-10T00:00:00.000000000
2021-09-14T00:00:00.000000000
2021-09-18T00:00:00.000000000
2021-09-22T00:00:00.000000000
2021-09-26T00:00:00.000000000
2021-09-30T00:00:00.000000000
2021-10-04T00:00:00.000000000
2021-10-08T00:00:00.000000000
2021-10-12T00:00:00.000000000
2021-10-16T00:00:00.000000000
2021-10-20T00:00:00.000000000
2021-10-24T00:00:00.000000000
2021-10-28T00:00:00.000000000
2021-11-01T00:00:00.000000000
2021-11-05T00:00:00.000000000
2021-11-09T00:00:00.000000000
2021-11-13T00:00:00.000000000
2021-11-17T00:00:00.000000000
2021-11-21T00:00:00.000000000
2021-11-25T00:00:00.000000000
2021-11-29T00:00:00.000000000
2021-12-03T00:00:00.000000000
2021-12-07T00:00:00.000000000
2021-12-11T00:00:00.000000000
2021-12-15T00:00:00.000000000
2021-12-19T00:00:00.000000000
2021-12-23T00:00:00.000000000
2021-12-27T00:00:00.000000000
2021-12-31T00:00:00.000000000
2022-01-04T00:00:00.000000000
2022-01-08T00:00:00.000000000
2022-01-12T00:00:00.000000000
2022-01-16T00:00:00.000000000
2022-01-20T00:00:00.000000000
2022-01-24T00:00:00.000000000
First
Second
Third
100000
80000
60000
40000
20000
0

As we can see, 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

1.2 Step 1: Explore the dataset

1.2.1 Data types

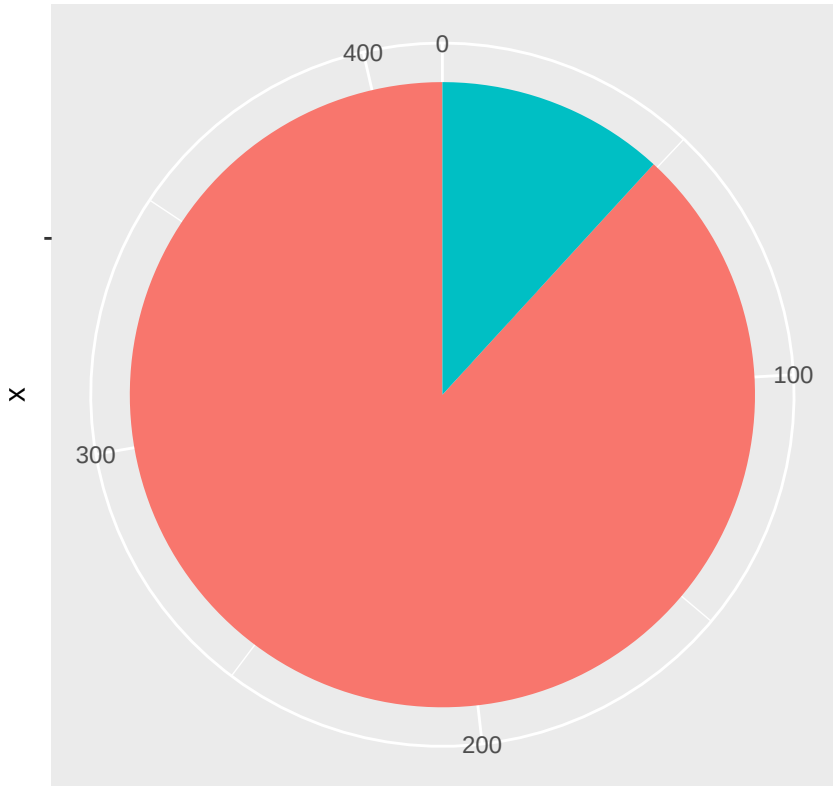
	x
First	double
Second	double
Third	double
Year	double
Month	double
Day	double
DayOfYear	double
Weekday	double
Quarter	double
IsMonthStart	logical
IsMonthEnd	logical

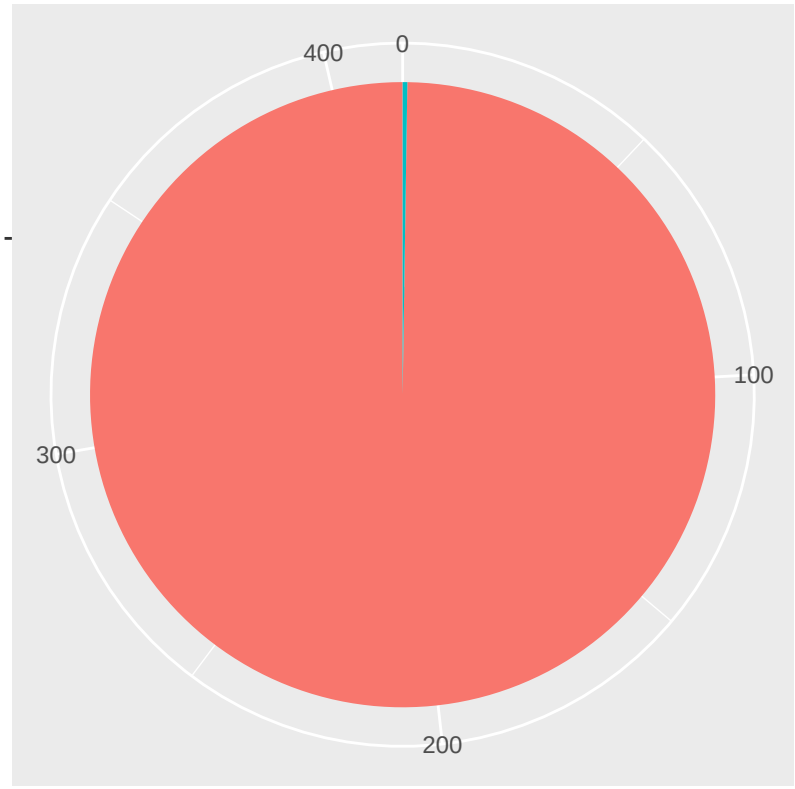
1.2.2 Data description

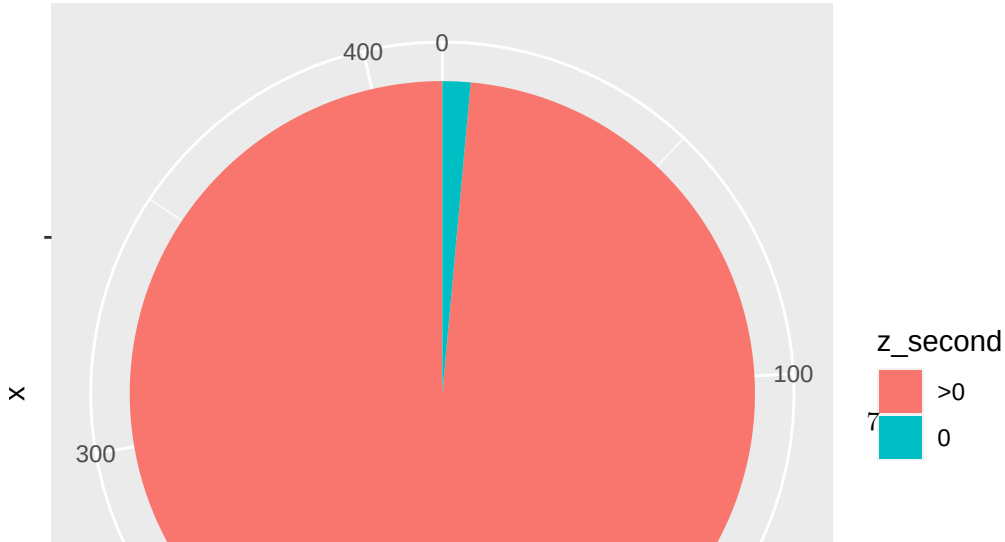
```
## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
First	1	415	11037.614458	1.397158e+04	4210	8067.084084	4505.6214	0	84537	84537	1.9844177	3.951509	685.8377044
Second	2	415	10367.351807	1.347722e+04	3998	7464.363363	5791.0356	0	78425	78425	1.9238326	3.660264	661.5705282
Third	3	415	8507.130120	1.744754e+04	6	4277.684685	7.4130	0	109810	109810	2.7468594	8.772511	856.4660313
Year	4	415	2021.004819	3.474905e-01	2021	2021.000000	0.0000	2020	2022	2	0.0733470	5.258013	0.0170576
Month	5	415	6.496385	3.759607e+00	7	6.495495	4.4478	1	12	11	-0.0039414	-1.332763	0.1845519
Day	6	415	15.800000	8.698159e+00	16	15.804805	10.3782	1	31	30	-0.0047243	-1.172368	0.4269758
DayOfYear	7	415	182.298795	1.153394e+02	182	182.063063	154.1904	1	366	365	0.0092348	-1.316336	5.6617903
Weekday	8	415	2.992771	2.000591e+00	3	2.990991	2.9652	0	6	6	0.0081057	-1.257893	0.0982051
Quarter	9	415	2.501205	1.171057e+00	3	2.501501	1.4826	1	4	3	-0.0028915	-1.480493	0.0574849
IsMonthStart	10	415	NaN	NA	NA	NaN	NA	Inf	-Inf	-Inf	NA	NA	NA
IsMonthEnd	11	415	NaN	NA	NA	NaN	NA	Inf	-Inf	-Inf	NA	NA	NA

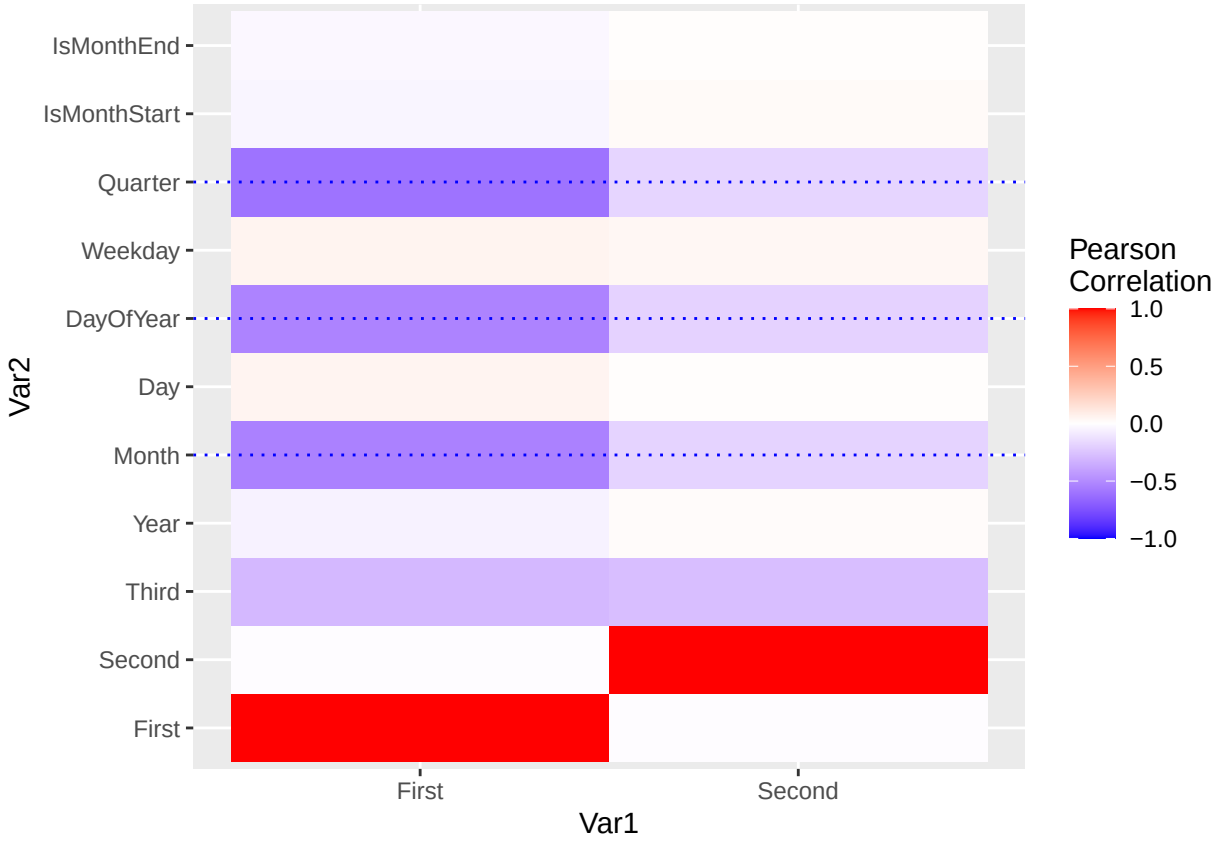
1.2.3 Zeroes


400
0
100
300
200
x
z_third
>0
0
z_third_count


400
0
100
300
200
x
z_first
>0
0
z_first_count


400
0
100
300
x
z_second
>0
0

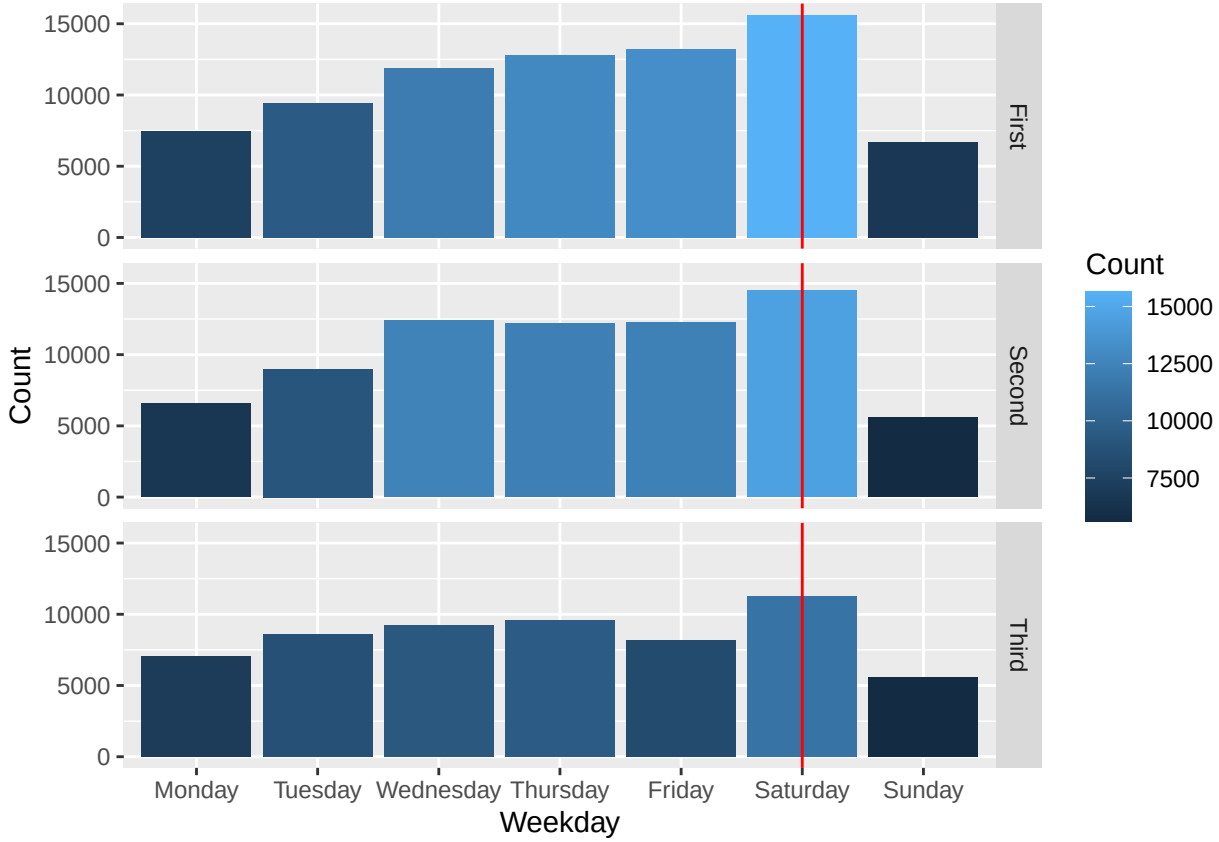
1.2.4 Correlations



Var1	Var2	value
First	Month	-0.5432969
Second	Month	-0.1888013
First	DayOfYear	-0.5343244
Second	DayOfYear	-0.1901012
First	Quarter	-0.6070344
Second	Quarter	-0.1799906

1.2.5 Weekdays

As you remember, I have a question.



Let's answer.

So, most of South West's people prefer to get a jab on Saturdays.

1.2.6 Missing values

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.

1.3 Step 2: Split sets, train a Machine Learning Model and Evaluate performance

Define necessary variables

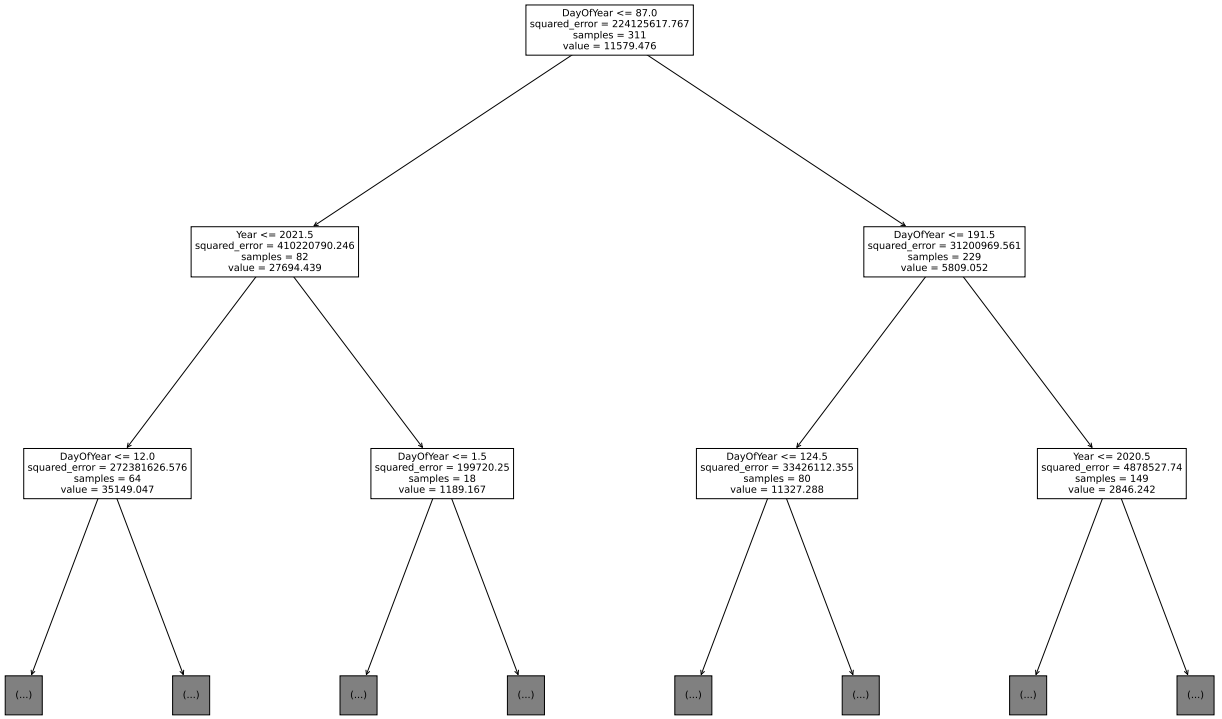
Prepare sets and train models using parameters.

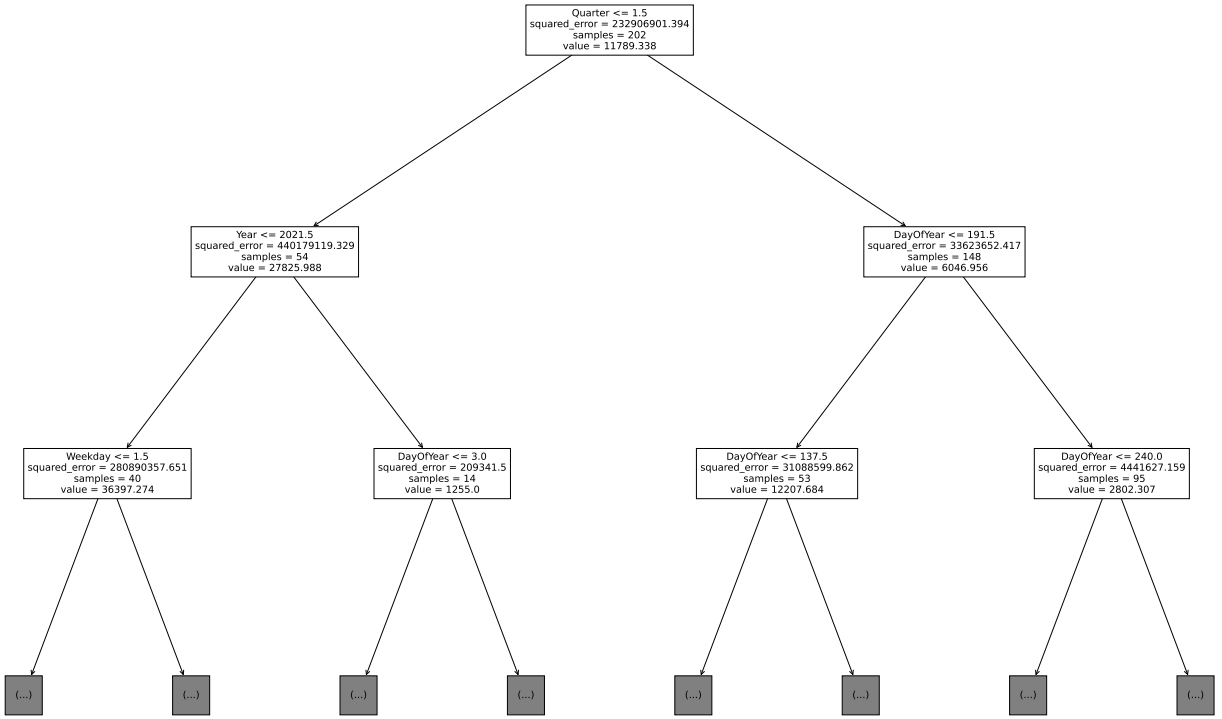
```
y_column = "First"
```

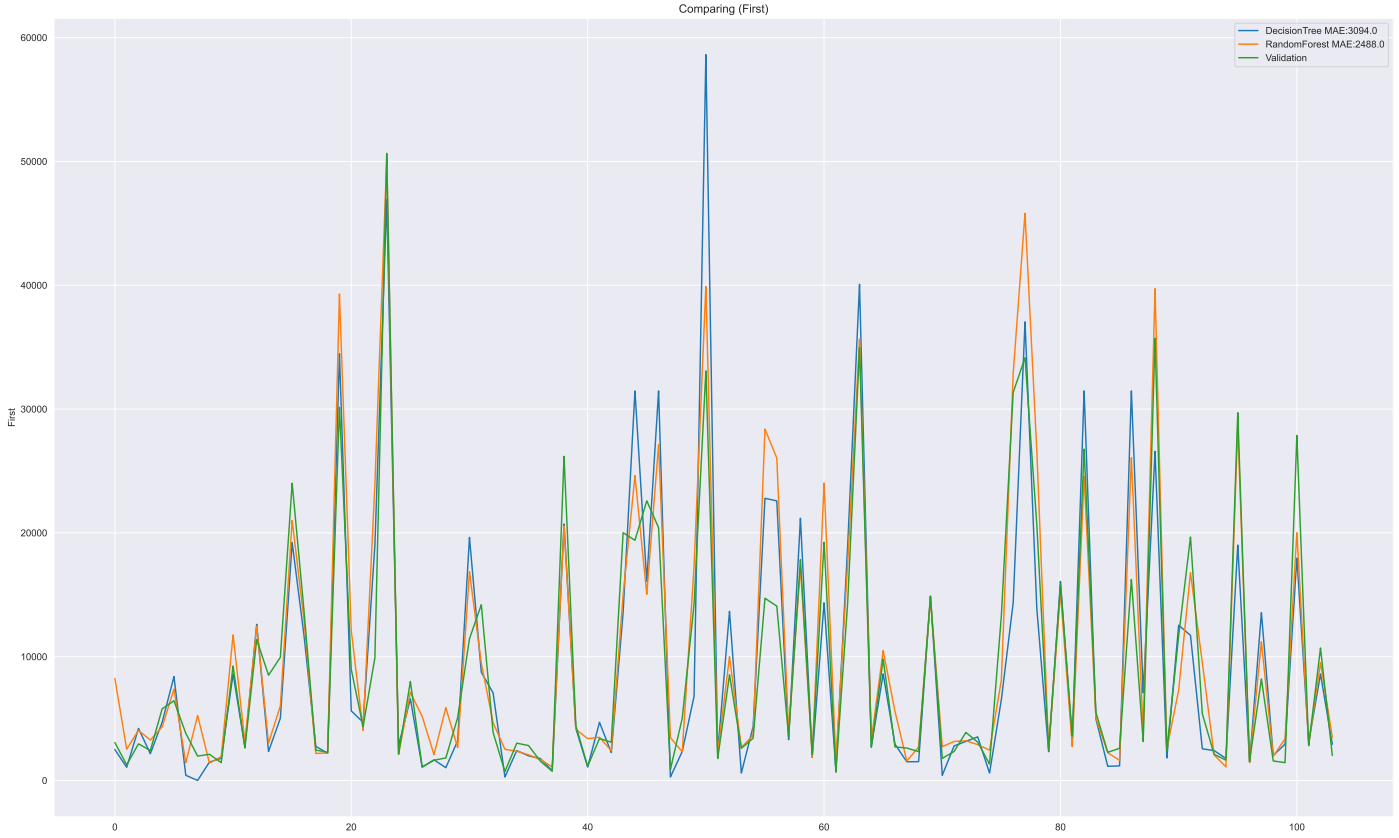
```
## DecisionTree:  0.719657929335243
```

```
## RandomForest:  0.774580856609961
```

Look at the tree

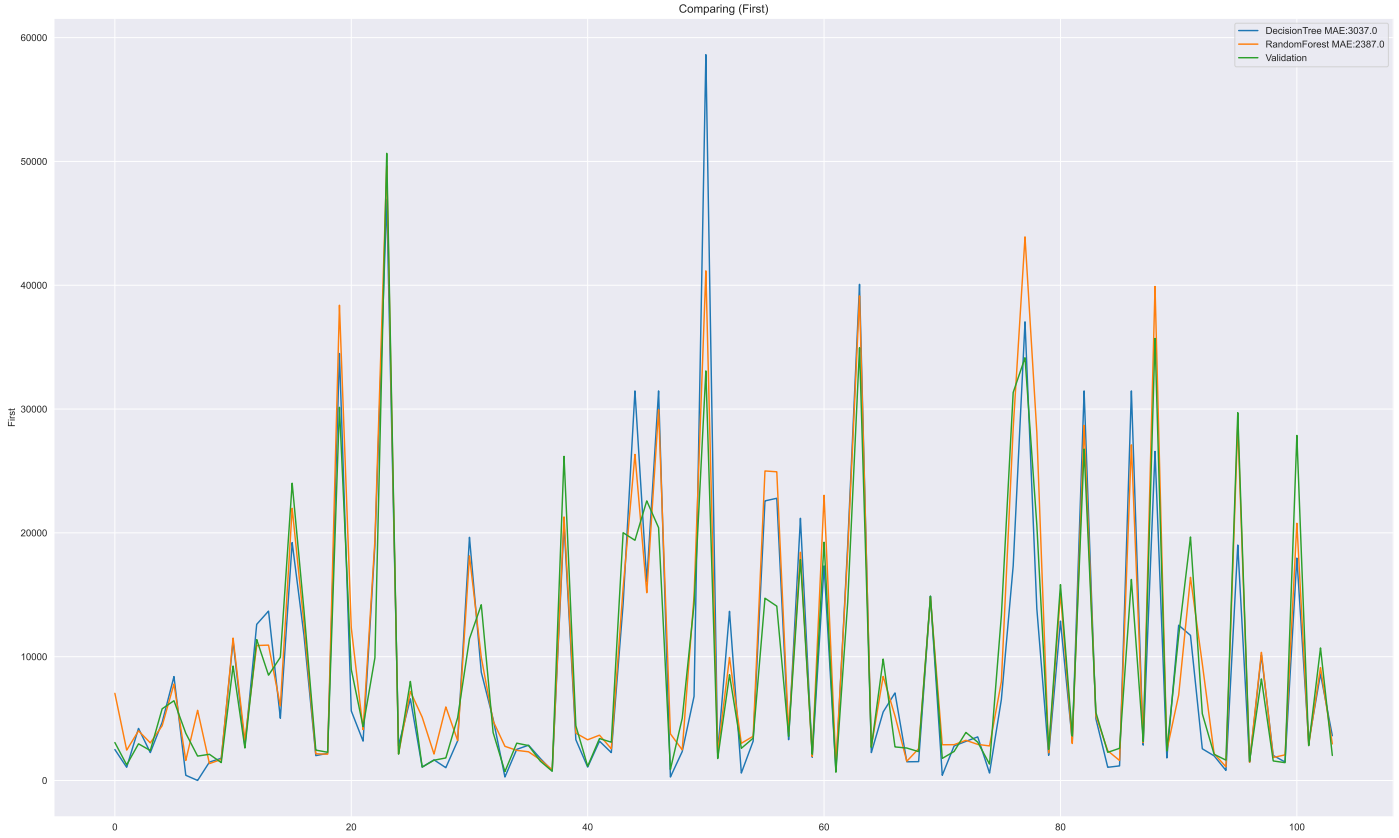

DayOfYear <= 87.0
squared_error = 224125617.767
samples = 311
value = 11579.476
Year <= 2021.5
squared_error = 410220790.246
samples = 82
value = 27694.439
DayOfYear <= 191.5
squared_error = 31200969.561
samples = 229
value = 5809.052
DayOfYear <= 12.0
squared_error = 272381626.576
samples = 64
value = 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
squared_error = 4401791 19.329
samples = 54
value = 27825.988
DayOfYear <= 191.5
squared_error = 33623652.417
samples = 148
value = 6046.956
Weekday <= 1.5
squared_error = 280890357.651
samples = 40
value = 36397.274
DayOfYear <= 3.0
squared_error = 209341.5
samples = 14
value = 1255.0
DayOfYear <= 137.5
squared_error = 31088599.862
samples = 53
value = 12207.684
DayOfYear <= 240.0
squared_error = 4441627.159
samples = 95
value = 2802.307
(...)
(...)
(...)
(...)
(...)
(...)
(...)
(...)




```
## DecisionTree:  0.7248630326024768
```

```
## RandomForest:  0.7837038702898657
```

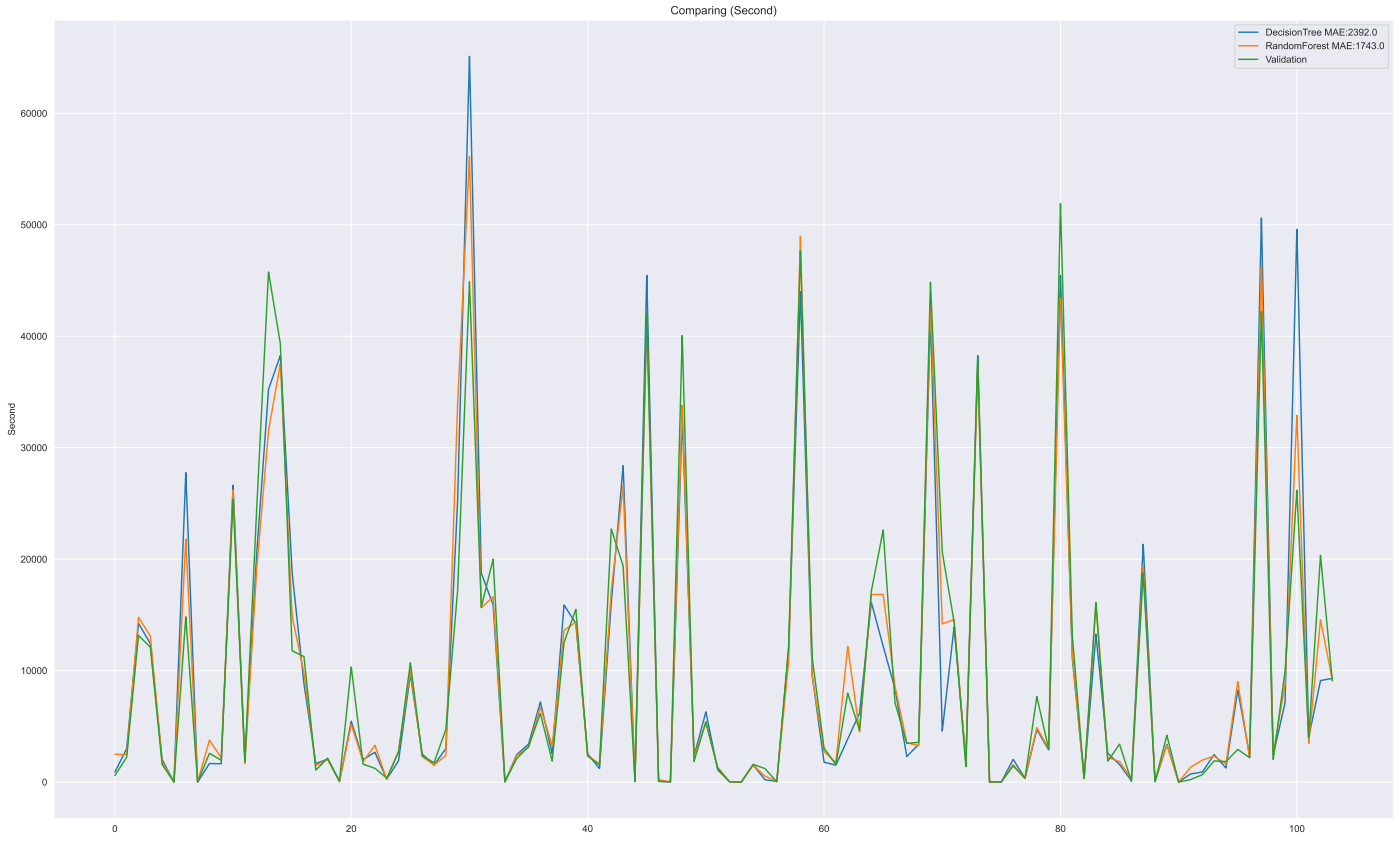


Repeat for the Second

```
y_column = "Second"
```

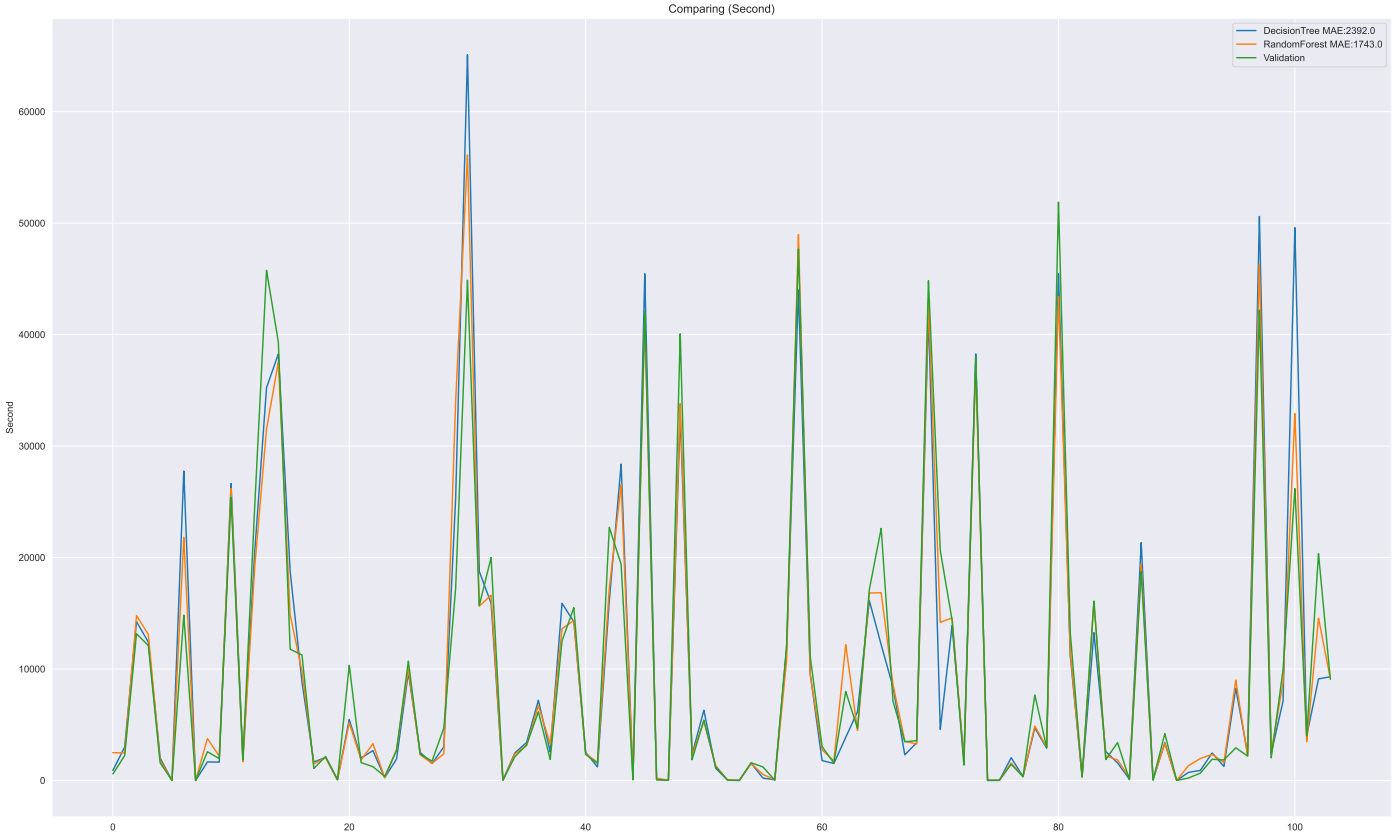
```
## DecisionTree:  0.7692636418171874
```

```
## RandomForest:  0.8318561653086721
```




```
## DecisionTree:  0.7692636418171874
```

```
## RandomForest:  0.8318561653086721
```

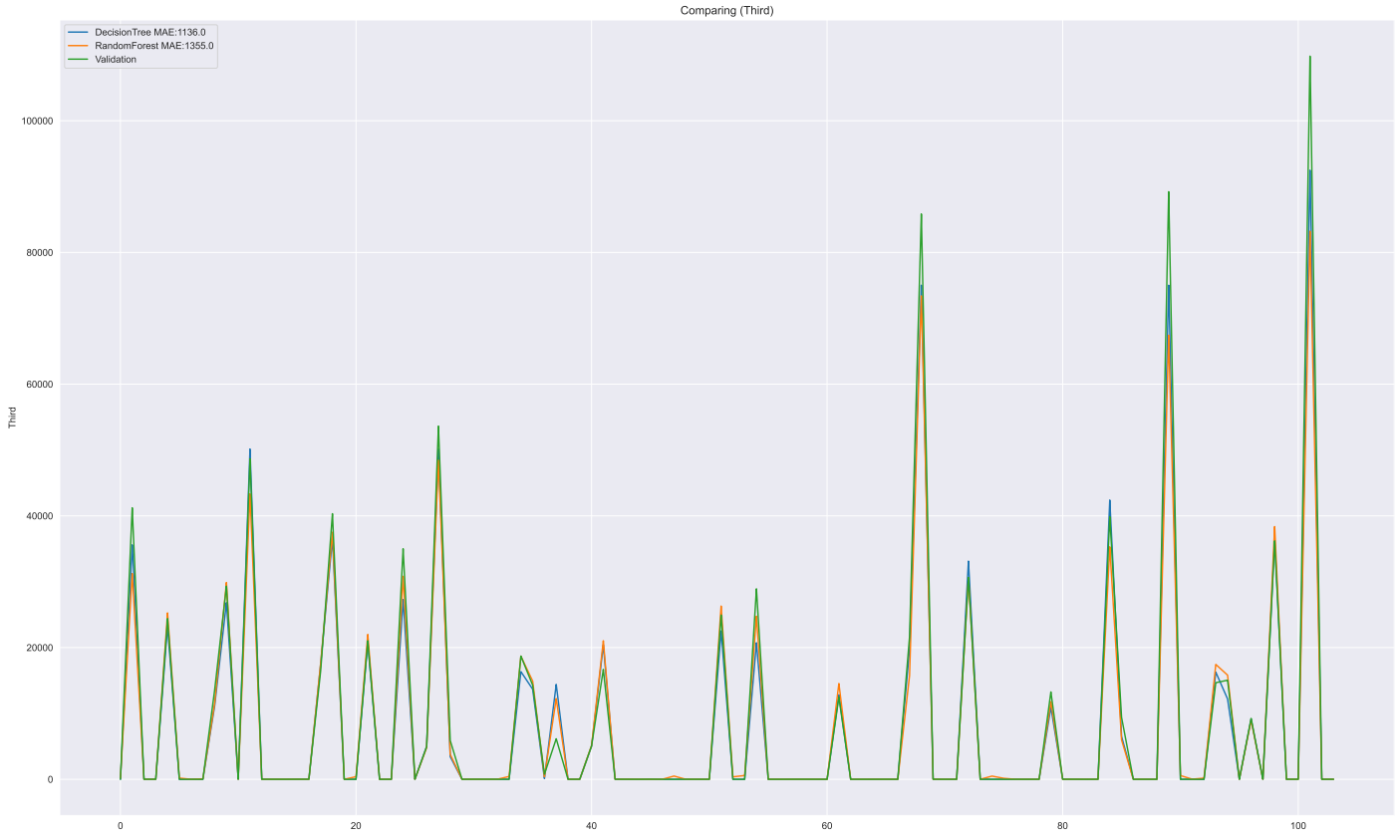


Repeat for Third

```
y_column = "Third"
```

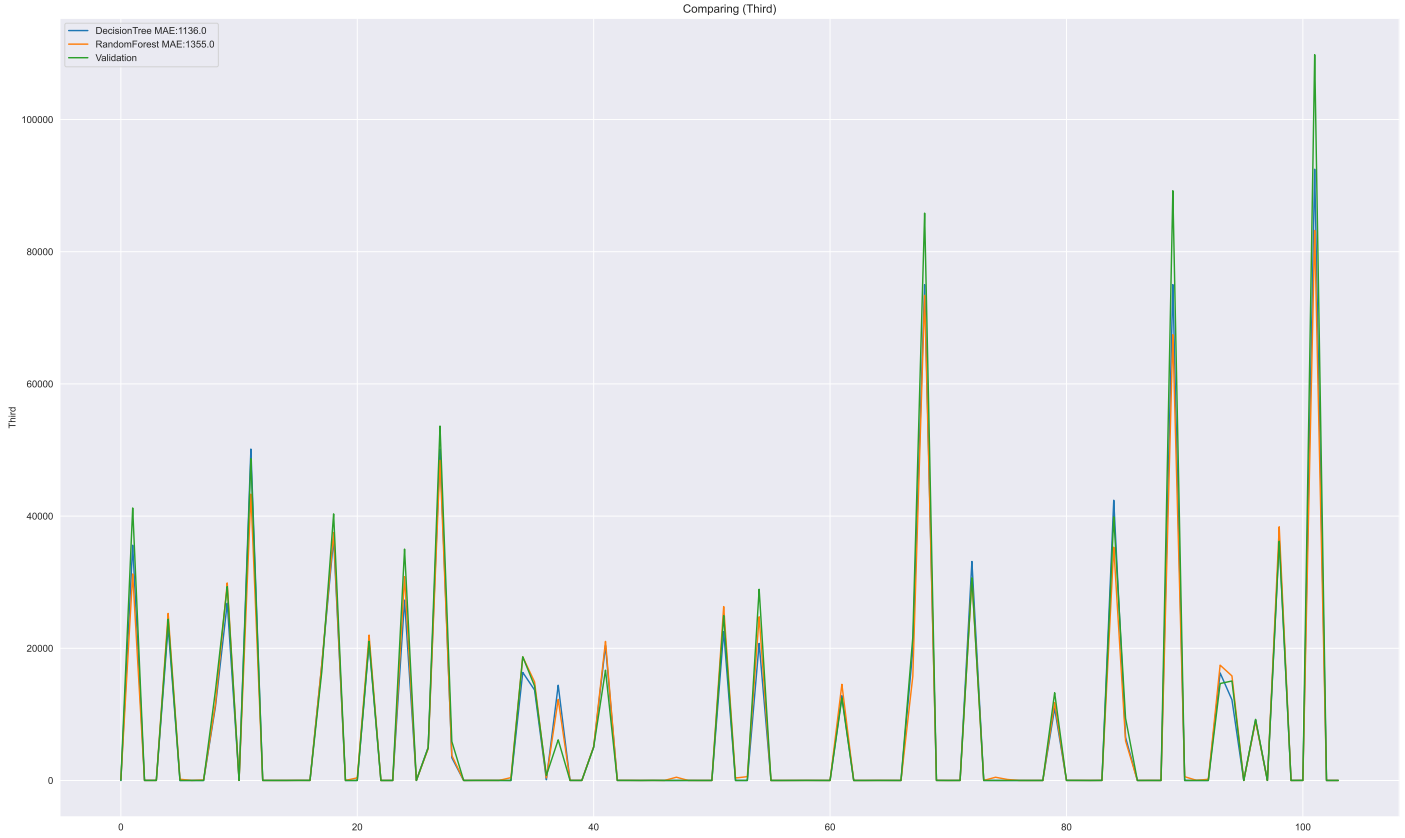
```
## DecisionTree:  0.8664423499345815
```

```
## RandomForest:  0.840745098066024
```




```
## DecisionTree:  0.8664423499345815
```

```
## RandomForest:  0.840745098066024
```



Compare the score with the mean value of the column that we predicted.

A combination of the following features give us the best result:

- Weekday,
- Year,

- DayOfYear