Covid-19 Project prediction

Statistical Methods for Data Science
a.a. 2021 - 2022

Cavuoti Lorenzo, Sicklinger Marco, Pinna Giovanni

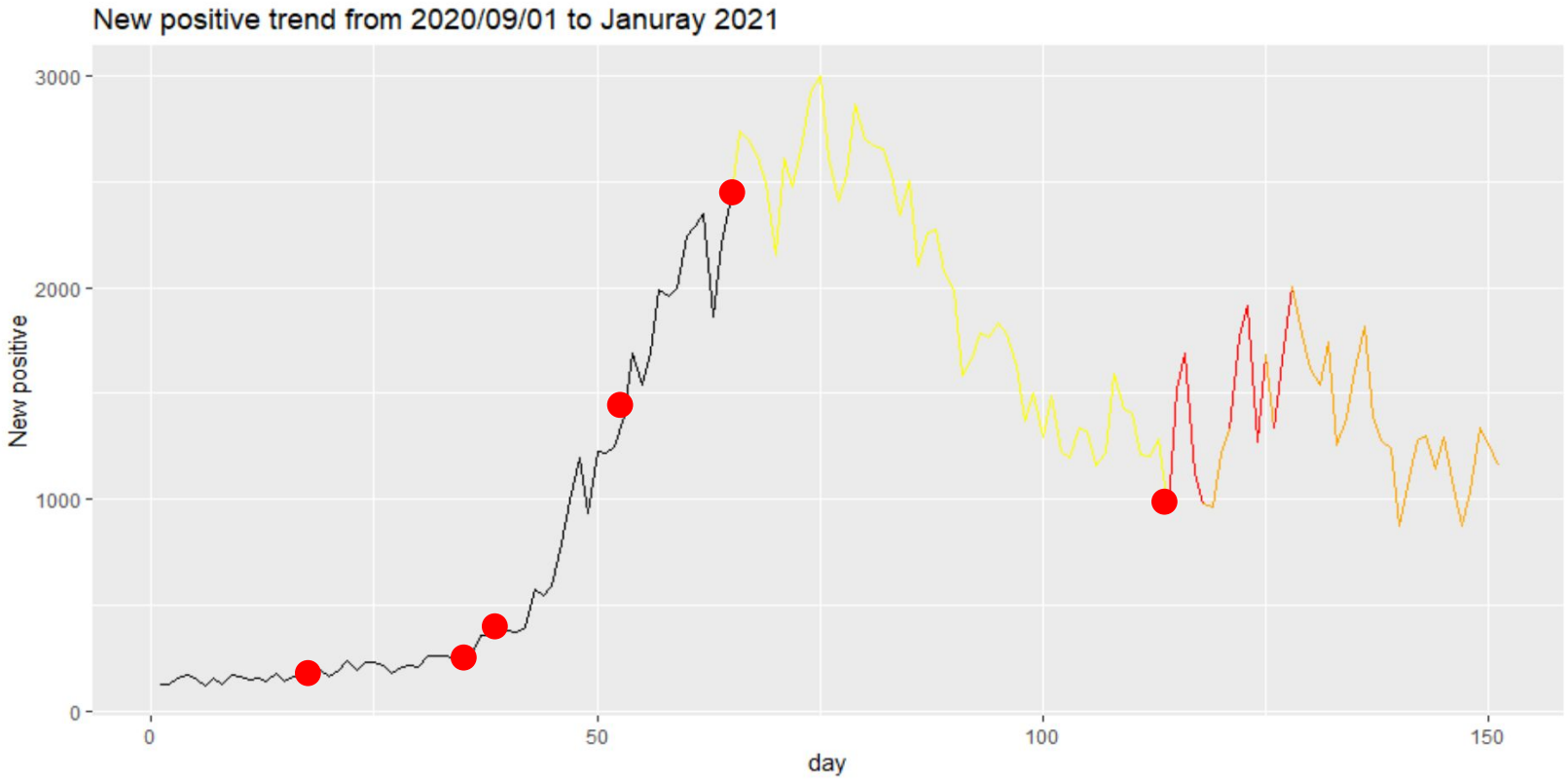
Target of the analysis

Our goal is to find the best model to predict 15 days into the future on the new positives of covid-19.

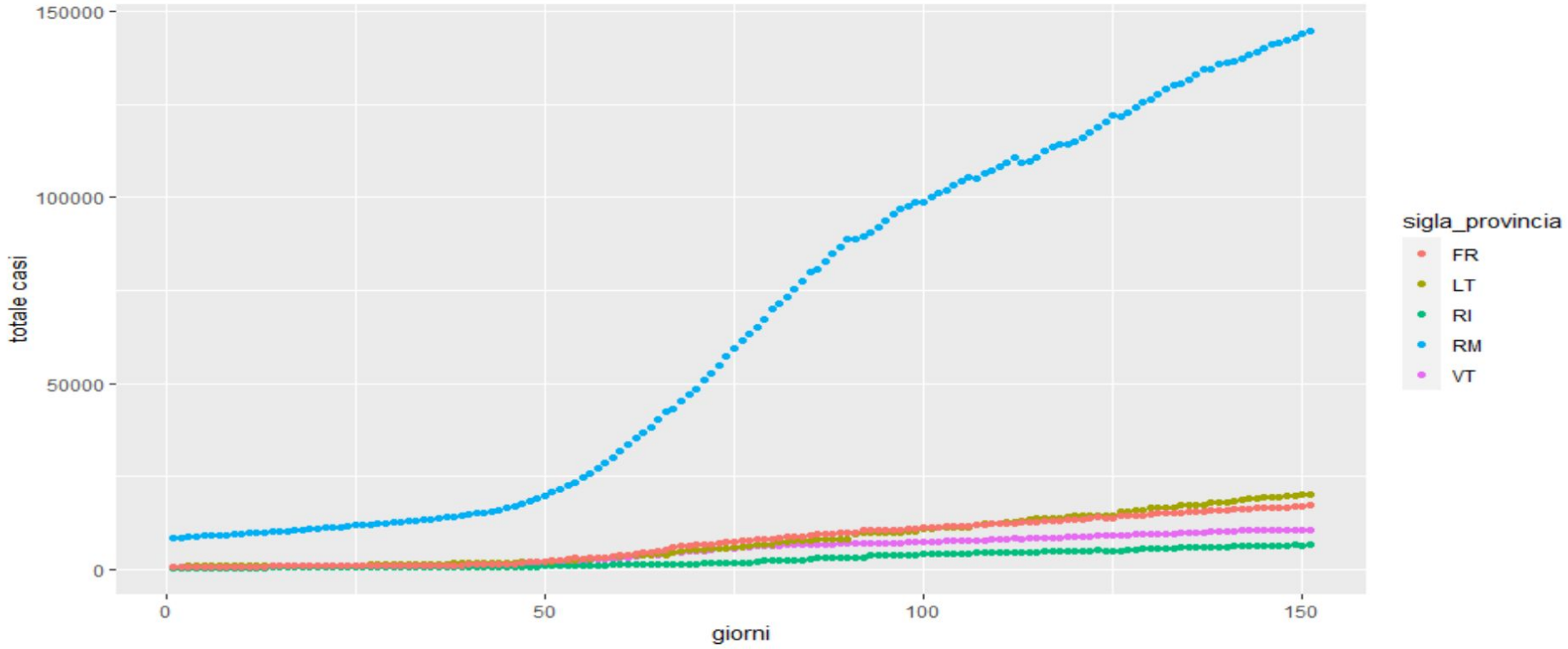
The data that have been provided to us are all data from 1 September onwards, which civil protection provides daily.

Pandemic history of Lazio

New positive trend over time



Importance of the provinces on total cases



Features original dataset

The dataset contains the following variables:

- **data:** Date of notification
- **stato:** Country of reference
- **codice_regione:** Code of the Region (ISTAT 2019)
- **denominazione_regione:** Name of the Region
- **lat:** Latitude
- **long:** Longitude
- **ricoverati_con_sintomi:** Hospitalised patients with symptoms
- **terapia_intensiva:** Intensive Care
- **ingressi_terapia_intensiva:** Daily admissions to intensive care
- **totale_ospedalizzati:** Total hospitalised patients
- **isolamento_domiciliare:** Home confinement
- **totale_positivi:** Total amount of current positive cases (Hospitalised patients + Home confinement)
- **variazione_totale_positivi:** New amount of current positive cases (totale_positivi current day - totale_positivi previous day)
- **nuovi_positivi:** New amount of current positive cases (totale_casi current day - totale_casi previous day)
- **dimessi_guarit:** i Recovered
- **deceduti:** Death: (cumulated values)
- **totale_casi:** Total amount of positive cases
- **tamponi:** Tests performed
- **casi_testati:** Total number of people tested

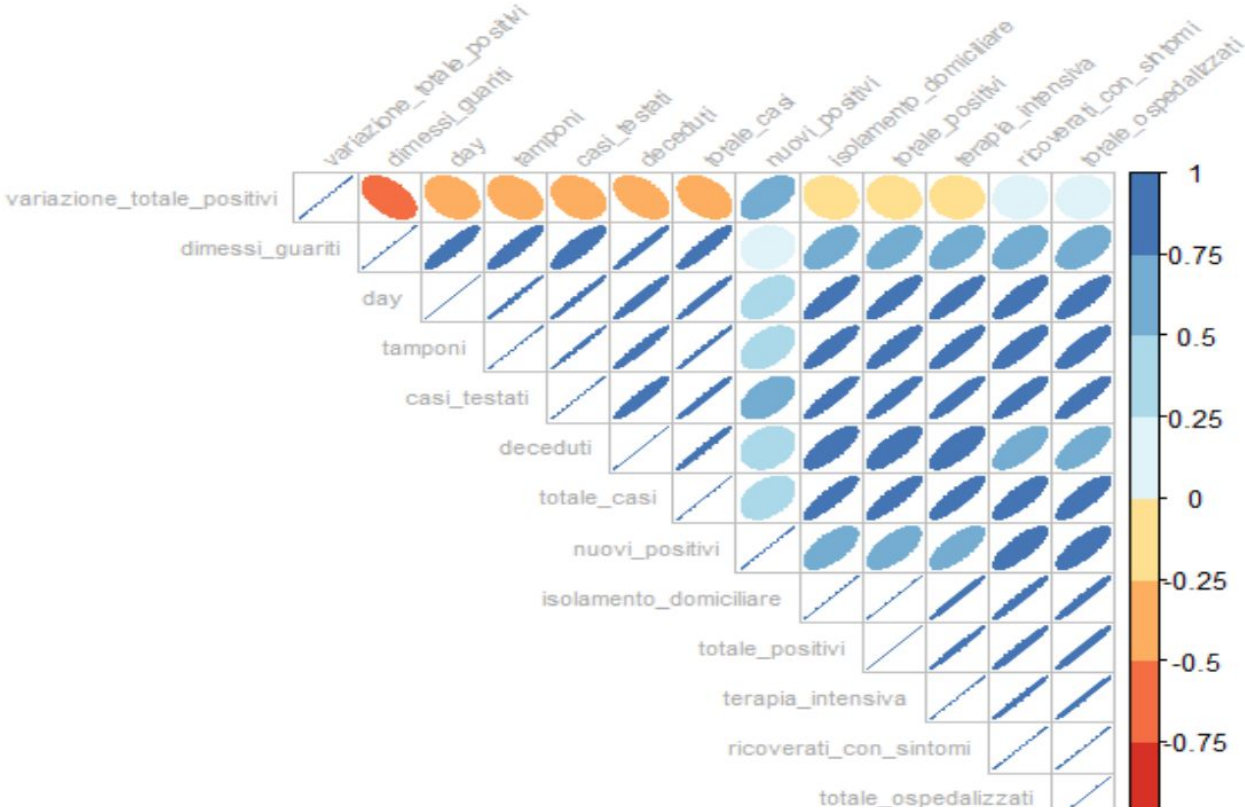
About new data used for prediction

To improve our models and prediction we have decided to introduce two features that we always know in advance.

- In particular, the first is the **color of the region**, even if Lazio in particular has always been yellow until Christmas.

- The other features introduced was the **high_school**, which takes on one value if it is open and another if it can only be done online.

Features Correlation



Features that we used

The dataset contains the following variables:

- **color:** color of the region Lazio
- **high_school:** if the school is open or online
- **new_positive_1prevpoint:** the new positives of the nth-day before the one under consideration
- **new_positive_2prevpoint:** the new positives of (nth-1)-day before the one under consideration
- **recovered_1prevpoint:** hospitalized of the nth-day before the one under consideration
- **hospitalized:** people in the hospital with covid-19
- **day:** day under consideration
- **tests:** cases tested

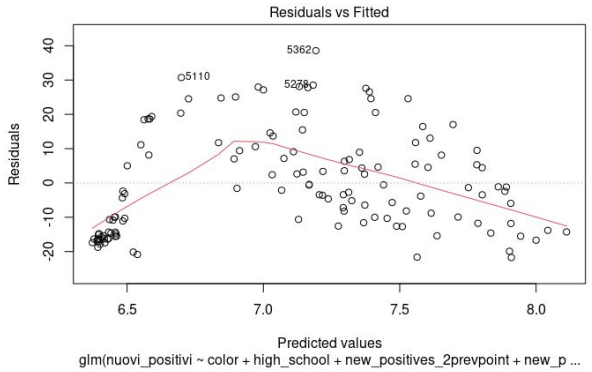
Regression with Poisson

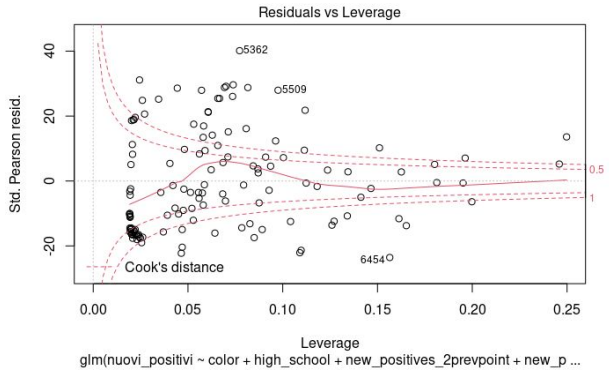
- **Poisson regression** is often used to model count data.
- As a consequence, modelling the outbreak of an infectious disease can be done within the Poisson regression framework.
- However, it is wise to expect that this model will not be the final model, due to possible **overdispersion** of the data.

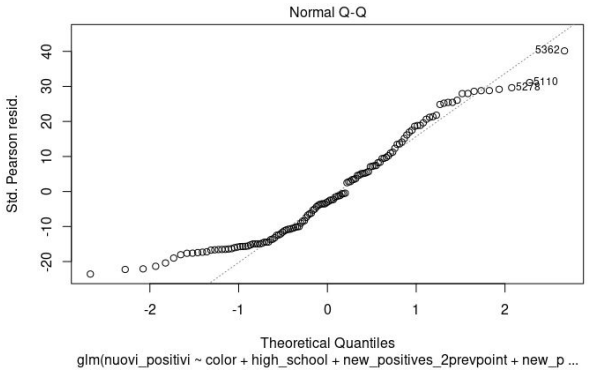
Regression with Poisson

```
glm(formula = nuovi_positivi ~ color + high_school + new_positives_2prevpoint +
    new_positives_1prevpoint + recovered_1prevpoint + tests_1prevpoint,
    family = poisson(link = "log"), data = lazio)
```

Akaike Information Criterion (AIC)	R^2
30197	0.606







Regression with Negative Binomial

- **Negative binomial regression** can be used for count data that exhibit overdispersion.

- It can be considered a generalization of the previous Poisson model: Negative Binomial has larger mean and an extra parameter which makes it more flexible for modelling overdispersed count data.

Regression with Negative Binomial: GLM and GAM approaches

GLM approach

```
glm.nb(formula = nuovi_positivi ~ color + high_school + new_positives_2prevpoint +
    new_positives_1prevpoint + recovered_1prevpoint + tests_1prevpoint,
    data = lazio, link = "log", init.theta = 4.474958335)
```

Akaike Information Criterion (AIC)	R^2
2037.25	0.604

GAM approach

```
gam(formula = nuovi_positivi ~ color + high_school + s(new_positives_2prevpoint) +
    s(new_positives_1prevpoint) + s(recovered_1prevpoint) + s(tests_1prevpoint),
    family = nb(link = "log"), data = lazio)
```

Akaike Information Criterion (AIC)	adjusted R^2
1648.60	0.968

The Generalized Additive Model is the best model for the dependent variable “nuovi_positivi”.

Regression with Negative Binomial

Estimates

- “color” factor level “yellow” is significant only at 0.1 level
- “high_school” factor is not significant

```
Parametric coefficients:
                        Estimate Std. Error z value Pr(>|z|)
(Intercept)              6.77997    0.07075  95.836  < 2e-16 ***
coloryellow              0.17891    0.09753   1.834  0.06660 .
colororange              0.35717    0.16634   2.147  0.03178 *
colorred                 0.46553    0.15194   3.064  0.00218 **
high_schoolin presence   0.00809    0.06050   0.134  0.89362
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Approximate significance of smooth terms:
                           edf Ref.df   Chi.sq p-value
s(new_positives_2prevpoint) 1.000  1.000    4.524 0.03342 *
s(new_positives_1prevpoint) 1.002  1.005    7.274 0.00713 **
s(recovered_1prevpoint)     2.642  3.370   13.919 0.00415 **
s(tests_1prevpoint)         8.537  8.778 2013.220 < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

R-sq.(adj) =  0.968   Deviance explained = 98.1%
```

Regression with Negative Binomial

Removal of non-significant covariate

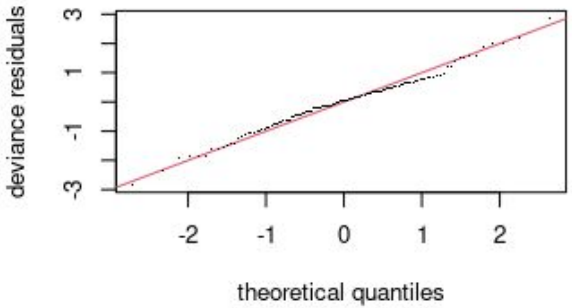
The improvement obtained by removing the non-significant variables are small, compared to the results obtained with the more complex model:

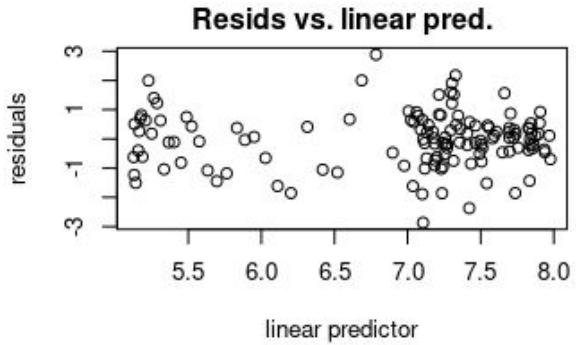
- AIC decreases to 1646.60
- R^2 does not change

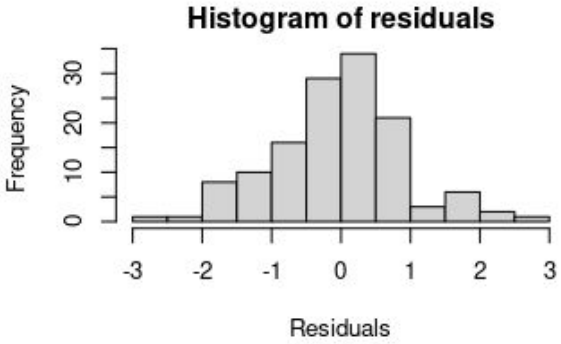
```
Parametric coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  6.78506    0.06443 105.302  < 2e-16 ***
coloryellow  0.17692    0.09685   1.827  0.06773 .
colororange  0.35223    0.16495   2.135  0.03273 *
colorred     0.46045    0.15068   3.056  0.00225 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Approximate significance of smooth terms:
                                edf Ref.df   Chi.sq p-value
s(new_positives_2prevpoint) 1.000  1.000    4.603 0.03195 *
s(new_positives_1prevpoint) 1.004  1.008    7.498 0.00639 **
s(recovered_1prevpoint)     2.581  3.295   14.613 0.00276 **
s(tests_1prevpoint)         8.572  8.793 2070.735 < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

R-sq.(adj) =  0.968   Deviance explained = 98.1%
```

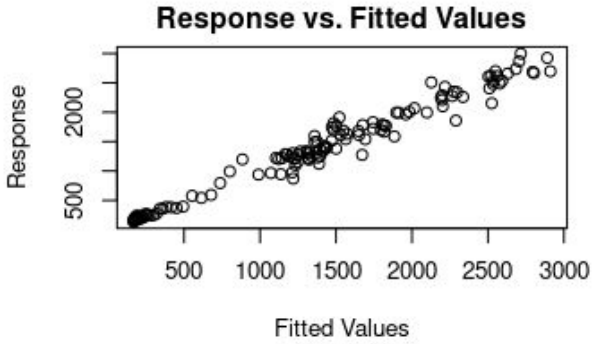
Regression with Negative Binomial

Diagnostic plots









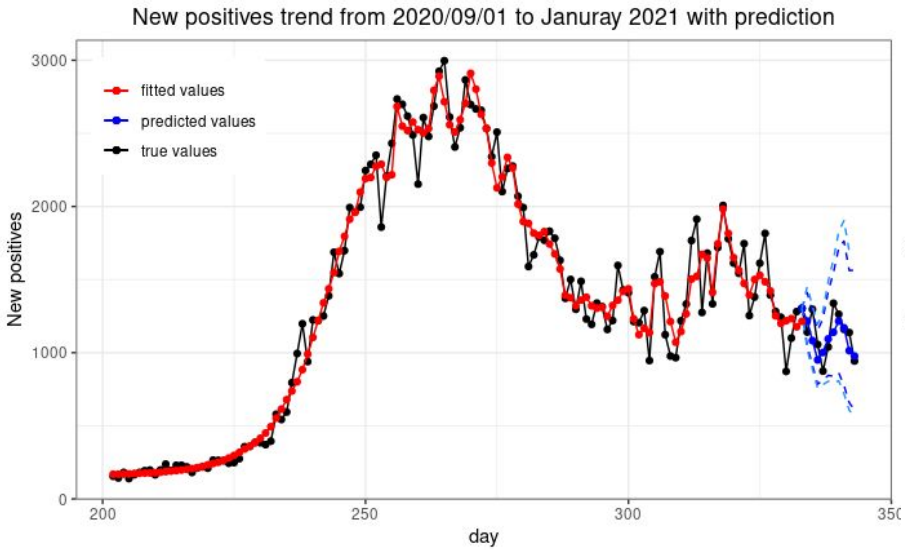
Predictions with Negative Binomial

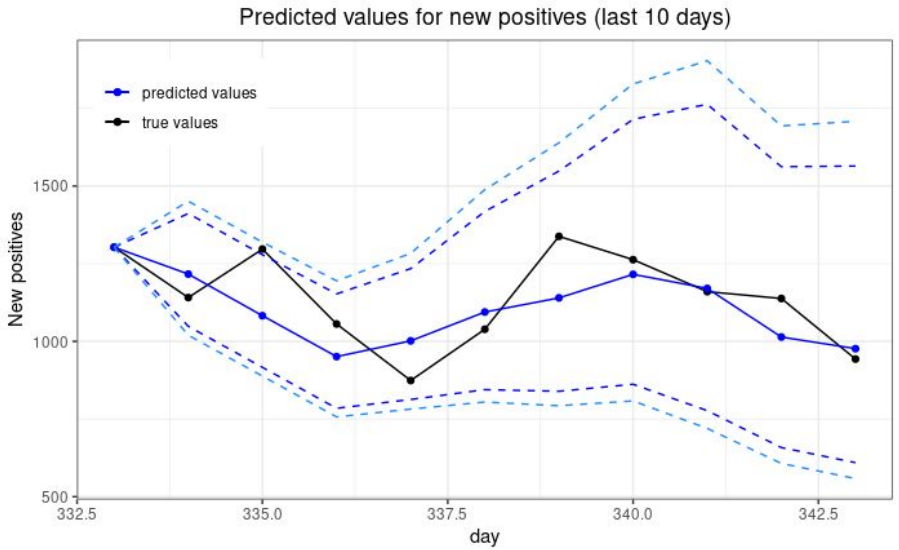
Concept

- Predictions of successive points in time are done exploiting values available only for previous points in time.
- If values related to covariates different from the dependent variable are to be used, it is necessary to implement predictions using only data available at the point in time at which “nuovi_positivi” has to be predicted.
- That is, if the predictions span over an n-days long period of time, we used the values covariates had n days earlier.

Predictions with Negative Binomial

Plots & results





Mean Absolute Percentage Error (MAPE)
8.7%

Non-parametric methods

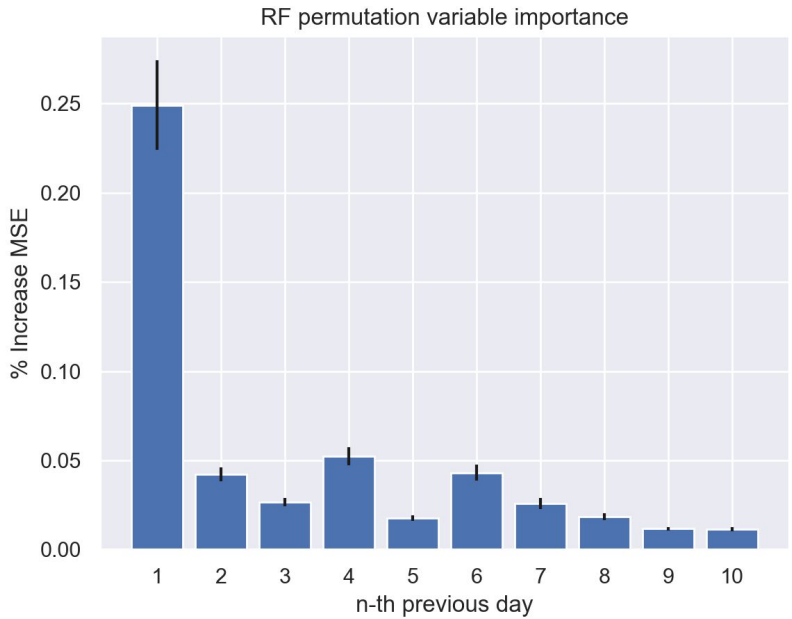
We will predict the new positives of tomorrow given the variables of the preceding days using:

- A random forest with 500 trees
- Gradient boosting implemented with XGBoost

Random Forest

Choice of window size

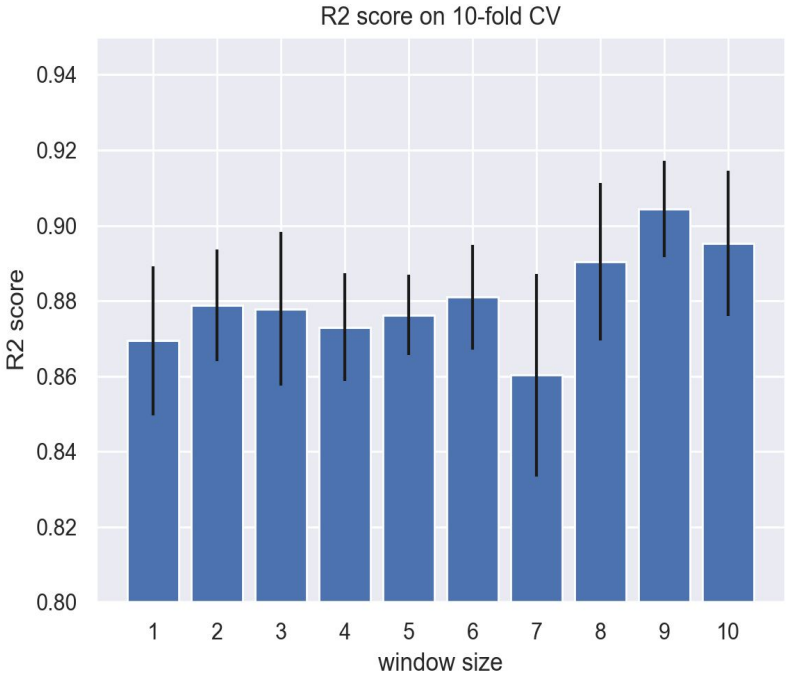
- We started by only considering the new positives values of the preceding days
- We see that increasing the window size provides diminishing returns



Random Forest

Choice of window size

- Looking at the R2 score there seems to be a benefit from considering a larger window
- However all the scores are within one standard deviation from each other
- According to Occam's razor we will choose a window size=1



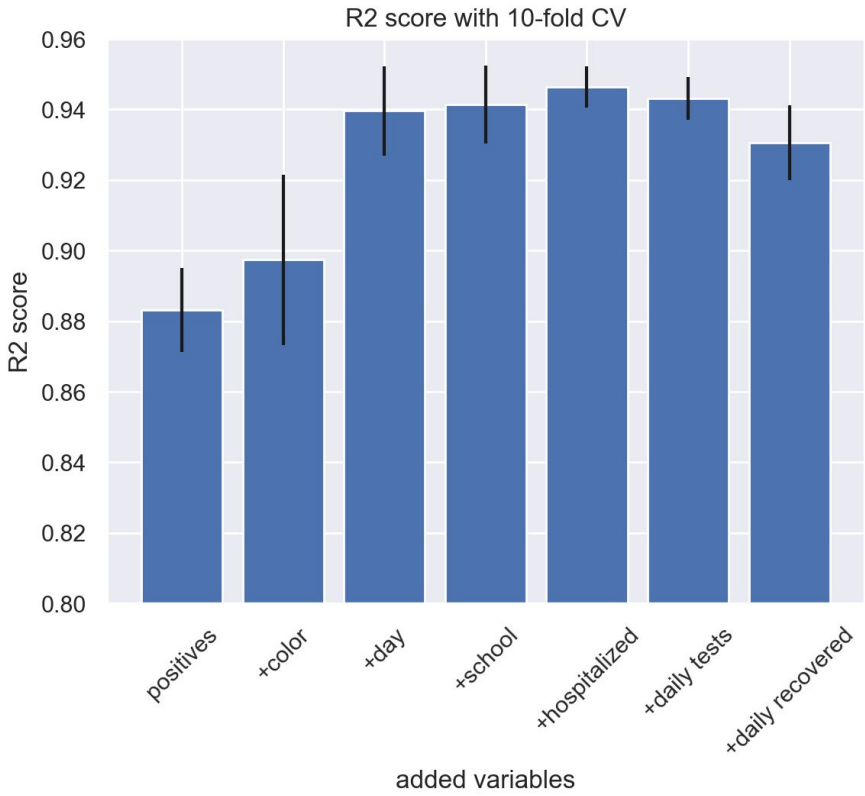
R2 window size=1
0.87±0.02

Random Forest

Considering more covariates

- We tried various combinations of variables mainly with a forward selection approach
- The best model considers the variables: positives, color and day

R2 best model
0.94±0.01



Regression with XGBoost

Concept

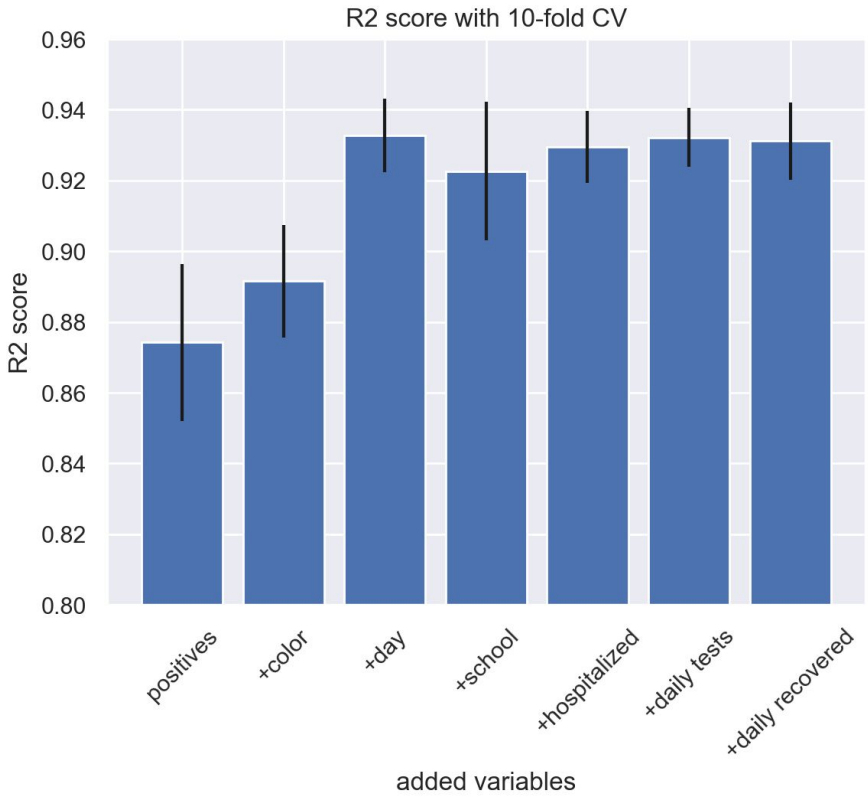
- XGBoost has over 10 hyperparameters and can overfit
- We setted a learning rate of 0.01 and n_estimators=10000 and left the others at default values
- Now our model will overfit so we keep track of the CV error during training and stop when it doesn't improve
- Testing is done on the test set

Regression with XGBoost

Results

- The best variable combination is the same as with random forest
- XGBoost performs worse than Random Forest

R2 best model
0.93±0.01



Recap on regression models

- GLM models seem to suffer from the assumption of linearity
- The Negative Binomial GAM performs the best
- The fact that GAM outperforms non-parametric methods suggests that feature interaction isn't important

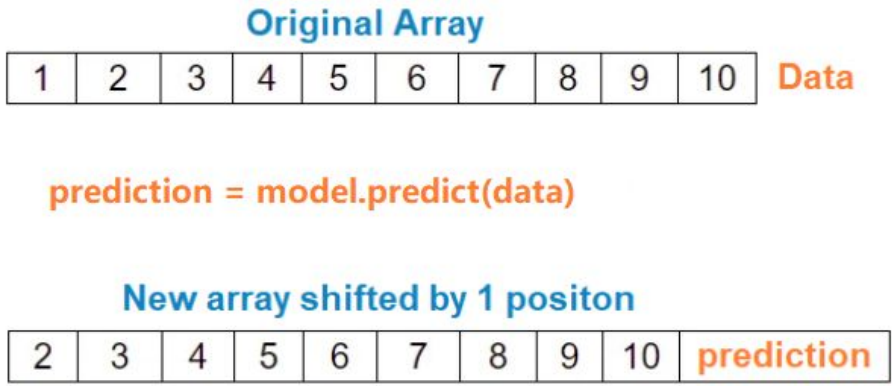
Model	R2 coefficient
Poisson GLM	0.606
Negative Binomial GLM	0.604
Negative Binomial GAM	0.968
Random Forest	0.94
Boosting	0.93

Long term predictions

Concept

- Considering the complexity of XGBoost and the lower R2, we will use a random forest for the predictions
- We will use a recursive approach to provide long term predictions

```
for i in 1:days
    prediction = model.predict(data)
    data = shift(data, prediction)
```



Long term predictions

Confidence intervals

- Using random forest we can easily provide confidence intervals for the prediction

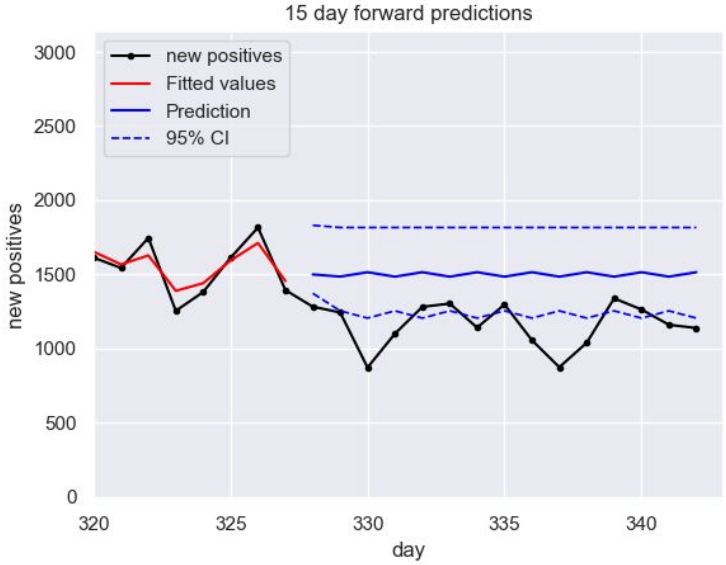
1. A simple approach is to evaluate all the trees in the forest then take the mean and the quantiles
2. Another one is to evaluate a tree at random at each prediction step and repeat this process n times, this will give n traces which we then compute the mean and quantiles

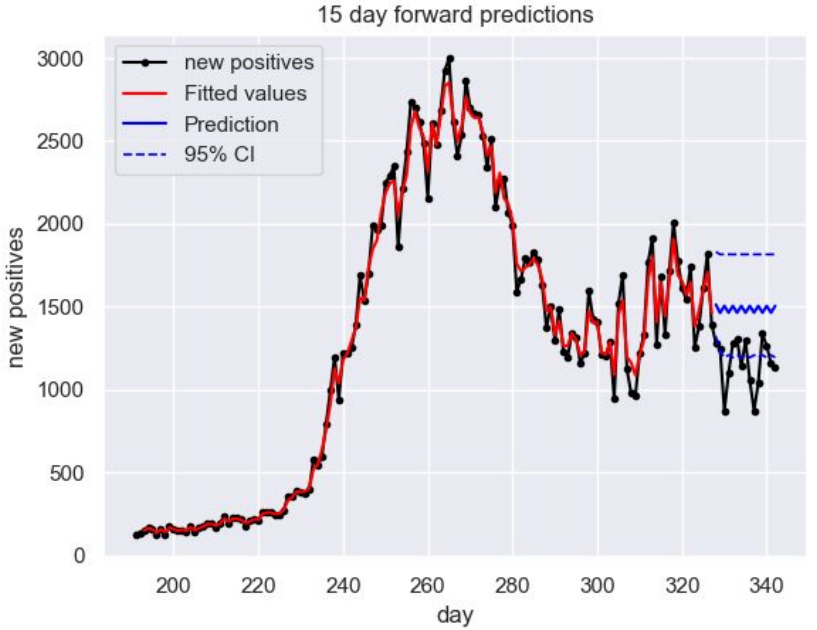
- The two methods provide similar results so we will use method 1

15 days predictions

Simple model

- The model fails when trying to predict more days





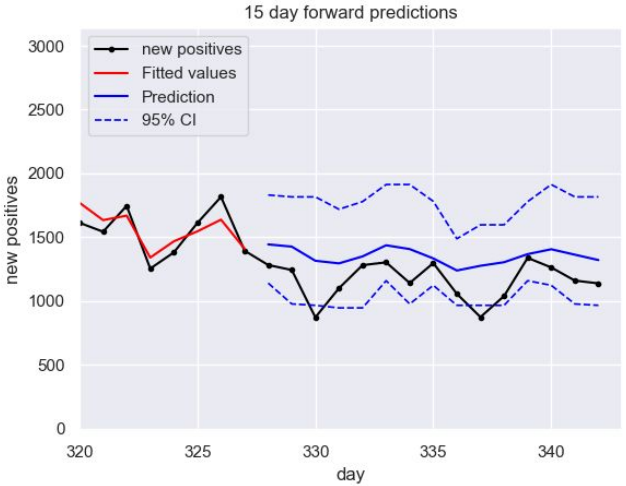
Mean Absolute Percentage Error (MAPE)
31%

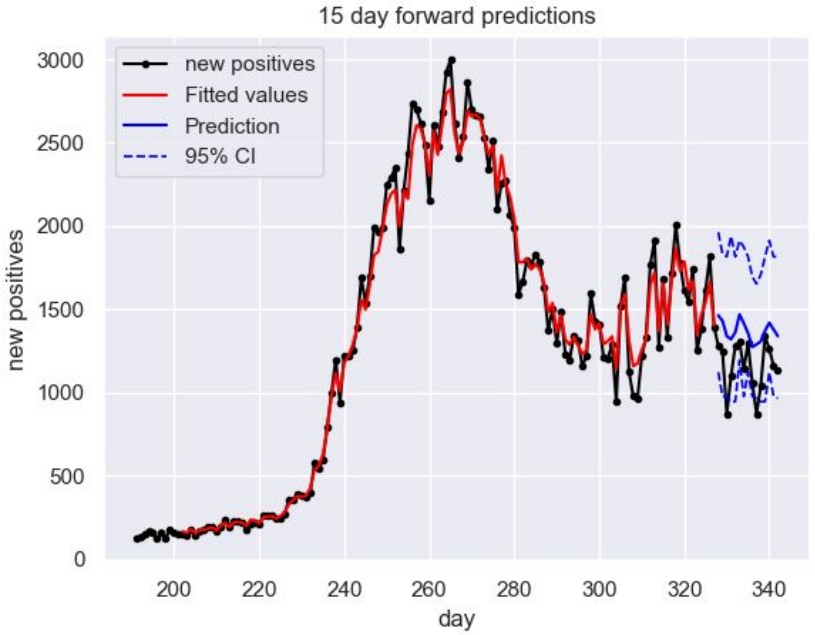
15 days predictions

A better prediction model

1. window size increased to 10
2. day_of_the_week = day%7

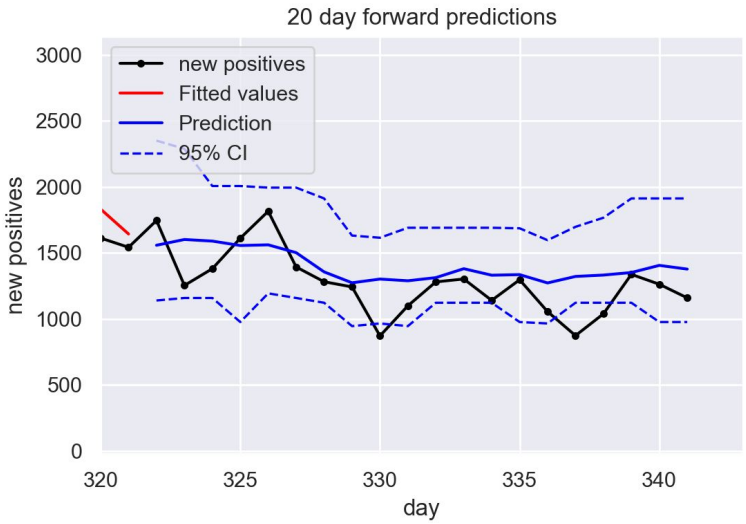
- Random forest benefits from larger windows than GLM/GAM

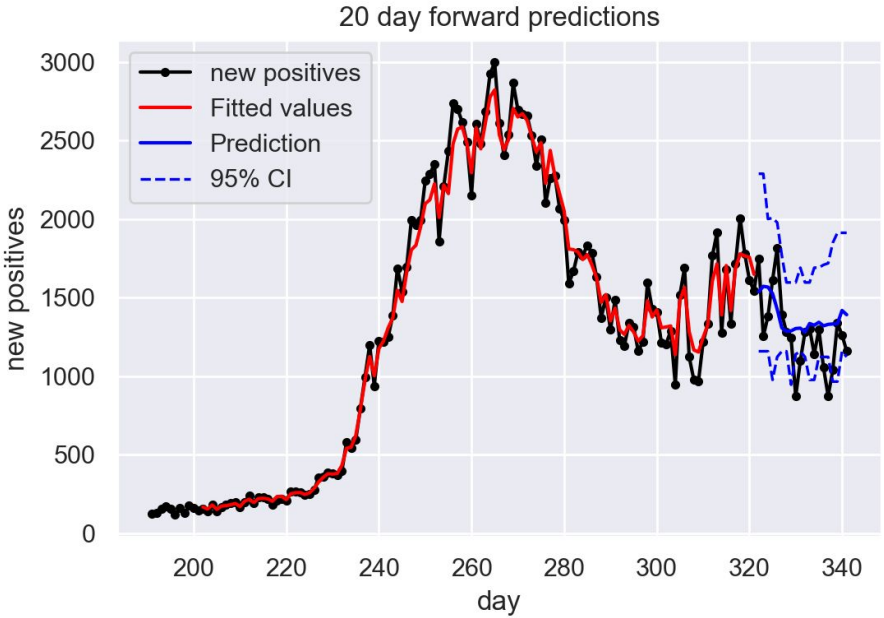




Mean Absolute Percentage Error (MAPE)
17%

20 day predictions

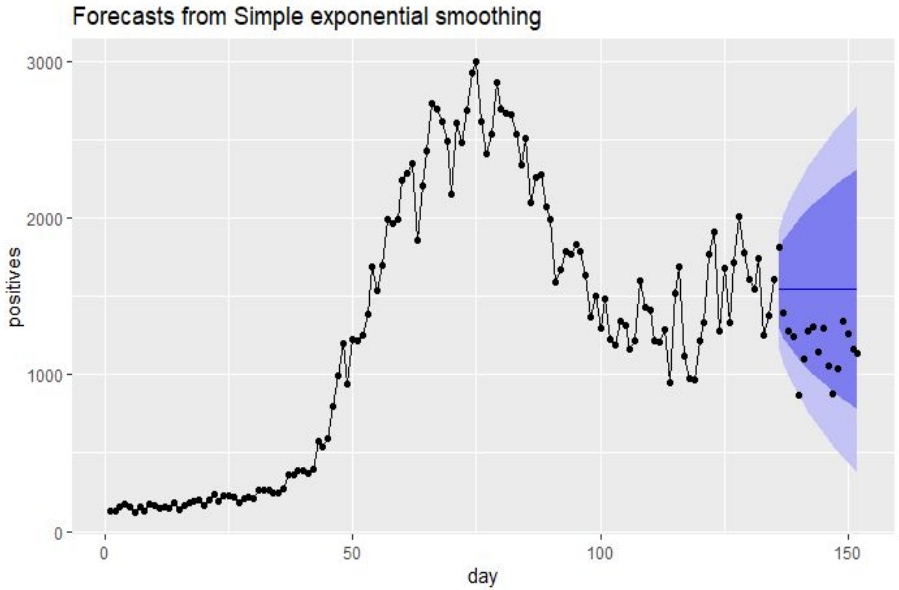


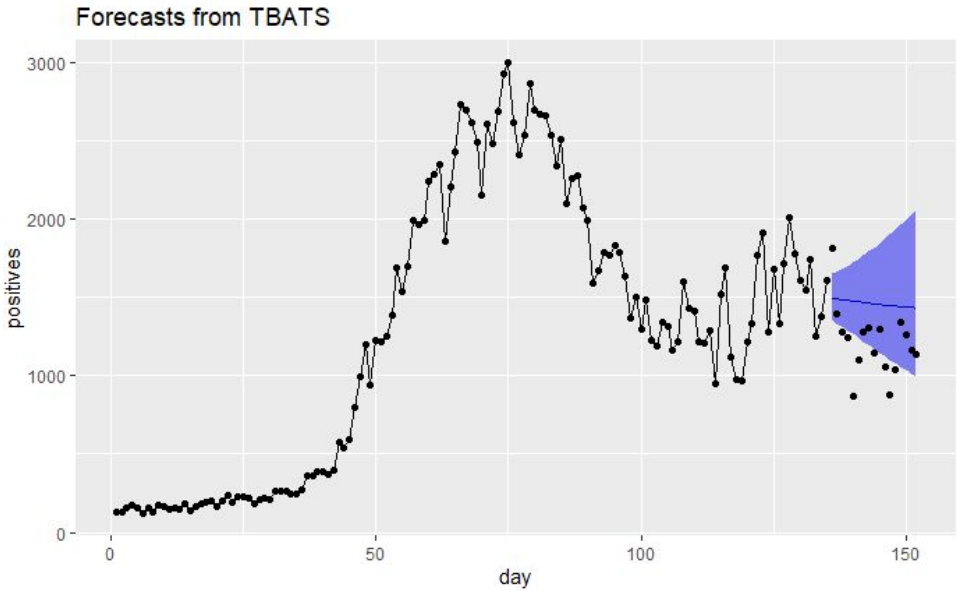


Mean Absolute Percentage Error (MAPE)
15%

Other models that we tried

R forecast package:





Recap on prediction models

- The negative binomial GAM performs the best for small prediction length
- Random forest allows to predict on longer intervals

Model	Prediction days	MAPE
Negative Binomial GAM	10	8.7%
Random Forest	10	18%
Random Forest	15	17%
Random Forest	20	15%

References

- Frank, R.J., Davey, N. & Hunt, S.P. Time Series Prediction and Neural Networks. *Journal of Intelligent and Robotic Systems* **31,** 91–103 (2001). https://doi.org/10.1023/A:1012074215150