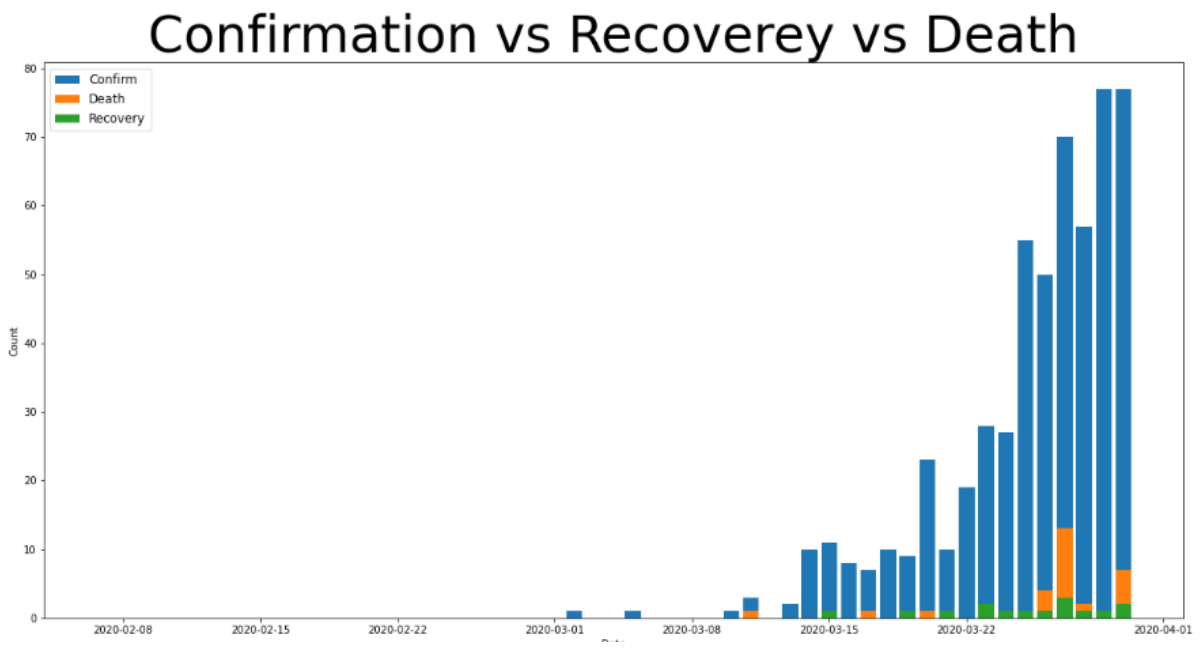
Prediction of Covid-19 (Morroco)

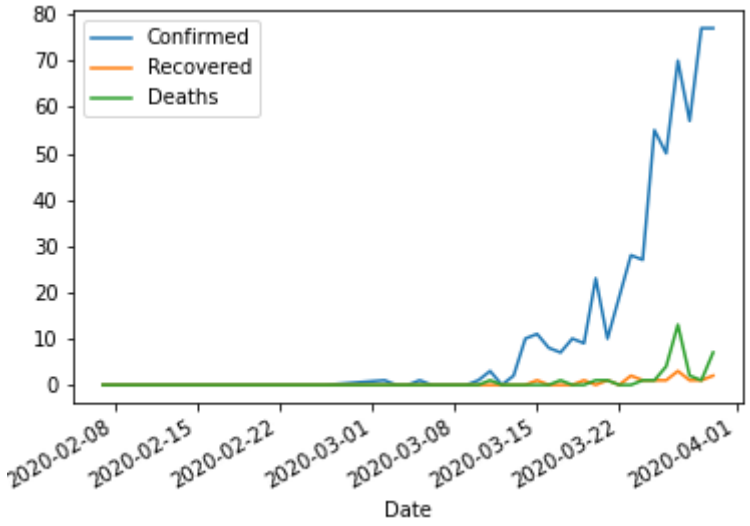
Database :

We worked on a database, updated at 6 p.m. according to the news from the Minister of Health.

	A	B	C	D	E	F	G	H	I
1	Jour	Cas testés négatifs	Cas testés positifs	Rétablis	Morts	Cas testés	% Patients	Cumul testes	Cumul Patients
2	07/02/2020	9	0	0	0	9	0,00%	9	0
3	15/02/2020	1	0	0	0	1	0,00%	10	0
4	26/02/2020	7	0	0	0	7	0,00%	17	0
5	02/03/2020	11	1	0	0	12	8,33%	29	1
6	03/03/2020	4	0	0	0	4	0,00%	33	1
7	04/03/2020	2	0	0	0	2	0,00%	35	1
8	05/03/2020	6	1	0	0	7	14,29%	42	2
9	06/03/2020	10	0	0	0	10	0,00%	52	2
10	07/03/2020	5	0	0	0	5	0,00%	57	2
11	09/03/2020	5	0	0	0	5	0,00%	62	2
12	10/03/2020	3	1	0	0	4	25,00%	66	3
13	11/03/2020	15	3	0	1	18	16,67%	84	6
14	12/03/2020	13	0	0	0	13	0,00%	97	6
15	13/03/2020	15	2	0	0	17	11,76%	114	8
16	14/03/2020	6	10	0	0	16	62,50%	130	18
17	15/03/2020	23	11	1	0	34	32,35%	164	29
18	16/03/2020	17	8	0	0	25	32,00%	189	37

Visualization :

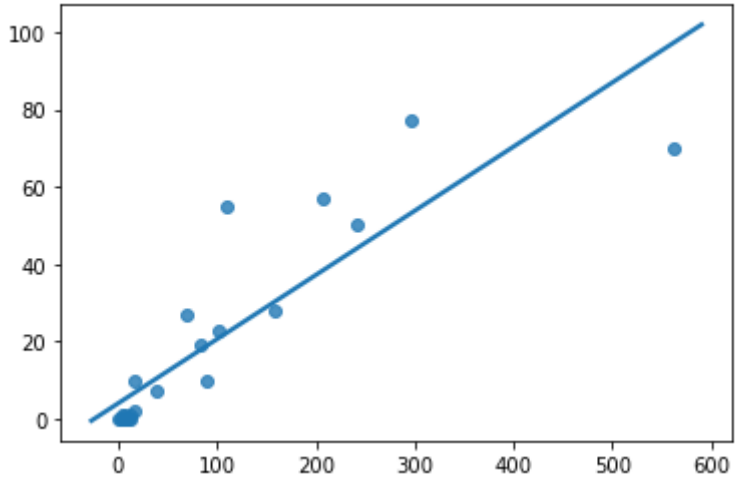




Prediction :

Simple Linear Regression

Simple linear regression is an approach for predicting a response using a single feature. It is assumed that the two variables are linearly related. Hence, we try to find a linear function that predicts the response value(Cas testés positifs) as accurately as possible as a function of the feature or independent variable(Cas testés).

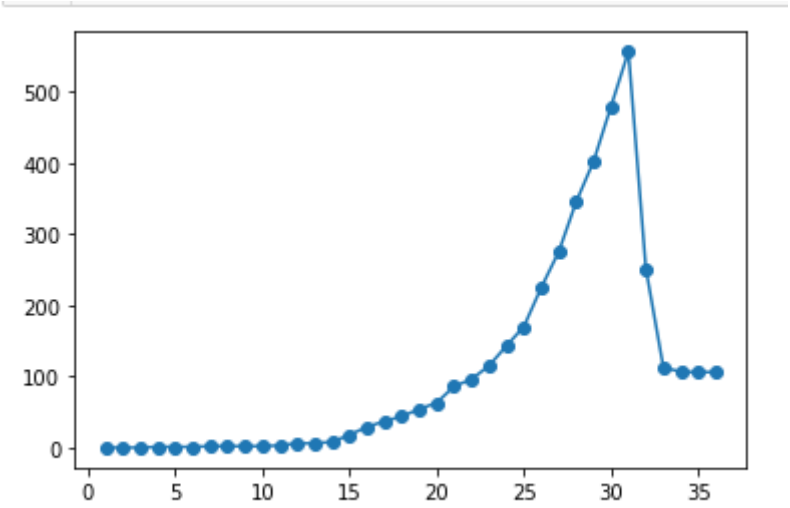


As we see in the graph, if we add the tested cases then the positive cases also increases. According to the equation : $y = 0.16611057 * x + 3.96449102$

Support Vector Machine :

An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. In addition to performing linear classification, SVMs can efficiently perform a non-linear classification, implicitly mapping their inputs into high-dimensional feature spaces.

⇨ We use this method here to predict positive test cases for the next 5 days.



From the given data, it looks like the death rate of the virus might reduce by 31st March which might not be a real estimation but If we have more data points then we can predict a bit better.

ARIMA :

After having executed the command auto_arima to have the parameters of the model.

Out[31]:

ARIMA Model Results

Dep. Variable:	D2.y	No. Observations:	29
Model:	ARIMA(0, 2, 4)	Log Likelihood	-95.609
Method:	css-mle	S.D. of innovations	5.576
Date:	Tue, 31 Mar 2020	AIC	203.217
Time:	15:08:54	BIC	211.421
Sample:	2	HQIC	205.787

	coef	std err	z	P>|z|	[0.025	0.975]
const	2.5120	1.483	1.694	0.090	-0.394	5.418
ma.L1.D2.y	-0.8384	nan	nan	nan	nan	nan
ma.L2.D2.y	1.1977	nan	nan	nan	nan	nan
ma.L3.D2.y	-0.8384	nan	nan	nan	nan	nan
ma.L4.D2.y	1.0000	nan	nan	nan	nan	nan

Roots

	Real	Imaginary	Modulus	Frequency
MA.1	-0.2849	-0.9586j	1.0000	-0.2960
MA.2	-0.2849	+0.9586j	1.0000	0.2960
MA.3	0.7041	-0.7101j	1.0000	-0.1257
MA.4	0.7041	+0.7101j	1.0000	0.1257

⇨ The results show that the cumulative total for March 31st would be 641, which means an addition of 84 cases tested positive today

Entrée [33]:

```
forcast = fit_model.forecast(steps=7)
pred_y = forcast[0].tolist()
pd.DataFrame(pred_y)
```

Out[33]:

	0
0	641.979765
1	727.653330
2	805.367129
3	892.194288
4	981.533433
5	1073.384562
6	1167.747677

LSTM:

LSTMs can be used to model problems in forecasting univariate time series.
These are problems composed of a single series of observations and a model is needed to learn from the series of past observations in order to predict the next value in the sequence.

And here, we are going to try to Predict the cumulation of positive cases in Morocco.

Entrée [44]:

```
scaler = MinMaxScaler()
scaler.fit(train_data)
scaled_train_data = scaler.transform(train_data)
scaled_test_data = scaler.transform(test_data)
n_input =5
n_features =1

generator = TimeseriesGenerator(scaled_train_data,scaled_train_data, length=n_input, batch_size=1)

lstm_model = Sequential()
lstm_model.add(LSTM(units = 50, return_sequences = True, input_shape = (n_input, n_features)))
lstm_model.add(Dropout(0.2))
lstm_model.add(LSTM(units = 50, return_sequences = True))
lstm_model.add(Dropout(0.2))
lstm_model.add(LSTM(units = 50))
lstm_model.add(Dropout(0.2))
lstm_model.add(Dense(units = 1))
lstm_model.compile(optimizer = 'adam', loss = 'mean_squared_error')
lstm_model.fit_generator( generator,epochs = 30)
```

```
Epoch 1/30
16/16 [==============================] - 6s 379ms/step - loss: 0.1401
Epoch 2/30
16/16 [==============================] - 0s 18ms/step - loss: 0.0730
Epoch 3/30
16/16 [==============================] - 0s 19ms/step - loss: 0.0663
Epoch 4/30
16/16 [==============================] - 0s 20ms/step - loss: 0.0621
Epoch 5/30
16/16 [==============================] - ETA: 0s - loss: 0.048 - 0s 19ms/step - loss: 0.0431
Epoch 6/30
16/16 [==============================] - 0s 19ms/step - loss: 0.0109
Epoch 7/30
16/16 [==============================] - 0s 18ms/step - loss: 0.0046
Epoch 8/30
16/16 [==============================] - 0s 18ms/step - loss: 0.0032
Epoch 9/30
16/16 [==============================] - 0s 18ms/step - loss: 0.0091
Epoch 10/30
```

The photo below gives us the predictions of the 4 days that follow for example March 31, 2020, the cumulative number of positive cases will be 302. Which is not fair because of the number of observations.

```
Entrée [48]:
1 prediction = pd.DataFrame(scaler.inverse_transform(lstm_predictions_scaled))
2 prediction
```

Out[48]:

	0
0	247.182697
1	301.829502
2	364.602821
3	428.321316
4	493.388039

Prophet Algorithm :

What is Prophet? Prophet is a facebooks' open source time series prediction. Prophet decomposes time series into trend, seasonality and holiday. It has intuitive hyper parameters which are easy to tune.

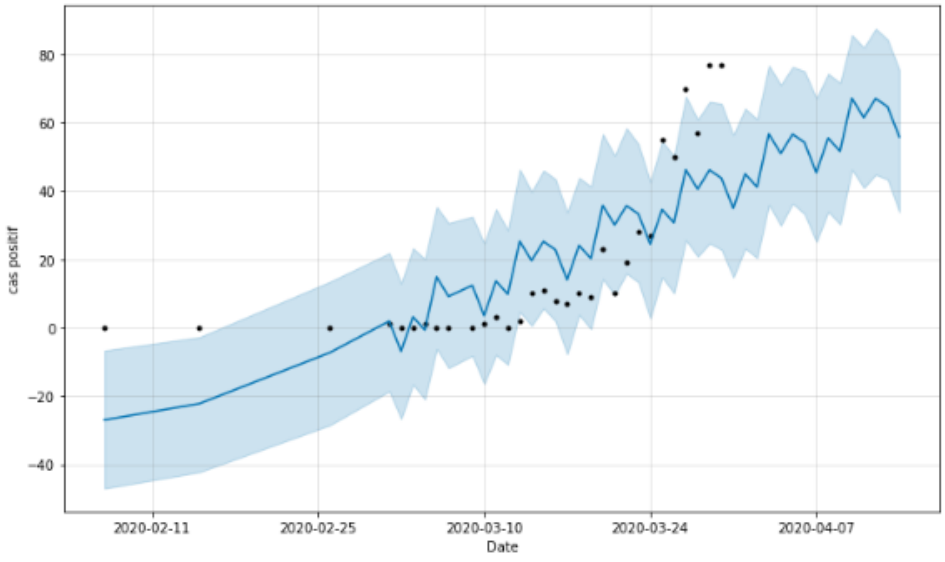
For this method we proceed as follows:

- Prediction of positive test cases
- Prediction of the dead
- Predicted recoveries

This automatic library gives us the following results for March 31:

- Positive cases: 35
- The restored: 1
- The dead: 3

For the results it is as follows, we will show the prediction of the positive tested cases .



Conculsion :

We will update the data in the database at 6 p.m. as usual and we will test our predictions.