Dengue Supervised Learning - Machine Learning

EDUARDO GARCÍA APARICIO, LUCÍA ALFONSO GARCÍA, and MANUEL VILLALBA MONTALBÁN

ACM Reference Format:

Eduardo García Aparicio, Lucía Alfonso García, and Manuel Villalba Montalbán. 2021. Dengue Supervised Learning - Machine Learning. 1, 1 (January 2021), 11 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

GitHub Repository

1 INTRODUCTION AND PECULIARITIES OF THE PROBLEM

"Your goal is to predict the total_cases label for each (city, year, weekofyear) in the test set. There are two cities, San Juan and Iquitos, with test data for each city spanning 5 and 3 years respectively. You will make one submission that contains predictions for both cities. The data for each city have been concatenated along with a city column indicating the source: sj for San Juan and iq for Iquitos. The test set is a pure future hold-out, meaning the test data are sequential and non-overlapping with any of the training data. Throughout, missing values have been filled as NaNs."[1]

We can classify this problem as a regression problem with time series. The greatest difficulty of the competition is that we don´t have access to the values we have to predict, also the prediction we have to do is on long term (5 years San Juan and 3 years Iquitos) so the problem is even more demanding.

The problem has time series, we have had to maintain an strict order while we have been working with this dataset because **we can not make predictions with future data**, so **shuffle methods** are already **discarded** for this problem.

For this reason, we payed close attention to the benchmark walkthrough, with the aim of having a good baseline.[2] In the benchmark, we must **highlight two clues** that they have been the **mains objectives** of our group in this project:

- *"However, the timing of the seasonality of our predictions has a mismatch with the actual results. One potential reason for this is that our features don't look far enough into the past–that is to say, we are asking to predict cases at the same time as we are measuring percipitation. Because dengue is misquito born, and the misquito lifecycle depends on water, we need to take both the life of a misquito and the time between infection and symptoms into account when modeling dengue. This is a critical avenue to explore when improving this model."* [2]

Authors' address: Eduardo García Aparicio; Lucía Alfonso García; Manuel Villalba Montalbán.

Manuscript submitted to ACM

This aspect may seem obvious and trivial at first, but it is not. We have to know what **impact each feature has** and most important, **when** it has a higher weight. For this reason it is critical discover the **delay between features and dengue cases**, and at the same time which of those are more **useful**.

- *"The other important error is that our predictions are relatively consistent–we miss the spikes that are large outbreaks. One reason is that we don't take into account the contagiousness of dengue. A possible way to account for this is to build a model that progressively predicts a new value while taking into account the previous prediction. By training on the dengue outbreaks and then using the predicted number of patients in the week before, we can start to model this time dependence that the current model misses."* [2]

Obtain the seasonality from the dataset is quite straightfoward. The problem is the outbreaks spikes. We believe that this is the most difficult part to predict because we have to consider previous cases, very strenuous task when we don´t have the label from tests.In this document we will see some approaches to solve this problem.

2 RESEARCH AND BASELINE

2.1 Dengue Fever

We started our research searching some useful information about dengue fever and its spread. A female mosquito that takes a blood meal from a person infected with dengue fever, during the initial 2- to 10-day febrile period, becomes itself infected with the virus in the cells lining its gut, so mosquitoes does not born with the disease, it has to bite an infected human first. About 8–10 days later, the virus spreads to other tissues including the mosquito's salivary glands and is subsequently released into its saliva. Mosquitoes will have the disease their whole life.

In humans, there is an initial 4-7 days incubation period. 80 % of people infected are asymptomatic.

So we estimate that the cycle of dengue is 2-5 weeks long

2.2 Feature engineering

To see the importance of the characteristics, the first thing we do is separate the data from both cities, since each one behaves differently and see which characteristics are related.

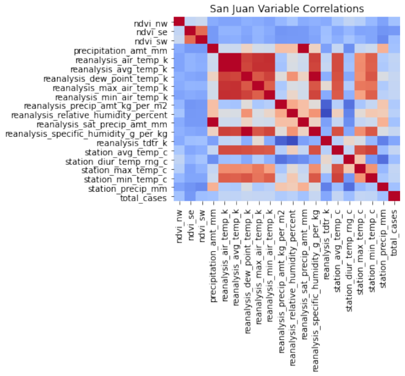


Fig. 1. San Juan Correlation

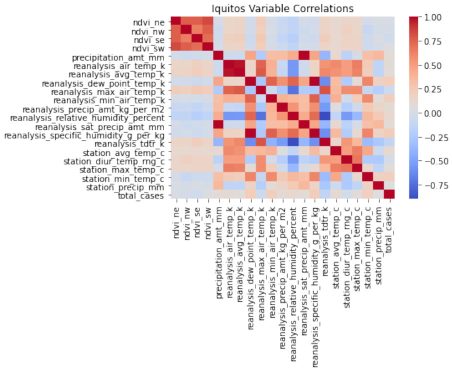


Fig. 2. Iquitos Correlation

As we can see, there is no variable that is directly related to the total cases of any of the cities, although there are some that are related to each other.

After analyzing this and seeing some differences between San Juan and Iquitos, we decided to investigate the mosquitoes that transmit Dengue and the climates that could favor their reproduction.

If we analyzed the cycle of life of the **mosquito of the Dengue**[3][5] , we saw that it can put hundreds of eggs in a day, which **can survive several months** without hatching waiting to be submerged under the water, once this happens, they become larva and they develop in a process that takes around 5 to 10 days, later it is transformed into pupa and it is maintained thus by about 3 days until finally it is transformed into a totally developed mosquito, which takes 3 days in leaving, depending on the temperature and can **live** between **1 to 2 months**.

If we look at the climate[4]of the cities, in **San Juan** we see that it concentrates most of the rain in **May, September and October**, so this could be a reason why most of the cases in this city occurred at the end of the year. However, in **Iquitos** the rains are not concentrated at any specific time, which is why we believe that the cases in this city do not have as many peaks. Precipitation and rainfall influence a greater availability of hatcheries and a greater frequency of feeding in water-stressed conditions.

That's why we came to the conclusion that the weather of a few weeks ago is the most influential in the cases of now, since we have to wait for the mosquito to be born after the rain, so we made a 'shift' with the characteristics. In order not to make a too high shift, we decided to set the limit of weeks to 18, being the characteristics modified as follows:

With which we get the following correlations: 3 5

As we can see, the correlation between the different variables has improved a little, although with the total cases it has not improved much. Even so, since we do not see that it has gotten worse either, we decided to continue with the datasets with shift.

2.3 Time Series Forescasting

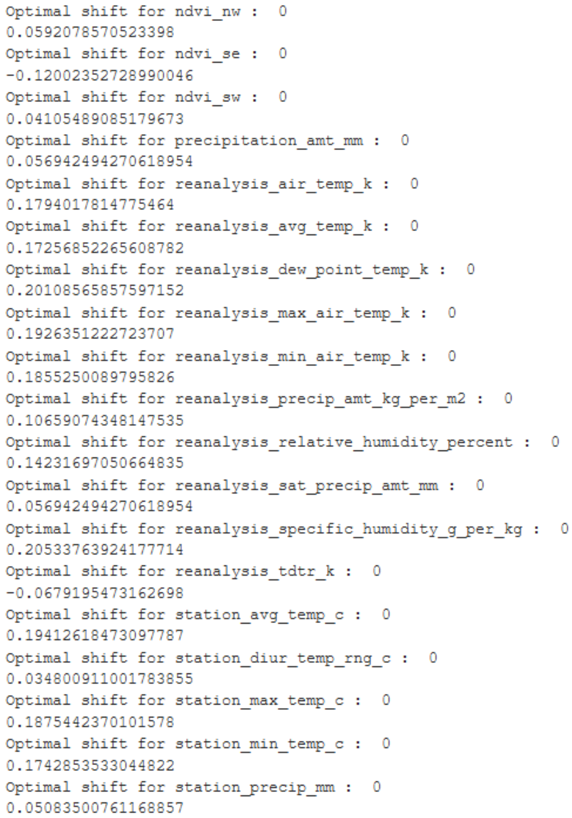
In order to achive the second goal (predict the outbreaks spikes), we bet and invest too much time studying Time Series Forecasting, we even read the book **Time Series Forescasting**[6]. This was in vain due to the **nature of the problem**. We can not apply autoregressive or ARIMA models if you are not able to update frequently your model with real data , even less if you do not have any of that data. So after learning the lesson in a hard way, we studied other techniques and models explained in the next points.

2.4 Baseline

Our baseline was the datadriven's benchmark[2]. We tried to improve it, changing some shift in the features. We obtained **24.7188 MAE** when we set a **shift** on all features of 6 weeks in the city of San Juan, and 2 weeks in Iquitos, so we improved the example at the beginning.

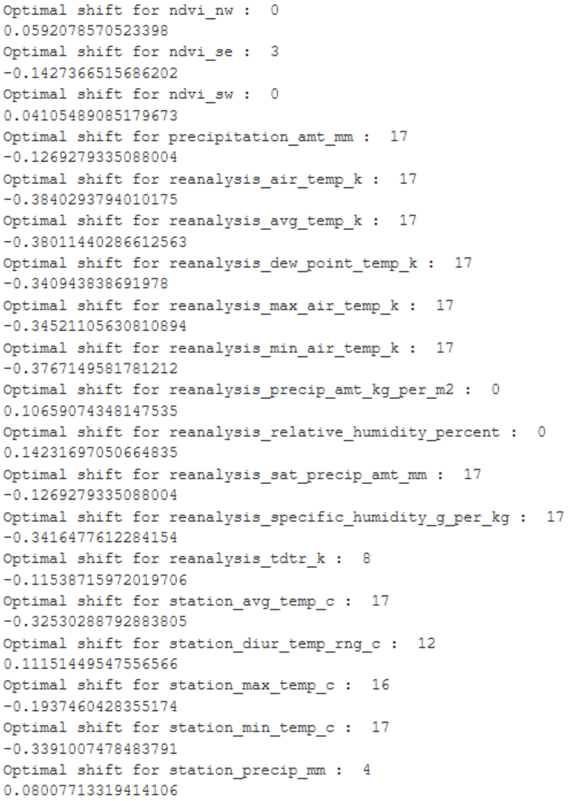
3 MODELS AND WORKFLOWS

After doing feature engineering and selecting in each feature the shift with better correlation with total_cases. We begin a search in width, looking for a model that gave us much better results or at least a noticeable improvement.



```
Optimal shift for ndvi_nw :  0
0.0592078570523398
Optimal shift for ndvi_se :  0
-0.12002352728990046
Optimal shift for ndvi_sw :  0
0.04105489085179673
Optimal shift for precipitation_amt_mm :  0
0.056942494270618954
Optimal shift for reanalysis_air_temp_k :  0
0.1794017814775464
Optimal shift for reanalysis_avg_temp_k :  0
0.17256852265608782
Optimal shift for reanalysis_dew_point_temp_k :  0
0.20108565857597152
Optimal shift for reanalysis_max_air_temp_k :  0
0.1926351222723707
Optimal shift for reanalysis_min_air_temp_k :  0
0.1855250089795826
Optimal shift for reanalysis_precip_amt_kg_per_m2 :  0
0.10659074348147535
Optimal shift for reanalysis_relative_humidity_percent :  0
0.14231697050664835
Optimal shift for reanalysis_sat_precip_amt_mm :  0
0.056942494270618954
Optimal shift for reanalysis_specific_humidity_g_per_kg :  0
0.20533763924177714
Optimal shift for reanalysis_tdtr_k :  0
-0.0679195473162698
Optimal shift for station_avg_temp_c :  0
0.19412618473097787
Optimal shift for station_diur_temp_rng_c :  0
0.034800911001783855
Optimal shift for station_max_temp_c :  0
0.1875442370101578
Optimal shift for station_min_temp_c :  0
0.1742853533044822
Optimal shift for station_precip_mm :  0
0.05083500761168857
```

Fig. 3. San Juan Correlation



```
Optimal shift for ndvi_nw :  0
0.0592078570523398
Optimal shift for ndvi_se :  3
-0.1427366515686202
Optimal shift for ndvi_sw :  0
0.04105489085179673
Optimal shift for precipitation_amt_mm :  17
-0.1269279335088004
Optimal shift for reanalysis_air_temp_k :  17
-0.3840293794010175
Optimal shift for reanalysis_avg_temp_k :  17
-0.38011440286612563
Optimal shift for reanalysis_dew_point_temp_k :  17
-0.340943838691978
Optimal shift for reanalysis_max_air_temp_k :  17
-0.34521105630810894
Optimal shift for reanalysis_min_air_temp_k :  17
-0.3767149581781212
Optimal shift for reanalysis_precip_amt_kg_per_m2 :  0
0.10659074348147535
Optimal shift for reanalysis_relative_humidity_percent :  0
0.14231697050664835
Optimal shift for reanalysis_sat_precip_amt_mm :  17
-0.1269279335088004
Optimal shift for reanalysis_specific_humidity_g_per_kg :  17
-0.3416477612284154
Optimal shift for reanalysis_tdtr_k :  8
-0.11538715972019706
Optimal shift for station_avg_temp_c :  17
-0.32530288792883805
Optimal shift for station_diur_temp_rng_c :  12
0.11151449547556566
Optimal shift for station_max_temp_c :  16
-0.1937460428355174
Optimal shift for station_min_temp_c :  17
-0.3391007478483791
Optimal shift for station_precip_mm :  4
0.08007713319414106
```

Fig. 4. San Juan Correlation with shift

Next we will explain the different models we have used to generate our predictions as well as their results in the competition.

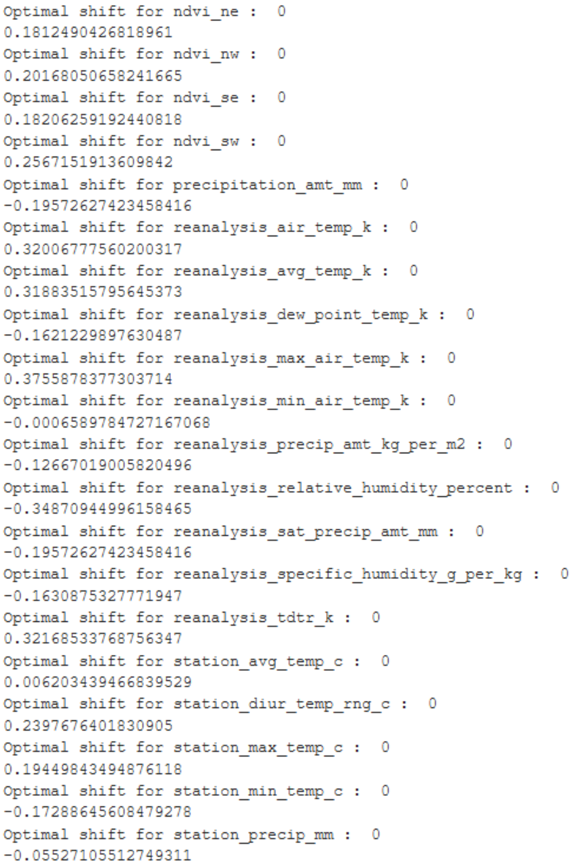
When training these models, the shift was disabled and the features we used were all of them, so it was not fully optimized. The only thing we modified were certain parameters such as the maximum depth or the criteria.

3.1 Decision Tree

This algorithm was the first we tested with and, because of that, it was the one we were testing the most with different configurations and values for its hyperparameters, which never had an MAE of less than **27**. This algorithm was later discarded due to the next drawbacks we found:

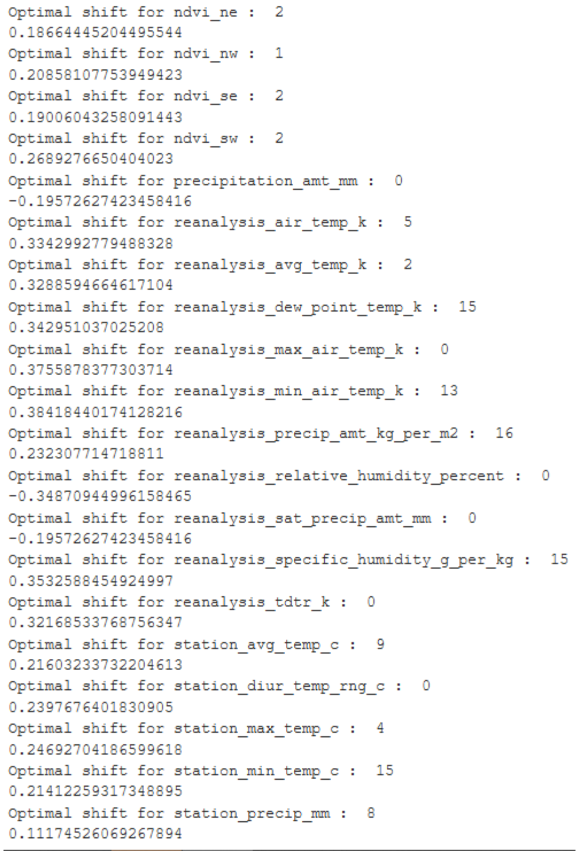
- A small change in the data can cause a large change in the structure of the decision tree causing instability, and as far as we know, diseases are not predictable and have a very erratic behaviour.
- This algorithm is inadequate for applying regression and predicting continuous values.

Results:



```
Optimal shift for ndvi_ne :  0
0.1812490426818961
Optimal shift for ndvi_nw :  0
0.20168050658241665
Optimal shift for ndvi_se :  0
0.18206259192440818
Optimal shift for ndvi_sw :  0
0.2567151913609842
Optimal shift for precipitation_amt_mm :  0
-0.19572627423458416
Optimal shift for reanalysis_air_temp_k :  0
0.32006777560200317
Optimal shift for reanalysis_avg_temp_k :  0
0.31883515795645373
Optimal shift for reanalysis_dew_point_temp_k :  0
-0.1621229897630487
Optimal shift for reanalysis_max_air_temp_k :  0
0.3755878377303714
Optimal shift for reanalysis_min_air_temp_k :  0
-0.0006589784727167068
Optimal shift for reanalysis_precip_amt_kg_per_m2 :  0
-0.12667019005820496
Optimal shift for reanalysis_relative_humidity_percent :  0
-0.34870944996158465
Optimal shift for reanalysis_sat_precip_amt_mm :  0
-0.19572627423458416
Optimal shift for reanalysis_specific_humidity_g_per_kg :  0
-0.1630875327771947
Optimal shift for reanalysis_tdtr_k :  0
0.32168533768756347
Optimal shift for station_avg_temp_c :  0
0.006203439466839529
Optimal shift for station_diur_temp_rng_c :  0
0.2397676401830905
Optimal shift for station_max_temp_c :  0
0.19449843494876118
Optimal shift for station_min_temp_c :  0
-0.17288645608479278
Optimal shift for station_precip_mm :  0
-0.05527105512749311
```

Fig. 5. Iquitos Correlation



```
Optimal shift for ndvi_ne :  2
0.18664445204495544
Optimal shift for ndvi_nw :  1
0.20858107753949423
Optimal shift for ndvi_se :  2
0.19006043258091443
Optimal shift for ndvi_sw :  2
0.2689276650404023
Optimal shift for precipitation_amt_mm :  0
-0.19572627423458416
Optimal shift for reanalysis_air_temp_k :  5
0.3342992779488328
Optimal shift for reanalysis_avg_temp_k :  2
0.3288594664617104
Optimal shift for reanalysis_dew_point_temp_k :  15
0.342951037025208
Optimal shift for reanalysis_max_air_temp_k :  0
0.3755878377303714
Optimal shift for reanalysis_min_air_temp_k :  13
0.38418440174128216
Optimal shift for reanalysis_precip_amt_kg_per_m2 :  16
0.232307714718811
Optimal shift for reanalysis_relative_humidity_percent :  0
-0.34870944996158465
Optimal shift for reanalysis_sat_precip_amt_mm :  0
-0.19572627423458416
Optimal shift for reanalysis_specific_humidity_g_per_kg :  15
0.3532588454924997
Optimal shift for reanalysis_tdtr_k :  0
0.32168533768756347
Optimal shift for station_avg_temp_c :  9
0.21603233732204613
Optimal shift for station_diur_temp_rng_c :  0
0.2397676401830905
Optimal shift for station_max_temp_c :  4
0.24692704186599618
Optimal shift for station_min_temp_c :  15
0.21412259317348895
Optimal shift for station_precip_mm :  8
0.11174526069267894
```

Fig. 6. Iquitos Correlation with shift

- San Juan Parameters: max_depth=4, criterion=MAE
- Iquitos Parameters: max_depth=4, criterion=MAE
- Total MAE: 29.4792

3.2 KNeighborsRegressor

We used the dataset after doing feature engineering (specifics shifts in each feature.)

When we tesed KNeighborsRegressor we used an hyperparamaters fuction in order to obtain a good number of neighbour (k with low MAE in subtesting) for each city and also determinate if it was better uniform or distance weights in each case.

The results in the competition were a **MAE between 30 and 26**, they were really poor, so we also tried changing others parameters like the algorithm used to compute the nearest neighbors (‘auto’, ‘ball_tree’, ‘kd_tree’, ‘brute’) or the metric (“euclidean”, “manhattan”, “chebyshev”) without any improvement.

3.3 Random Forest

The next algorithm was Random Forest, an assembly made up of a multitude of decision trees. The importance of Random Forest for us is not whether it was good or bad predicting cases of dengue in Iquitos or San Juan, but because it showed the importance of each feature we could be able to use those same features to train other models and get better results.

Results:

- San Juan Parameters: n_estimators=1000, max_depth=5, criterion=MAE
- Iquitos Parameters: n_estimators=1000, max_depth=5, criterion=MAE
- Total MAE: 27.4871

3.4 Ada Boost

Following the decision tree line, we also wanted to test Ada Boost with the default parameters and where the instance weights are adjusted based on the current prediction error. We chose this method as it allowed us to improve the accuracy of our regressor hence making it flexible and because theoretically is not prone to overfitting, though there is no precise proof for this.

Results:

- San Juan Parameters: n_estimators=150, max_depth=9, criterion=MAE
- Iquitos Parameters: n_estimators=300, max_depth=4, criterion=MAE
- Total MAE: 26.9203

3.5 Stochastic gradient descent

In the line of boosting algorithms, we also found that Stochastic could be useful, since we want to find the best parameters for the data to minimize loss and maximize accuracy. One of the drawbacks is that Stochastic gradient descent uses a common learning rate for all parameters and for problems with a large number of parameters, this could be problematical. We didn't know if we were going to have this kind of problem in our case so we tried, but the results were not good probably because of what we have said before about the SGD and its difficulties when dealing with many features, so we continued our research.

Results:

- San Juan Parameters: n_estimators=150, max_depth=5, criterion=MAE
- Iquitos Parameters: n_estimators=300, max_depth=4, criterion=MAE
- Total MAE: 27.7291

3.6 FaceBook's Prophet

At that moment, we wanted to get away from the scikit library and try new models. We found that Facebook had developed a model for time series, so we decided to test it. The results were normal, a MAE about 26, more or less the same as the results of the other models.

Results:

- San Juan Parameters: trend flexibility=0.32 , yearly_seasonality=4 ,weekly_seasonality=auto
- Iquitos Parameters: trend flexibility=0.32 , yearly_seasonality=4 ,weekly_seasonality=4
- Total MAE: 25.2439

3.7 Voting Ensemble

After all these models, we wanted to test the Voting Ensemble, with which we average the predictions of the selected models. In this part, to calculate the mean, we chose the models that gave us the best MAE and then we made the mean. The results obtained did not vary much and we expected to improve the result, but we did not achieve it so we discarded this ensemble.

Results:

- San Juan Models: Decision Tree, Random Forest
- Iquitos Models: Decision Tree, Random Forest, Ada Boost
- San Juan Parameters: Same as the ones mentioned above
- Iquitos Parameters: Same as the ones mentioned above
- MAE: 28.4820

4 INTERMEDIATE THOUGHTS

After this unsuccessful searching a good model to begin with. We started to wonder what was wrong in our approach.

Our hypothesis was that you can not surpass a MAE of 24-25 in the testing without predicting outbreaks spikes, so this was the moment when we started to study time series forecasting, in order to have previous dengue cases in consideration while doing new predictions.

After that and we finally realize that we can not use time series problem so we tried others models and approaches.

We also believe that the lack of parameterization and tuning on our models could be the cause of bad results too.

5 MODELS AND WORKFLOWS, PART 2

5.1 LSTM

In order to achieve a prediction that have in consideration previous dengue cases we implemented a LSTM model.

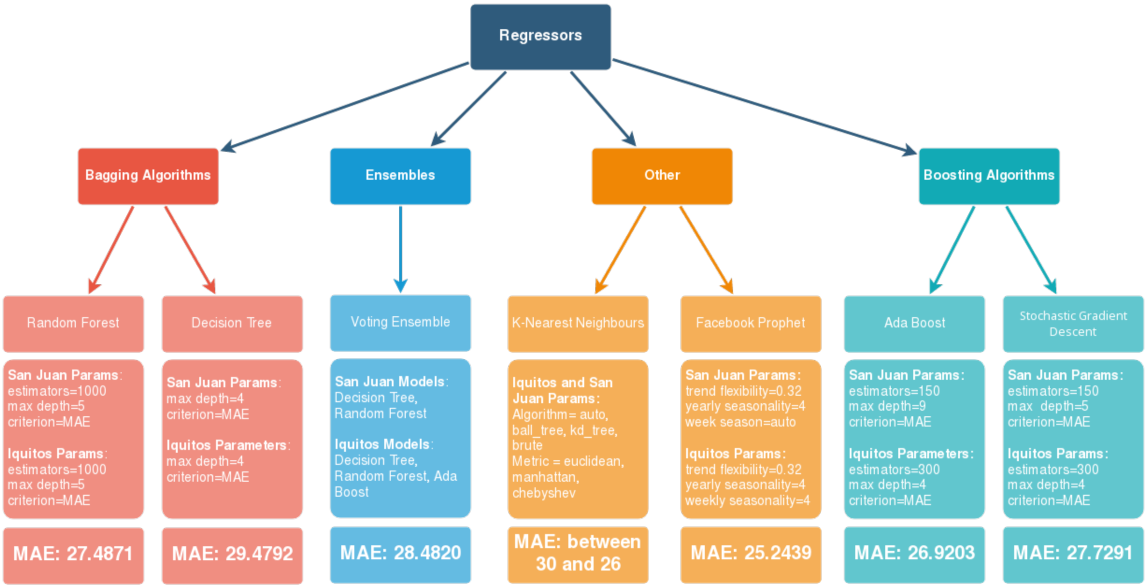

Regressors
Bagging Algorithms
Ensembles
Other
Boosting Algorithms
Random Forest
Decision Tree
Voting Ensemble
K-Nearest Neighbours
Facebook Prophet
Ada Boost
Stochastic Gradient Descent
San Juan Params: estimators=1000 max depth=5 criterion=MAE
Iquitos Params: estimators=1000 max depth=5 criterion=MAE
San Juan Params: max depth=4 criterion=MAE
Iquitos Parameters: max depth=4 criterion=MAE
San Juan Models: Decision Tree, Random Forest
Iquitos Models: Decision Tree, Random Forest, Ada Boost
Iquitos and San Juan Params: Algorithm= auto, ball_tree, kd_tree, brute Metric = euclidean, manhattan, chebyshev
San Juan Params: trend flexibility=0.32 yearly seasonality=4 week season=auto
Iquitos Params: trend flexibility=0.32 yearly seasonality=4 weekly seasonality=4
San Juan Params: estimators=150 max depth=9 criterion=MAE
Iquitos Parameters: estimators=300 max depth=4 criterion=MAE
San Juan Params: estimators=150 max depth=5 criterion=MAE
Iquitos Params: estimators=300 max depth=4 criterion=MAE
MAE: 27.4871
MAE: 29.4792
MAE: 28.4820
MAE: between 30 and 26
MAE: 25.2439
MAE: 26.9203
MAE: 27.7291

Fig. 7. List of models and their results

LSTM is an artificial recurrent neural network architecture, unlike standard feed forward neural networks, LSTM has also feedback connections, so it can not only process single data points, but also entire sequences of data. So we believed that are perfect for this task.

After some minor difficulties with the scaler, we obtained our firsts predictions with this model, and they were unsatisfactory, the results in the test were around a **MAE between 30 and 27**.

6 OPTIMIZATION AND DEFINITIVE MODELS

6.1 LSTM

We tried to setup and tunning or LSTM model, we can say that we had some improvements because we lower the **MAE error to 26.2572** changing the batch, the number of epochs and also adding the final selection of shifts in the features. We also changed the loss method to mse and added a pair of layers to reduce over fitting.

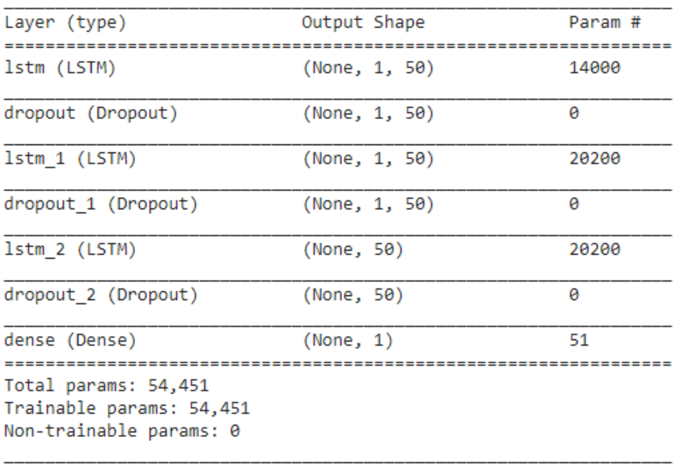
LSTM was a very difficult model for us to tune and setting up. Our final model of LSTM has been the next one: 8

We are inexperienced tunning these kind of models and the lack of time did not help us, we almost match the benchmark result so we think that it is fine for the first try.

We also tried to predict other models errors with our LSTM model but as always it did not improve nothing.

6.2 Random Forest

We have used random forest previously in this project, but this time we used hyper-parameters method to obtain the best configuration in order to achieve better results in the test. We used GridSearchCV for this task.



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
lstm (LSTM)                  (None, 1, 50)             14000     
_________________________________________________________________
dropout (Dropout)            (None, 1, 50)             0         
_________________________________________________________________
lstm_1 (LSTM)                (None, 1, 50)             20200     
_________________________________________________________________
dropout_1 (Dropout)          (None, 1, 50)             0         
_________________________________________________________________
lstm_2 (LSTM)                (None, 50)                20200     
_________________________________________________________________
dropout_2 (Dropout)          (None, 50)                0         
_________________________________________________________________
dense (Dense)                (None, 1)                 51        
=================================================================
Total params: 54,451
Trainable params: 54,451
Non-trainable params: 0
_________________________________________________________________
```

Fig. 8. LSTM model

GridSearchCV implements a fit and a score method, we used a cross-validation splitting strategy with 5-fold to avoid overfitting and also to simulate different situtations (just seasonality or outbreaks spikes).

GridSearchCV allows us to find the best paramaters in the next fields:

- criterion: Error to evaluate the testing ['mse', 'mae']
- n_estimators: Number of trees in random forest [32,64,128]
- max_features: Number of features to consider at every split [None, 'auto', 'sqrt]
- max_depth: Maximum number of levels in tree [8,4,2]
- min_samples_split: Minimum number of samples required to split a node [2, 4, 6]
- min_samples_leaf: Minimum number of samples required at each leaf node [8, 12, 16]
- boostrap: Method of selecting samples for training each tree ['True', 'False']

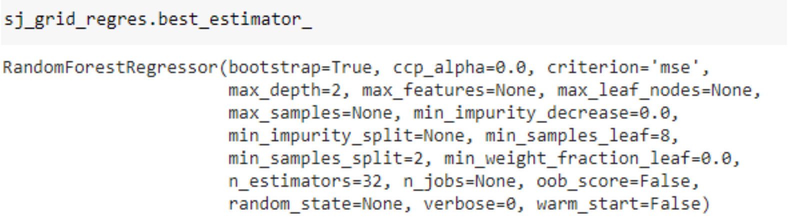
Our best result with this model has been a **MAE of 24.8293** and we can say that it has been one of our best results. The best parameters for each city has been the next ones: 9 10

7 FINAL COMPETITION RESULT AND CONCLUSIONS

Our final and best result has been **MAE of 24.8293** with Random Forest with hyper-parameters. In ELM group we are disappointed due to we expected a much better result.

7.1 Conclusions

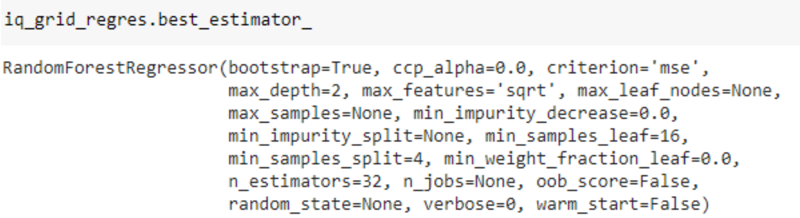
It is possible that with more organization and time, we would have obtained better results, or at least, our workflows would have been more coordinated.



```
sj_grid_regres.best_estimator_
```

```
RandomForestRegressor(bootstrap=True, ccp_alpha=0.0, criterion='mse',
                      max_depth=2, max_features=None, max_leaf_nodes=None,
                      max_samples=None, min_impurity_decrease=0.0,
                      min_impurity_split=None, min_samples_leaf=8,
                      min_samples_split=2, min_weight_fraction_leaf=0.0,
                      n_estimators=32, n_jobs=None, oob_score=False,
                      random_state=None, verbose=0, warm_start=False)
```

Fig. 9. San Juan Random Forest best parameters



```
iq_grid_regres.best_estimator_
```

```
RandomForestRegressor(bootstrap=True, ccp_alpha=0.0, criterion='mse',
                      max_depth=2, max_features='sqrt', max_leaf_nodes=None,
                      max_samples=None, min_impurity_decrease=0.0,
                      min_impurity_split=None, min_samples_leaf=16,
                      min_samples_split=4, min_weight_fraction_leaf=0.0,
                      n_estimators=32, n_jobs=None, oob_score=False,
                      random_state=None, verbose=0, warm_start=False)
```

Fig. 10. Iquitos Random Forest best parameters

One of our main goals has been working smarter rather than harder, researching information and knowledge instead of trying to configure the parameters of the models randomly. It is a pity that our labor is not reflected in our final result.

7.2 Next steps

First of all, this project needs a global reorganization, because we have our work divided in 5 Google Colabs, and one good first next step could be fuse them in one unique python program. This new baseline could be very helpful to set up new workflows.

In order to obtain better results we should check everything we have done and do a classification of what is working and helping to the competition result and what is not. Starting with the feature engineering, we have seen that with some shifts there are more correlation in some features, but we think there is something more to exploit in it that we have not discover it yet.

We think that the approach of predicting everything (seasonality and spikes) with just one model is mistaken. We recommend using one model to predict just the seasonality and other one to predict the error of the previous model, maybe there is even more seasonality in the error that we could predict too.

On the other hand, we still believe that LSTM is a very promising model and it is appropriate for this problem. We do not know what we have done wrong while we have been using it, but maybe we committed a bad feature selection, used a incorrect number of epochs or batch, or we just do not how to set up its layers correctly.

If everything we have say does not achieve better results in the competition, we should change our objetives and just search for new solutions or approaches.

REFERENCES

[1] Driven Data Problem description

[2] Driven Data benchmark walkthrough

[3] Aedes aegypti

[4] Climate

[5] Influencia de la temperatura ambiental en el mosquito Aedes spp

[6] Time Series Forecasting by Jason Brownlee