The quest for correlation between dengue and historical weather measurements

---

This document describes my effort to be competitive in predicting in the dengue forecasting competition, DengAI, hosted by DrivenData. I used the competition as a capstone project in the Professional Certificate in Data Science course hosted on EDX by HarvardX. Dengue fever is a mosquito-borne disease that occurs in tropical and sub-tropical parts of the world. Symptoms are similar to the flu in mild cases, in severe cases however, dengue fever can cause death. I followed a CRISP-DM like process: background study, analyze and prepare data, configure machine learning model, validate and submit results to leaderboard. Predictive power of time series can be increased by adding moving averages. The final model is able to predict the seasonal pattern of dengue. The conclusion proposes using ensemble modelling to add the capability to predict outbreaks.

Keywords: time series, impute, anomaly, (rolling) correlation, moving average, seasonality, forecast, prophet, data visualization, R

Problem description

The dataset is a time series about laboratory confirmed dengue cases in two cities: San Juan and Iquitos. Test data for each city spans for five and three years respectively. The test set is a pure future hold-out, meaning the test data are sequential to and non-overlapping with any of the training data. Both the training and test data contain missing values. Station meteorological readings are in Celsius while the reanalysis readings are in Celsius.

The goal of the competition is to predict the total dengue cases for each city per week in the test set. The evaluation metric is mean absolute error. My mission is to add predictive power to the data by investigating moving averages and rolling correlation between the total dengue cases and metrological variables included in the data set. I used this competition to gain experience with time series decomposition, anomaly detection (outliers) and the algorithm open sourced by Facebook for time series forecasting: Prophet.

Methods

I performed the following tasks to design and test a forecasting model that predicts dengue cases with Facebook's forecasting model Prophet.

- Study background and formulate design considerations for the forecast model.
- Perform data wrangling: load datasets, convert Kelvin to Celsius.
- Exploratory data analysis.
- Impute missing values with Kalman Smoothing.
- Feature engineering, design and train forecast model.
- Re-evaluate analysis and design.

Results background study

Epidemiology perspective:

- Dengue is endemic, it occurs every year. The forecast model will benefit from research of seasonality and anomaly's.
- Both humans and mosquitos can only transmit the virus after an incubation period. Which suggests that we need to investigate lagging values within the data to find the best correlations with the current number of dengue cases. The total dengue cases per week only contains confirmed cases. Performing lab tests and gathering the results also takes time.

Ecology and geological perspectives:

- *Aedes aegypti*, the principal mosquito of dengue viruses, is an insect closely associated with humans and our dwellings. The mosquito lays her eggs on the sides of containers with water. Eggs hatch into larvae after a rain or flooding. This indicates that total precipitation during a period could be predictive.
- It is very difficult to control or eliminate these mosquitoes because they have adaptations to the environment that make them highly resilient. Seasonality is most likely to repeat it selve in the nearby future.

- The data spans two different geological locations. One forecast model per city allows for specific business rules which probably will yield the best overall results.

Historical facts about dengue outbreaks

The test data about San Juan starts at 29-04-2008 and ends at 25-06-2013. An online blog, see references, described two outbreaks in Puerto Rice during this timeframe:

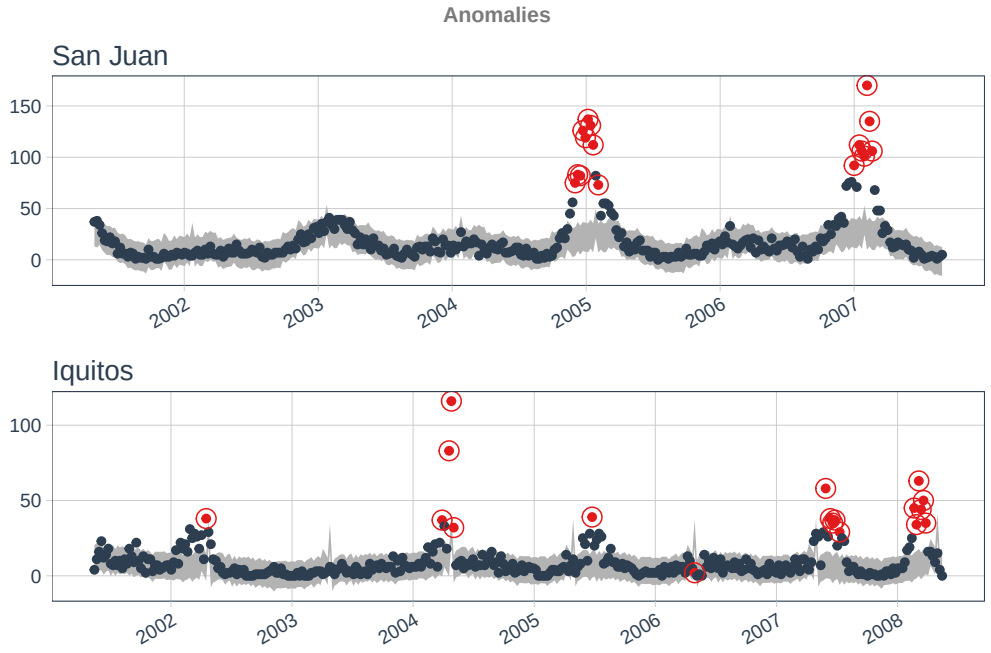
- Epidemic declared 26 February 2010 that lasted until 30 December 2010, and claimed 28 lives.
- Epidemic declared 08 October 2012 that lasted until 17 July 2014 (almost 2 years!).

The two different geological locations: San Juan and Iquitos

Anomalies detected with R package Anomalize

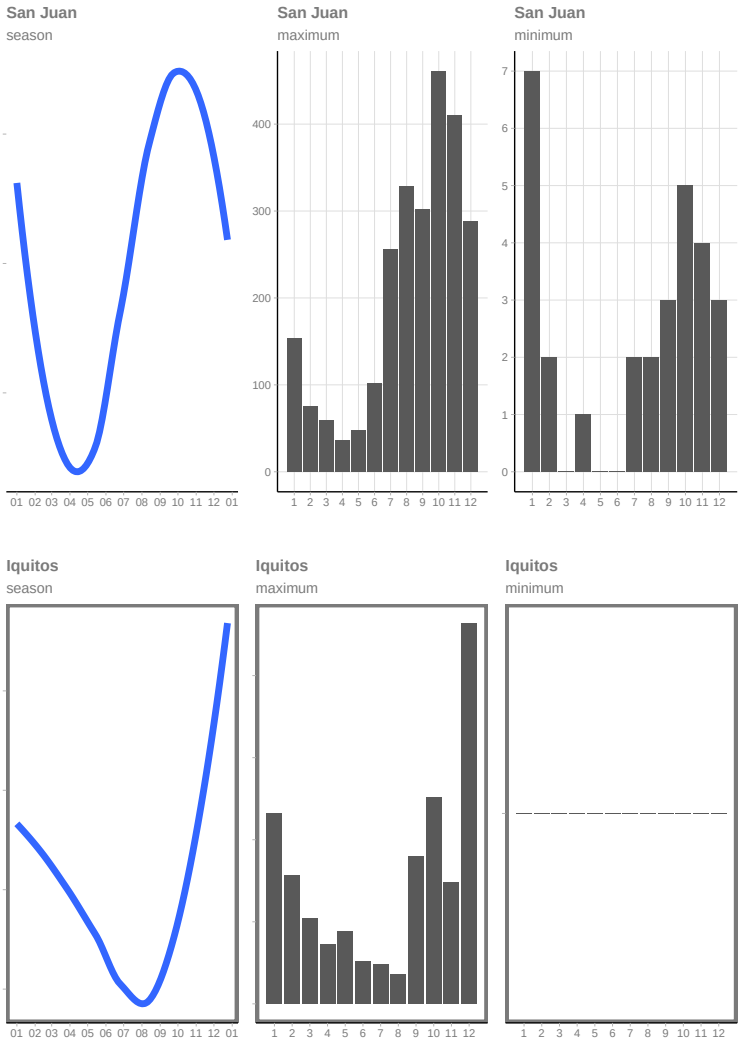
Data points that are outliers, or an exceptional event, are considered anomalies. The Anomalize package is designed for time series and contains two methods for automated anomaly detection: Inner Quartile Range (IQR) and Generalized Extreme Studentized Deviate test (GESD). Anomaly detection can be easily done on small datasets by plotting the data with boxplots, however, it becomes increasingly more difficult on large time series. Hence, I decided to give Anomalize a try and used GESD to spot anomalies. In GESD anomalies are progressively evaluated removing the worst offenders and recalculating the test statistics and critical values.

It first decomposes , divides, the subject in the time series, total dengue cases per week, into four columns that are observed, season, trend, and remainder. The anomalies are detected in the remainder, which is observed minus its season and trend components. The default method 'STL' for decomposition is used and a 12 months frequency and trend to calculate data for the plot below.



How does seasonality differ between the two cities?

Dengue cases in both cities show seasonal patterns. However, the start and the end of the dengue seasons differ between the cities. The seasonal pattern below is based on data calculated by the Anomalize package with decomposition method 'STL'.



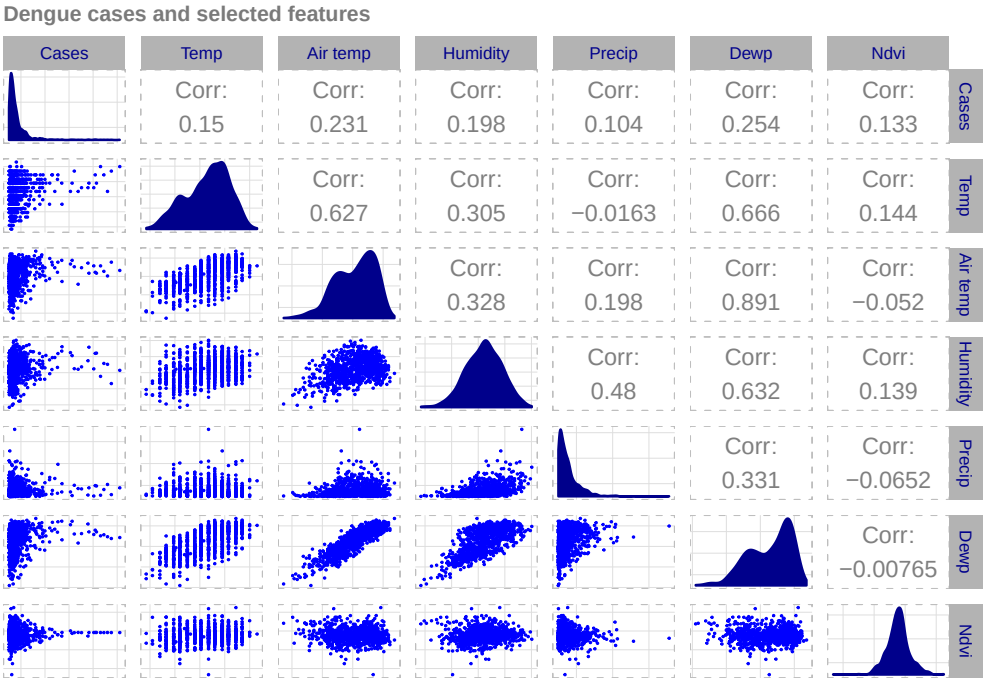
Iquitos has seen zero cases in all months, San Juan only during spring.

San Juan and the quest for correlation

Initial trial runs resulted in a more than triple mean absolute error for the forecast of San Juan compared to Iquitos. Hence, I decided to prioritize on San Juan for explorative data analysis. The data violates two Pearson correlation assumptions: it has numerous outliers and shows non-normal distributions. Therefore I assumed that correlations based on Spearman's rank-order correlation are a better fit with the data.

Correlation of total dengue cases with features

The figure below illustrates the distributions and correlations between the selected features and the total dengue cases per week.

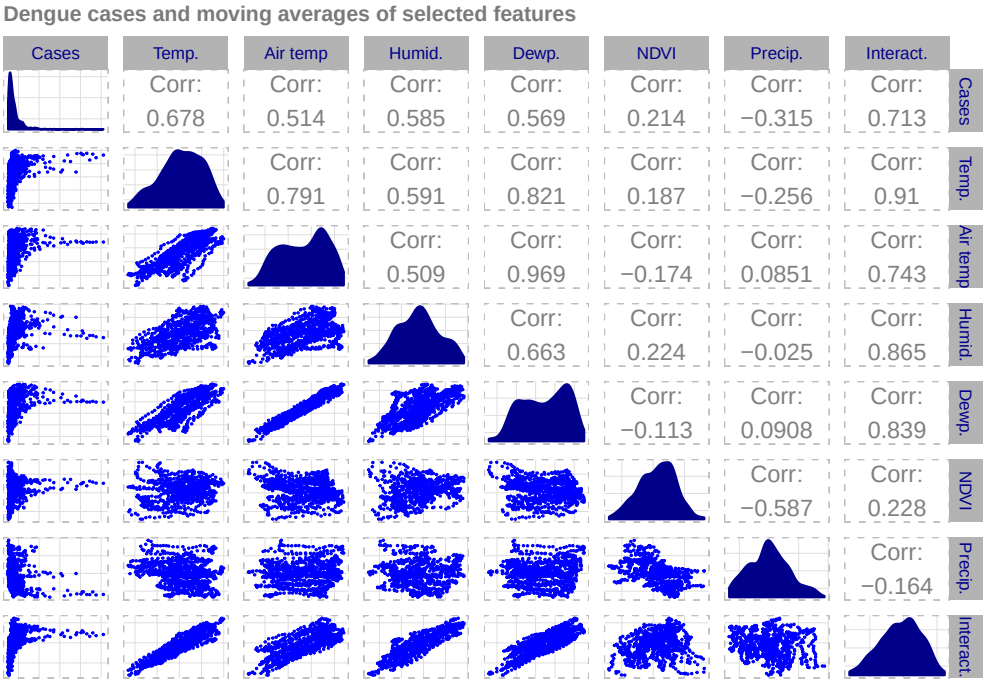


Improving correlation by adding lagging features

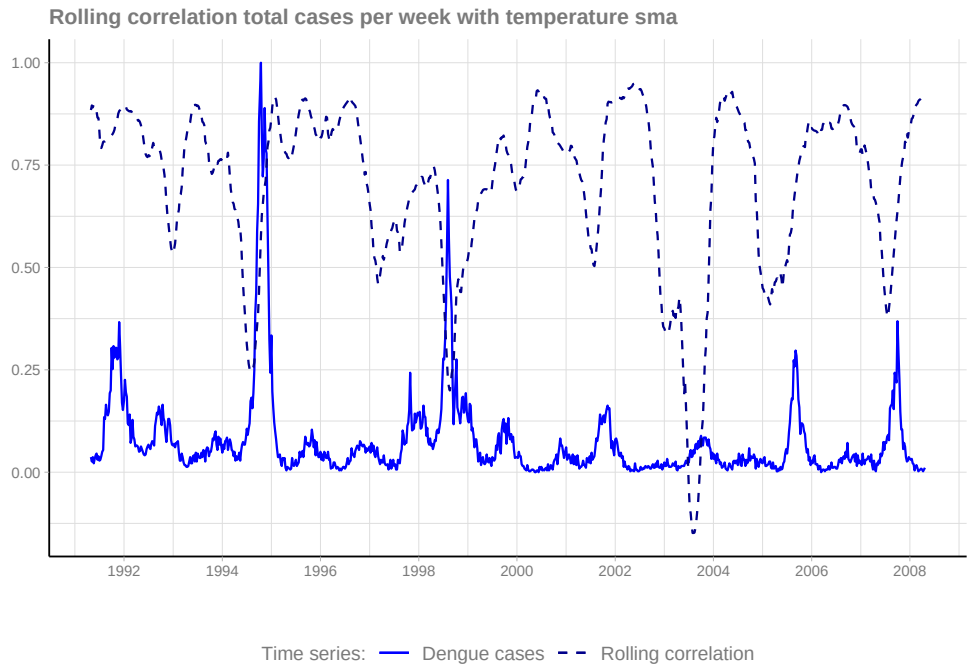
Not Surprisingly, the correlation of total dengue cases with climate features over the same week, in which the dengue cases were reported, is disappointing.

I wrote a function that adds lagging versions of the variables by iteration through look back periods starting from 1 to 25 weeks. Next I added the an interaction variable, temperature multiplied by humidity, because both variables contribute to the growth of the mosquito population. The resulting interaction variable has the best correlation with dengue cases, but can we trust this relationship to be predictive in the future? Are correlations static?

First, let's have a look at the illustration below with distributions and correlations of the moving averages added to the data. Please note that a rolling sum was used for precipitation amount in millimeters and moving averages for all others.



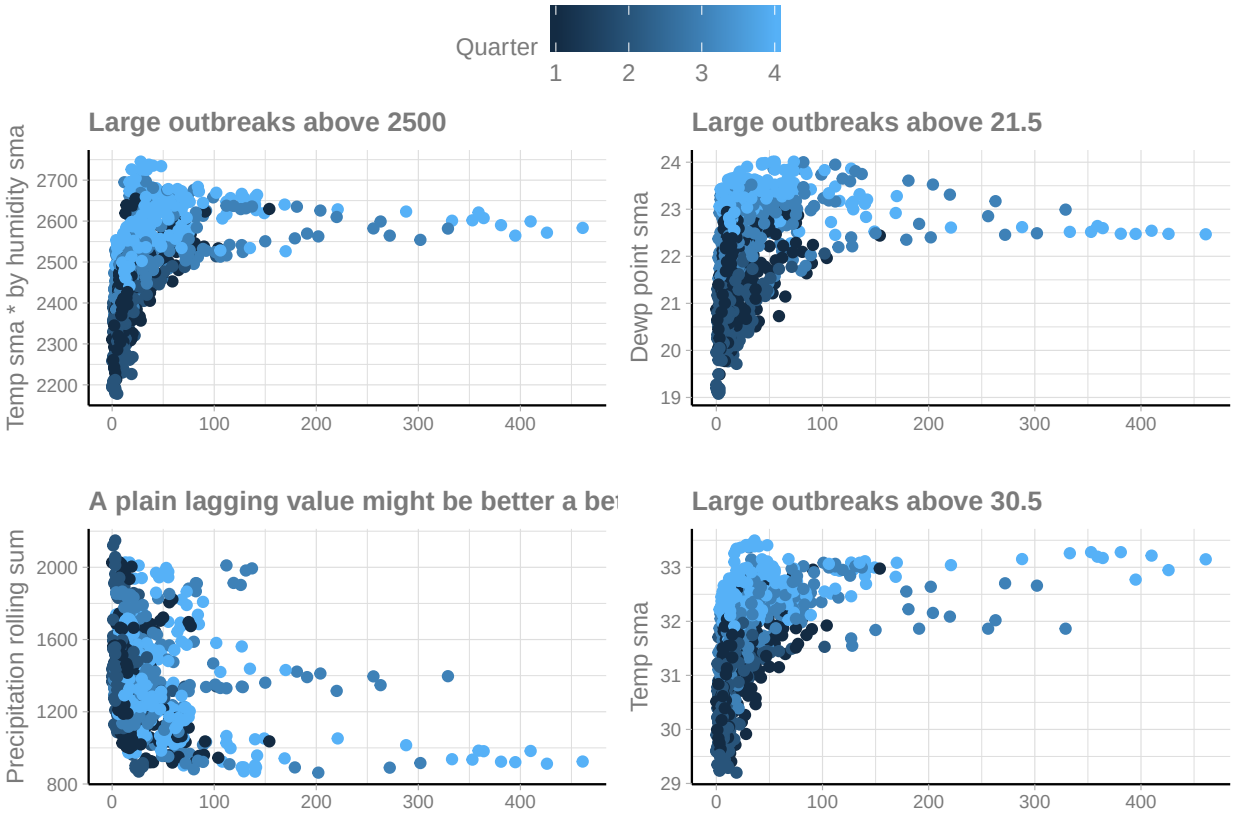
By determining the best look back period for the moving averages the correlations show a significant improvement. The interaction variable has the highest correlation. Correlations in timeseries are not static. Which is illustrated by the figure shown below.



The rolling correlation, 52 weeks look back period, of the temperature moving averages is mostly above 0.75 but potentially drops to 0.25 or even as low as 0.016. In 2003 it reverses from a positive correlation to a negative correlation with the total dengue cases.

San Juan, what patterns are revealed by scatter plots?

The scatter plots reveal distinct patterns. Unfortunately these patterns coincide with seasonal patterns: outbreaks always reach their highest level in the last quarter of the year and so do the values of the variables. The distinctive characteristics of the patterns is therefore limited. I used a rolling sum for precipitation, but the scatter plot indicates that a plain lagging value might be a better fit.



Machine learning model

About Prophet

Prophet is forecasting tool open sourced by Facebook available in Python and R. It includes an algorithm and ready to go plots for visual analysis. I choose this package in R out of curiosity and because it is said to be optimized for time series with an reasonable number of missing observations or large outliers. At its core, the Prophet procedure is an additive regression model with the following main components:

- Piecewise linear or logistic growth curve trend. Prophet automatically detects changes in trends by selecting changepoints from the data.
- Yearly seasonal component modeled using Fourier series and weekly seasonal component using dummy variables.
- A user-provided list of important holidays.

Settings

Initial trial runs showed that Prophet predicts large values below zero when using a linear growth model. I did not succeed in creating a stable model with linear models and decided to go for a logistic model which is a better fit for count data (counts of dengue cases per week). I also replaced outlier values - 1.5 * IQR(total dengue cases) – with missing values. This triggers Prophet to predict these values which resulted in a predicted trend without outbreaks. I used the following settings for both San Juan and Iquitos:

- Changepoint prior scale. If trend changes are being overfit (to much flexibility) or underfit (not enough flexibility) you can adjust the strength of the sparse prior using this setting. By default this parameter is set to 0.05. Increasing it will make the trend more flexible. I changed this setting to 0.001. With outliers removed it provides a stable seasonal pattern.
- Yearly seasonality. The default Fourier order for seasonality is 10. It determines how quickly the seasonality can change. I did some trial runs and concluded that 5 provides the best result.

- By default Prophet fits additive seasonality's (mode), meaning the effect of the seasonality is added to the trend to get the forecast. I assumed that in the dengue time series, the seasonality is a constant additive factor, rather than that it grows with the trend. I concluded that multiplicative only applies to the short outbreak periods.
- Oddly enough Prophet continues to predict below zero values using a logistic growth model with a floor of zero. The final model uses a cap equal to 1.5 * IQR(total dengue cases) and a floor of 20 for San Juan and 4 for Iquitos.

Feature selection

Extra regressors are put in the linear component of the model, so the underlying model is that the time series depends on the extra regressor as either an additive or multiplicative factor. The extra regressor must be known for both the history and for future test dates. It thus must either be something that has known values (such as temperature), or something that has separately been forecasted.

San Juan

The San Juan model uses the following variables as predictors:

- Temperature: 25 weeks moving average. Because it has the highest correlation with dengue cases.
- Humidity: 18 weeks moving average. It has an higher correlation with dengue cases but lower correlation with temperature then dew point.
- Precipitation: 52 weeks rolling sum. Because it showed no correlation with humidity and mosquitos just love human made containers with stagnant water. It is also the only variable with a negative correlation with dengue cases.

Iquitos

The Iquitos model uses the following variables as predictors:

- Air temperature: 8 weeks moving average.
- Humidity: 3 weeks moving average

- Precipitation: 6 weeks rolling sum;

San Juan: Classify years with outbreak using decision tree

The Prophet model described above will do a decent job for predicting seasonal patterns in dengue cases. However it does not predict outbreaks. I added a simple decision tree, based on a random forest algorithm, that classifies per year whether or not an outbreak will take place.

To be accurate we should be modeling per month or even week. However, the aim is just to see if such approach would be feasible. Hence, I created a simple set of training data and a model which uses following ingredients:

- The train data per year has variable called "outbreak" that classifies each year as one with or without an outbreak ("yes" or " no"). A year is considered to have an outbreak if the total dengue cases per semester reached 125 or more in the second semester. This variable is the dependent variable which the algorithm must predict.
- Independent variables per year: average temperature and sum of precipitation in the third quarter of each year (outbreaks mostly start in the fourth quarter).

The random forest classifier was trained for San Juan with default settings on data spanning from 1991 until 2005. It predicted correctly 2006 as a year without outbreak and 2007 as a year with outbreak. The "unseen" years in the test data for the competition where classified as follows:

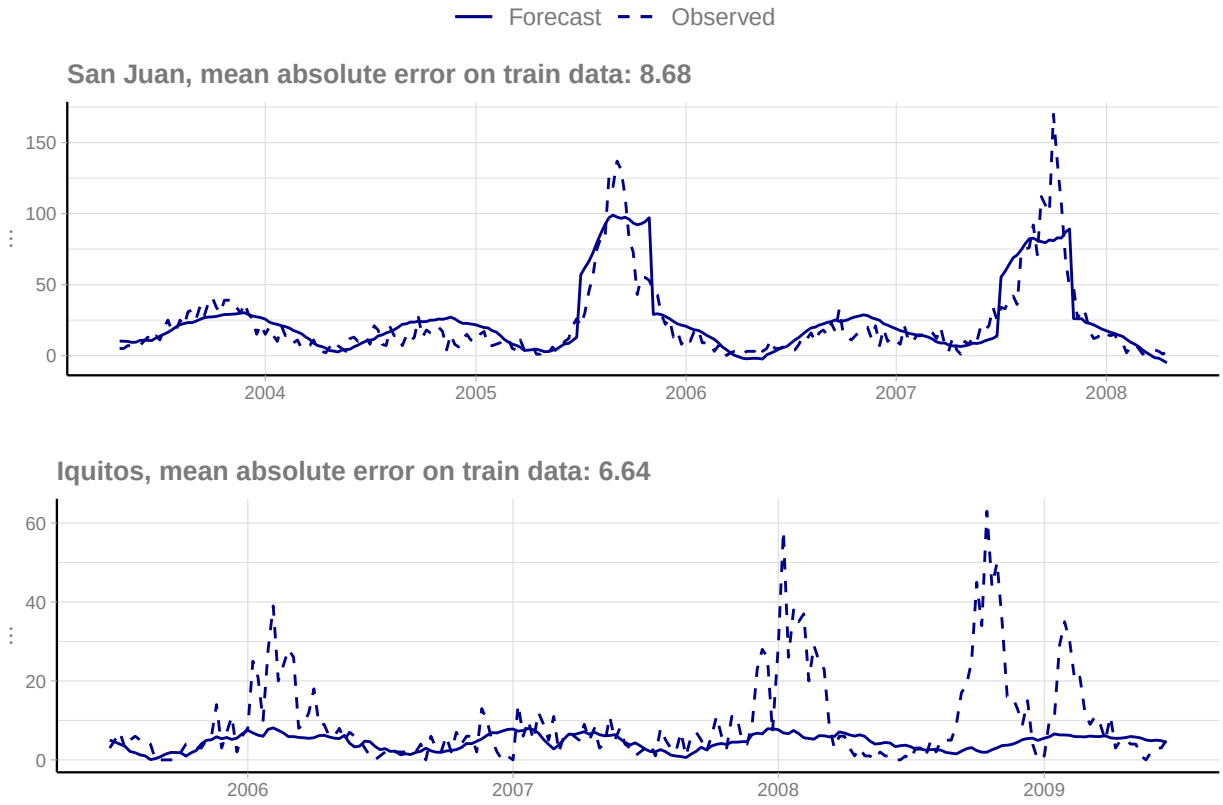
- 2008 No, which is plausible.
- 2009 Yes, which is incorrect based on the historical facts.
- 2010 Yes, which is plausible.
- 2011, No , which is plausible.
- 2012, Yes, which is plausible.

A shortcoming of this approach is that it predicts on a yearly basis. I therefore assumed that outbreaks always start in August and end in October. During this time frame the predicted dengue cases are multiplied by an arbitrary 3 if a year is classified as a year with outbreak.

Model evaluation

The model performs very well on the training data as shown below. It is however not yet competitive on the leaderboard. The submitted data results in a mean absolute error of 24.3846, which defeats the benchmark as published by DrivenData, but is much higher than the results during model validation. I assume that this is caused by not being accurate enough in predicting the outbreak periods.

The figures below show the results based on training data. Outbreak prediction was not added to the Iquitos model.



Discussion

- Prophets predicts negative values even when history does not contain these values. Issue persist on smaller scale in logistic models. Solved by replacing negative values with minimum seen values. I did not find the root cause.
- Prophets automated detection of trend change points can be a potential source of overfitting. Manually setting trend change points in Prophet allows for a human in the loop machine learning system. This is a potential solution for the bias the algorithms develops towards a specific trend direction due to large outliers in observed dengue totals.
- The total dengue cases per week are non-negative integers which arise from counting. Other algorithms, that focus on count data, might be a better fit.
- I did not analyze the probabilities that are the basis for the classification model. Accuracy might improve with a higher or lower probability needed to classify a yeas as a year with an outbreak.
- I was able to improve on forecasting the outbreaks by adding a multiplier to the time series of San Juan: '0' for timeframes without an outbreak, '0.5' during minor outbreaks (e.g. 1991) , '1.5' during medium outbreaks (e.g. 1998) and '2' during huge outbreaks (e.g. 1994). The timeframes for outbreaks in the unseen test data can be derived from the following blog: https://www.puertoricodaytrips.com/dengue-puerto-rico/. This approach resulted in rank 154 of 5456 on the leaderboard. I dropped the approach from this study because it is cheating. It does however confirm that predicting outbreak periods with a tree based algorithm could be a plausible and winning solution.

Conclusion

I learned how to add predictive value to time series by calculating moving averages on interval data (e.g. temperature) and rolling totals on precipitation. Correlation in time series is not static as shown with an example. Investigating rolling correlations when analyzing time series can prevent the analyst from presenting wrong assumptions.

The model defeats the benchmark by 5,55% but is not yet competitive. This is probably due to the fact that it lacks the ability to predict the periods in which the outbreaks will happen and the magnitude of these outbreaks. This study does show that it might be feasible to design a model that predict dengue cases by distributing objectives over different machine learning algorithms. Examples of objectives are to predict the seasonal trend and the magnitude of an outbreak.
The competition was to predict dengue cases over a future period of multiple years given meteorological values per week. In real live we do not have these future values other than short term weather forecasts. Short term predictions using autocorrelation based algorithms will yield much better accuracy.

References

Resources consulted during background study

I read, and used, content on the following websites during the background study about dengue:

- https://www.who.int/denguecontrol/mosquito/en/
- https://www.cdc.gov/dengue/epidemiology/index.html
- https://gisgeography.com/ndvi-normalized-difference-vegetation-index/

Resources consulted for technical design

I recommend reading the following online posts which I used as a basis for the technical solution:

- https://facebook.github.io/prophet/docs/multiplicative_seasonality.html
- https://www.datacamp.com/community/tutorials/detect-anomalies-anomalize-r

R packages used in software code

The following R open source packages are used in the software code which is described, or used, in this document:

- **'tidyverse'**, a collection of R packages for data science. Includes 'dplyr' for data wrangling tasks like data manipulation and combining multiple files into a train and a test (unseen) dataset. Ships with 'ggplot2' for data visualization.
- **'lubridate'** provides date functions, e.g. extract month and year from a date.
- **'imputeTS'** great at imputing missing values in structured time series with, among others, Kalman Smoothing.
- **'TTR'** for calculating moving averages.
- **'weathermetrics'** includes a function to convert from Kelvin to Celsius.
- **'tibbletime'** on its own has useful functions for manipulating time-based tibbles. Working with time-based tibbles is a prerequisite for the 'anomalize' package.
- **'anomalize'** enables a tidy workflow for detecting anomalies (outliers) in structured time series. I used it for anomaly detection and time series decomposition.

- **'prophet'** is a forecasting tool open sourced by Facebook. It is the core of the forecasting model described in this document.
- **'GGally'** is used to visualize pairwise comparison of multivariate data.
- **'ggpubr'** arranges multiple data visualizations into one figure.
- **'ggthemes'**, provides 'ggplot2' themes and scales that replicate the look of data visualizations.