<<Project Report>>

Title:

DengAI, Predicting Disease Spread

(DrivenData)

Team Members:

Somayeh Mohammadpour, SXM155431

Matthew Chen, MDC150630

Suvanam Saikumar, SXS155933

Fall 2017

INDEX

1. **Introduction.....3**
2. **Problem Definition and Algorithm.....3-5**

2.1 Task Definition.....3

2.2 Algorithm Definition.....4

2.2.1 Generalized Linear Regression.....5

2.2.2 Random Forest.....5

3. Preprocessing Technique.....5

4. Experimental Evaluation.....6

4.1. Methodology.....6

4.2 Results.....6-7

4.2.1 Generalized Linear Regression.....7-8

4.2.2 Random Forest.....9-10

4.3 Discussion.....10

5. Future Work.....10

6. Conclusion.....11

7. Contribution of team members.....11

8. References.....11

1. Introduction

Dengue fever is a mosquito-borne disease that occurs in tropical and sub-tropical parts of the world. In mild cases, symptoms are similar to the flu: fever, rash, and muscle and joint pain. In severe cases, dengue fever can cause severe bleeding, low blood pressure, and even death.

Since this illness is carried by mosquitoes, the transmission dynamics of dengue are related to climate variables such as temperature and precipitation. The relationship of the number of dengue fever cases and the climate is complex and an increasing number of scientists are hypothesizing that climate change is likely the reason for the increase in cases and could cause problems worldwide.

Using the Machine Learning packages available in spark we aim to find the relationship between the climate and dengue fever cases in hopes that it will aid in research and resource allocation to help fight this illness. This task is not simple and will require very strong machine learning algorithms to be able to predict the cases accurately.

2. Problem Definition and Algorithm

2.1 Task Definition

The main goal of this data is to be able to predict the number of dengue fever cases given a set of input features. The website that there is five years of San Juan dengue fever data and three years of Iquitos dengue fever data. The test data also contains no overlap of input rows since it is a future hold-out. The data also contains some NaNs that will need to be filled.

As input, we have the following set of information:

- City - City name sj and iq, abbreviations for San Juan and Iquitos, respectively
- Year - The year
- WeekOfYear - The week of the year out of 52
- week_start_date - The date of the first day of the week

Climate Information

- station_max_temp_c - The maximum temperature in celsius
- station_min_temp_c - The minimum temperature in celsius
- station_avg_temp_c - The average temperature in celsius
- station_precip_mm - The precipitation in millimeters
- station_diur_temp_rng_c - The Diurnal Temperature in celsius
- precip_amt_mm - The total precipitation in millimeters
- reanalysis_sat_precip_amt_mm – Total precipitation in millimeters
- reanalysis_dew_point_temp_k – Mean dew point temperature in kelvin
- reanalysis_air_temp_k – Mean air temperature in kelvin
- reanalysis_relative_humidity_percent – Mean relative humidity percentage
- reanalysis_specific_humidity_g_per_kg – Mean specific humidity in grams per kilogram

- reanalysis_precip_amt_kg_per_m2 – Total precipitation in kilograms per squared meter
- reanalysis_max_air_temp_k – Maximum air temperature in kelvin
- reanalysis_min_air_temp_k – Minimum air temperature in kelvin
- reanalysis_avg_temp_k – Average air temperature in kelvin
- reanalysis_tdtr_k – Diurnal temperature range in kelvin

Normalized Vegetation Index

- ndvi_se - Pixel southeast of city centroid
- ndvi_sw - Pixel southwest of city centroid
- ndvi_ne - Pixel northeast of city centroid
- ndvi_nw - Pixel northwest of city centroid

we have totally 21 features, a composite key, (city, year, weekofyear), and a label for each as a column known as total_cases. The total number of training cases, including both San Juan and Iquitos, is 1456 and the total number of test cases is 416.

2.2 Algorithm Definition

Since the prediction variable is the count of the total number of cases,it can only be a continuous variable and will not be bounded. So, we have chosen two regression models, Generalized Linear Regression algorithm and the Random Forest Ensemble method. We chose these two algorithms because we would like to compare two algorithms two each other. We would also like to compete against the other participants in this competition so we chose the Random Forest algorithm since it is very effective at predicting labels from datasets.

2.2.1 Generalized Linear Regression

The Generalized Linear Regression algorithm takes the features of the input data into account and looks for a variant of the input function that best fits the labels provided. The input functions that are allowed are dependent on the input family that the GLM is going to be using. The reason for choosing the generalized linear models is that is allows us to use response data which can take any value like (1's or 0's-logistic regression) or it can take number of counts (poisson distribution).

```
Glr = GeneralizedLinearRegression(family="poisson" , link="sqrt" , maxIter=10,     regParam=0.3)

Model = glr.fit(trainingData)

prediction = model.transform(testingData)
```

2.2.2 Random Forest

The Random Forest Algorithm takes the features of the input data and tries to split on features until the max allowed depth has been reached or it see that it has been split enough. The input for the Random Forest algorithm are the number of trees that need to be trained, the max depth, which is the depth limit we want each tree to go before stopping, and the max bucket

size, the size of the bucket we want to the algorithm to randomly select from at each subtree root. Since this dataset uses a regression label the impurity can only be variance.

```
rfr = RandomForestRegressor(impurity="variance" , numTrees = 50, maxBin = 15, maxDepth= 14)

Model = rfr.fit(trainingData)

prediction = model.transform(testingData)
```

3. Preprocessing Technique

When we are given the data the labels and the features are separate. So we decided to combine them into one dataframe. So, the resulting dataframe for the training data was the index, the features, and the labels. The next step in preprocessing is to create a features vector for the dimensionality reduction and the training of the algorithms later. For the dimensionality reduction 14 features seemed to have the best predicting power. Once we get the reduced features vector we then need to split on the city because the city are independent of each other and will create a "noisy" prediction if they are not separated. The last step in our preprocessing was to fill in any null values with a zero because we need all the rows when we test against the Data Driven MAE score.

4. Experimental Evaluation

4.1 Methodology

The one criteria we are using to evaluate the performance of the algorithms is by observing how close the predictions are to the labels. The metric we are using to determine this is the Mean Absolute Error. This metric increases when predictions are not performing well and will decrease the closer you are to the label. The experiment is testing if we can predict the number of Dengue Fever cases that occur given certain environmental parameters. The independent variables are city, year, and a week of year, where city is more independent than the year and weekofyear. The dependent variables are the week_start_date and the other environmental variables that are listed above. We see the week_start_date as a feature because as time passes a spot that does not have dengue fever to begin with, may start to have cases after a certain date.

When we received this training data, the train set was split into a features file and a labels file that had to be combined in the preprocessing of the data. Once we combine the dataset we could see that the cases were related to time in some way and makes the data seem realistic because this illness is seasonal like the Flu. When you briefly look at the training data that is given to you, you can see that the number of cases of dengue fever increase during the end and beginning of the year and drop in the middle of the year. Once we had the training data combined into one dataframe we used Dimensional Reduction and Cross Validation to get the

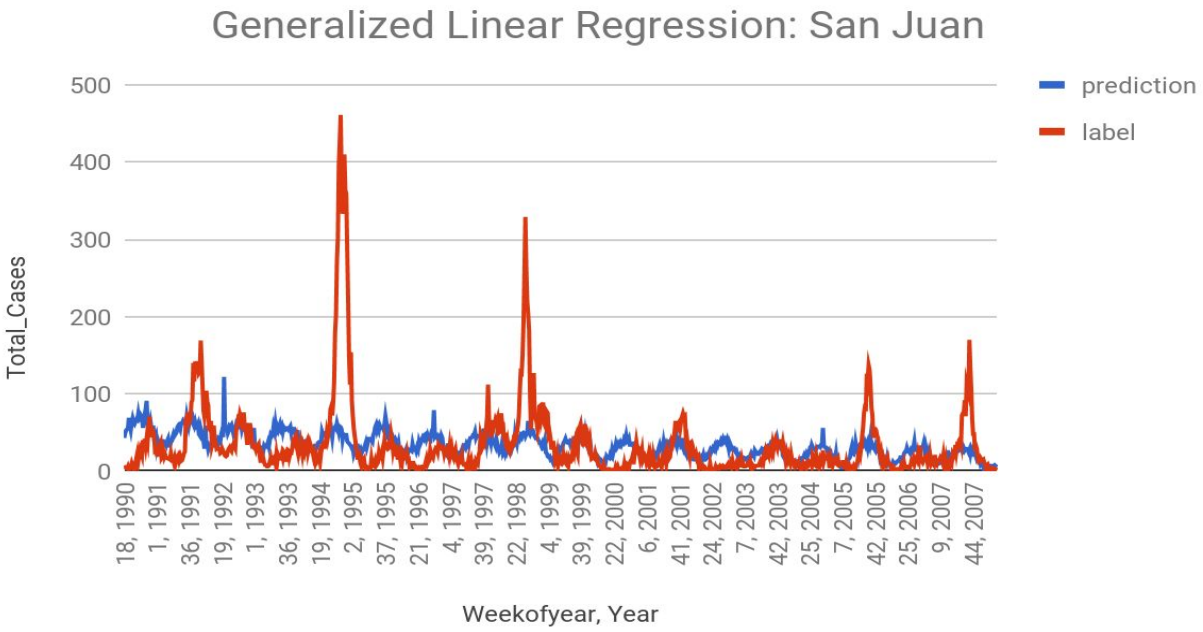
best results. To determine the best set of parameters we had to pick a range to test and see if that would return better results than a previous range. We then trained and tested the model and used it to predict the total_cases that would occur given the city, date, and week. The way the website is determining the accuracy of this data is by using the Mean Absolute Error metric. Since the website represents the accuracy of the data using the Mean Absolute Error metric, this is how we will be determining the performance of the Generalized Linear Regression model and the Random Forest Ensemble model.

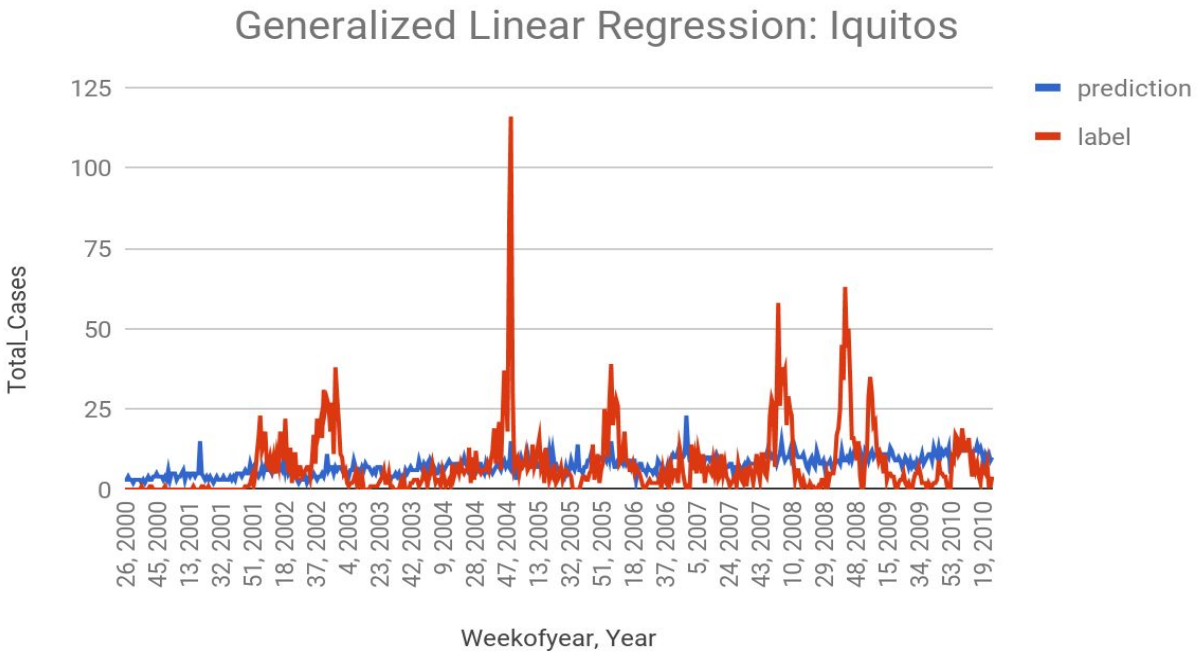
4.2 Results

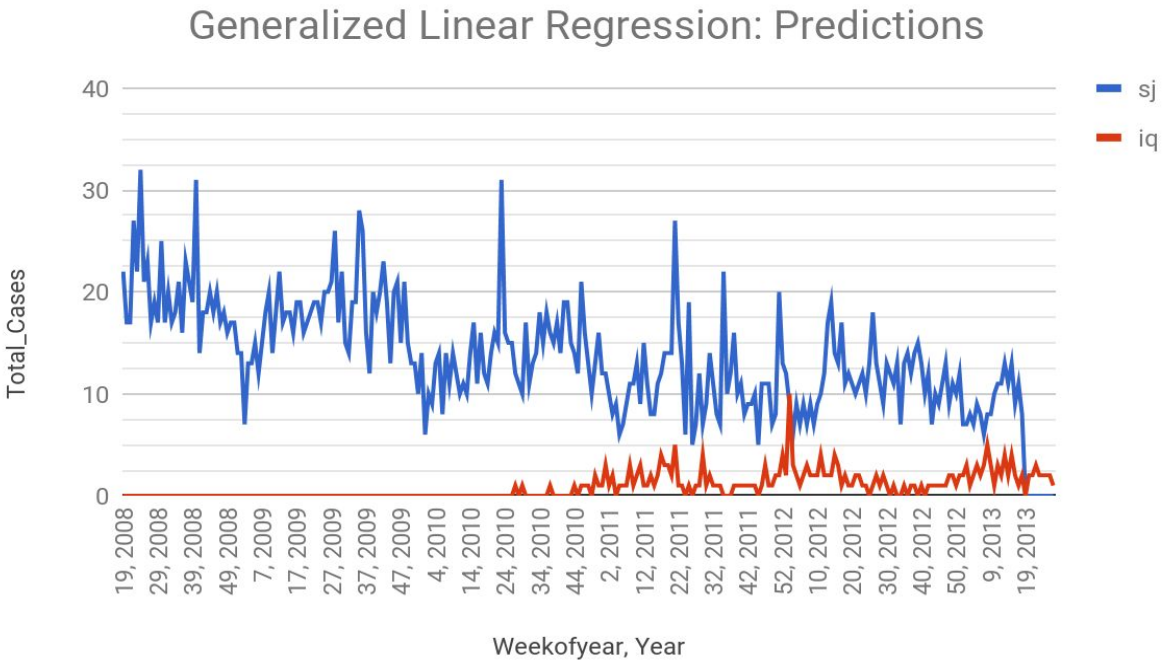
From the data in the training dataset we can make the assumption that Dengue Fever cases will greatly increase during the months of October to February. We can see that the Random forest algorithm can predict the data better since it also has Dengue Fever predictions that spike around the same time. In the Generalized Linear Regression algorithm we can see that the predictions are more constant than we would like which is understandable. The predictions of the Random Forest algorithm are similar to the Generalized Linear Regression algorithm but the predicting power is more accurate than the GLM which is what we expect to occur. This is statically significant because we can see that from the Random Forest predictions that we should expect the number of cases to increase near the end of the year. Using the Generalized Linear Regression algorithm and the Random Forest algorithm to predict the total number of cases every week, showed that the Generalized Linear Regression algorithm does not perform as well as the Random Forest Ensemble Method. When we were training and testing the data, we wanted to see how well we performed and to make sure we did not overfit the algorithms to the data. So, we calculated the MAE of each of the algorithm's predictions on the data we had. To further decrease the MAE we needed to throw away some features that were not as helpful as others and we needed to find the right parameters to use.. So, to do this we used Dimensional Reduction and Cross Validation. When we were testing the number of features we would use after Dimensional Reduction we determined that the optimal number of features that are needed were 14 features. For the Cross Validation portion of GLM in this experiment we determined that the optimal range for max iterations, maxIter, would be 8-12 and the regularization parameter, regParam, would be 0.4-0.8. For the Cross Validation portion of RFR in this experiment the optimal range for maximum depth, maxDepth, was 20-25 and the optimal range for the maximum number of bins, maxBins, was 32-48.

	MAE (Mean Absolute Error)	
City	GLM	RFR
San Juan	25.7094	8.3258
Iquitos	6.2615	2.2019

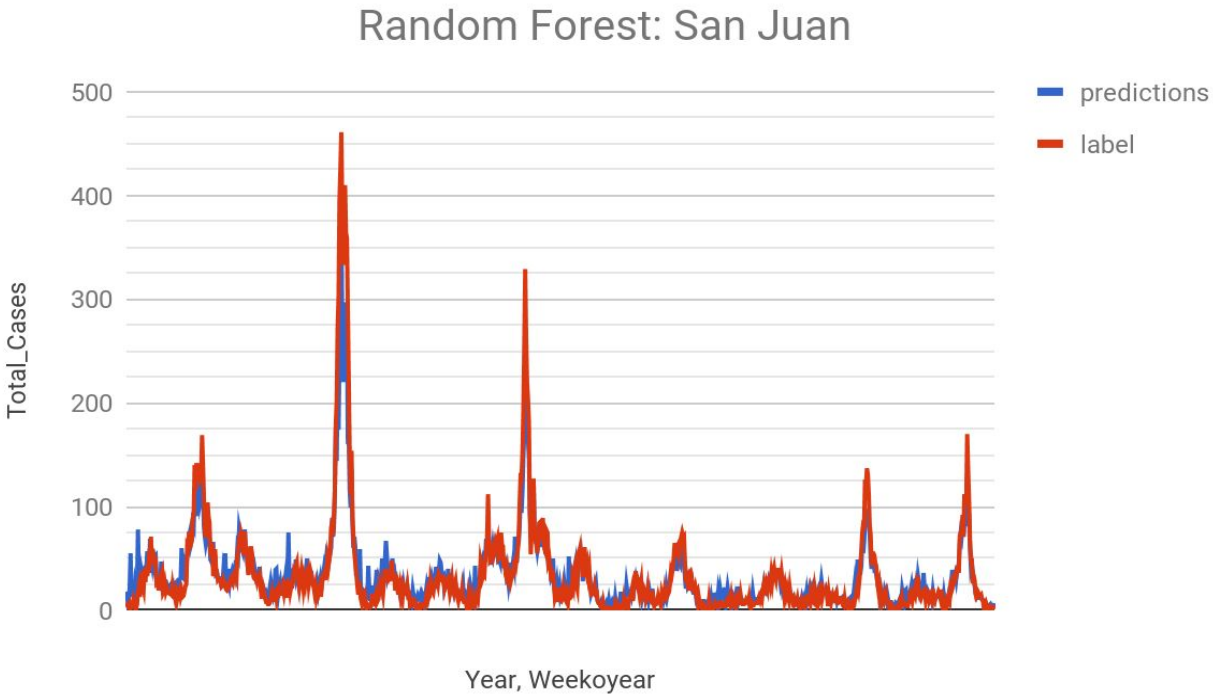
4.2.1 Generalized Linear Regression

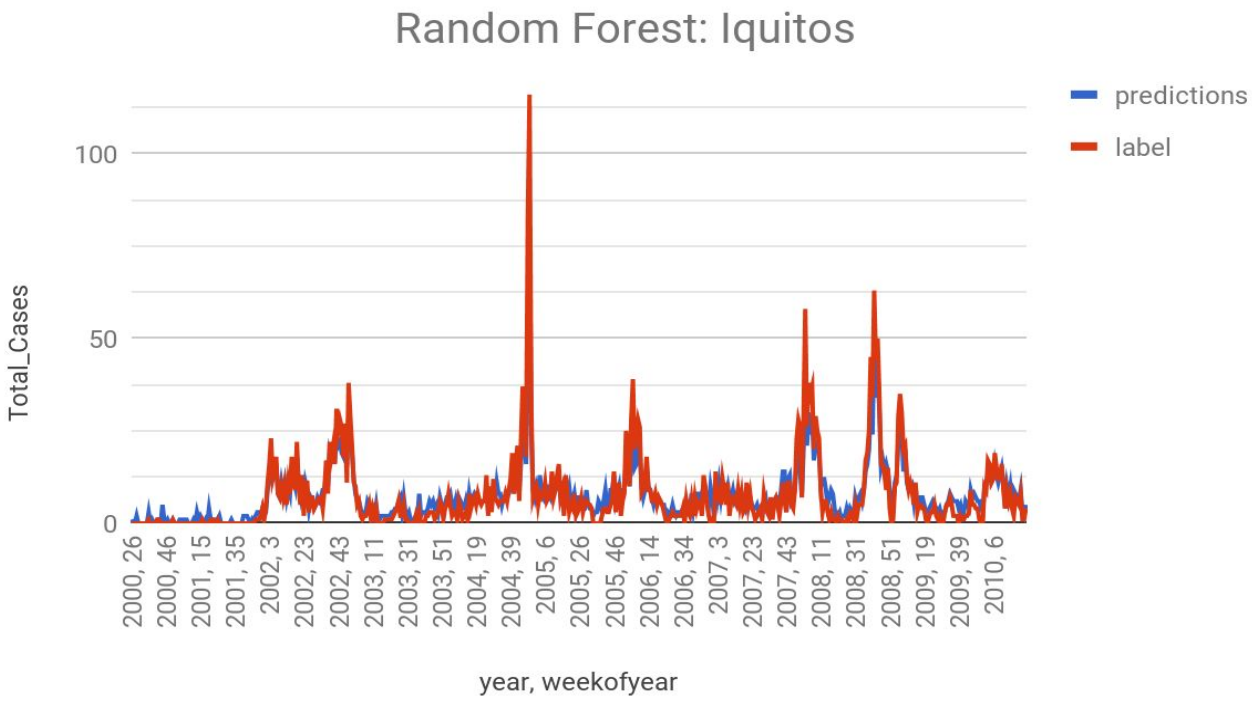


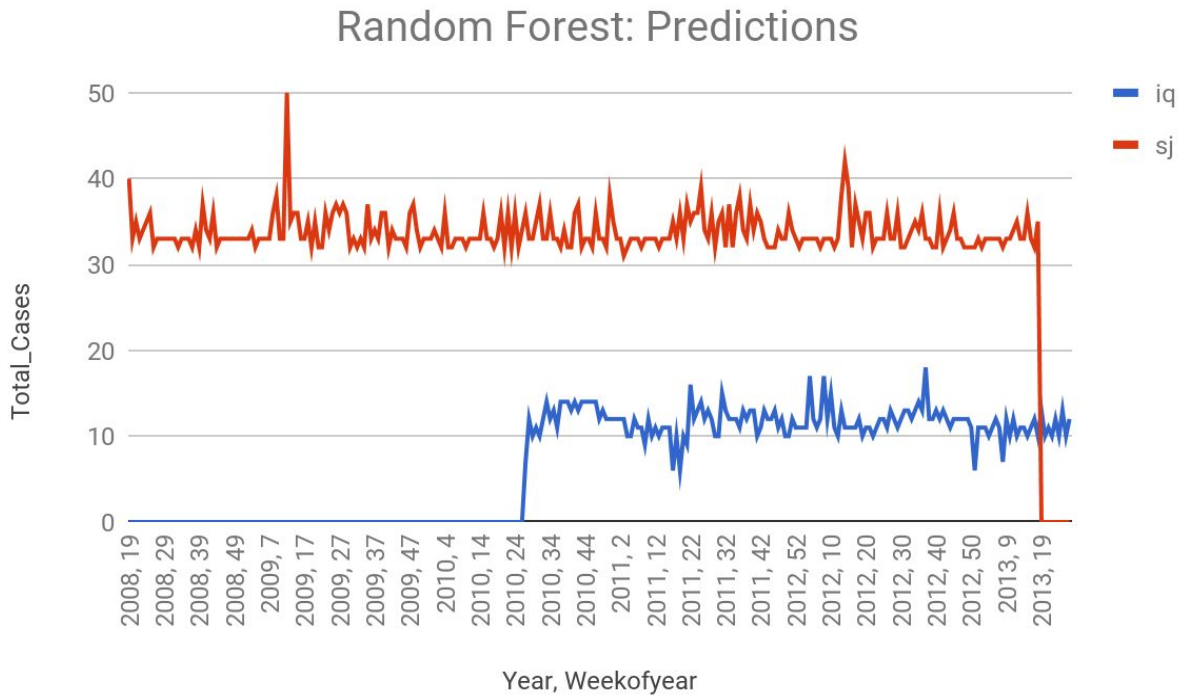

Generalized Linear Regression: Iquitos
prediction
label
Total_Cases
125
100
75
50
25
0
26, 2000
45, 2000
13, 2001
32, 2001
51, 2001
18, 2002
37, 2002
4, 2003
23, 2003
42, 2003
9, 2004
28, 2004
47, 2004
13, 2005
32, 2005
51, 2005
18, 2006
37, 2006
5, 2007
24, 2007
43, 2007
10, 2008
29, 2008
48, 2008
15, 2009
34, 2009
53, 2010
19, 2010
Weekofyear, Year


Generalized Linear Regression: Predictions
sj
iq
Total_Cases
40
30
20
10
0
19, 2008
29, 2008
39, 2008
49, 2008
7, 2009
17, 2009
27, 2009
37, 2009
47, 2009
4, 2010
14, 2010
24, 2010
34, 2010
44, 2010
2, 2011
12, 2011
22, 2011
32, 2011
42, 2011
52, 2012
10, 2012
20, 2012
30, 2012
40, 2012
50, 2012
9, 2013
19, 2013
Weekofyear, Year

4.2.2 Random Forest Regression







4.3 Discussion

Yes, our hypothesis was supported more heavily from the Random Forest algorithm rather than the Generalized Linear Regression. The Generalized Linear regression algorithm seems to be more suited toward datasets that have a more consistent pattern and does not fit datasets that have a more randomized pattern, such as this one. The Random Forest Algorithm is better suited for this because if you look at the data we can see that it was better in fitting the dataset than the Generalized Linear Regression algorithm. Since the RFR algorithm is not dependent on a function like the GLM algorithm is RFR will naturally perform better. So, since the training dataset has no repeated function, there is no way the GLM will be able to accurately predict any future cases.

5. Future Work

We can do more analysis by considering other extra features like life expectancy and living conditions of the people. If we can include these kind of features we can build a more efficient model which predicts the total number of cases at any location. We can also build an more efficient model by using deep learning techniques.

Another problem with our project is that we are using only two algorithms to try and predict the number of cases that can occur at a certain time. In a normal situation we would be able to change algorithms if we need to. Another problem is that predictions of the regression algorithms. These regression algorithms output decimal approximations rather than integers.

This can cause problems with evaluation since the labels for the dataset requires integers and not decimals.

6. Conclusion

The results from the Machine Learning algorithms we chose show that the data can be predicted to a certain degree. The results also show that ensemble methods will have a higher success rate in accurately predicting the cases that will occur than the non-ensemble methods. We can see that from the predictions and the training data itself that the number of cases increase during the end of the year so that will help prepare communities for the illness. Given this information, ensemble methods should be used to predict the cases since they are more well adapted for this kind of dataset.

7. Contribution of team members

Somayeh Mohammadpour - Wrote and put together the report

Matthew Chen - Trained and tested the Random Forest Algorithm

Suvanam Saikumar - Trained and tested the Generalized Linear Regression Algorithm

8. References

1. https://spark.apache.org/docs/2.1.0/ml-features.html#pca
2. https://spark.apache.org/docs/2.1.0/ml-classification-regression.html
3. http://spark.apache.org/docs/2.1.0/
4. https://www.drivendata.org/competitions/44/dengai-predicting-disease-spread/