

For this problem we are investigating a time-series problem. Specifically, trying to determine the number of expected weekly cases of dengue in two cities (San Juan and Iquitos, Peru). It is expected that this project might be able to help understand the relationship between climate and dengue dynamics, and improve research initiatives and resource allocation to fight pandemics.

We are supplied with a training set and a test set from the website [drivendata.org](https://drivendata.org).

(<https://www.drivendata.org/competitions/44/dengai-predicting-disease-spread>) Both sets contain 23 independent variables. There is also a training set for the dependent variable, number of dengue cases in a particular week. For San Juan, the training data is from 4-30-1990 to 4-29-2008. (936 weeks) For Iquitos the range is 07-01-2000 to 06-30-2010 (520 weeks) The independent variables (besides the week of year of a particular row) are:

Related to vegetation (Normalized Difference 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

Satellite precipitation measurements:

- `precipitation_amt_mm` – Total precipitation

Related to climate forecast system reanalysis:

- `reanalysis_sat_precip_amt_mm` – Total precipitation
- `reanalysis_dew_point_temp_k` – Mean dew point temperature
- `reanalysis_air_temp_k` – Mean air temperature
- `reanalysis_relative_humidity_percent` – Mean relative humidity
- `reanalysis_specific_humidity_g_per_kg` – Mean specific humidity
- `reanalysis_precip_amt_kg_per_m2` – Total precipitation
- `reanalysis_max_air_temp_k` – Maximum air temperature
- `reanalysis_min_air_temp_k` – Minimum air temperature
- `reanalysis_avg_temp_k` – Average air temperature
- `reanalysis_tdtr_k` – Diurnal temperature range

Related to daily climate data weather station measurements:

- `station_max_temp_c` – Maximum temperature

- `station_min_temp_c` – Minimum temperature
- `station_avg_temp_c` – Average temperature
- `station_precip_mm` – Total precipitation
- `station_diur_temp_rng_c` – Diurnal temperature range

Related to time:

- `weekofyear` – The particular week of a given year
- `year` – The given year

The dependent variable for the training sets is:

- `total_cases`: this is the number of cases of dengue for that week.

(note, the sets also contain the city identifier, and information that is related to the week of observation. This means that these could also be classified as features, but as we break out the data by city, it is no longer a feature for that set)

This is a time series problem, which is a well-defined type of problem for data scientists to approach. This data also contains cross-sectional data as captured at the same time period of one week.

## Literature

To back our assumption that this is an appropriate problem, and ways to approach, we found the following articles.

Hyndman, Rob et al. (2014) *Forecasting: Principles and Practice, Chapters 5 and 8*. Otexts.com.

Nedjadi, Taoufik, "Tackling dengue fever: Current status and challenges", December 2015, *Virology Journal*, <https://doi.org/10.1186/s12985-015-0444-8>

Van Buuren, Stef et al. "mice: Multivariate Imputation by Chained Equations in R", December 2011, *Journal of Statistical Software*, Volume 45, Issue 3.

Wongkoon, S et al. "Development of Temporal Modeling for Prediction of Dengue Infection in Northeastern Thailand." 2012, *Asian Pacific Journal of Tropical Medicine*, 5(3), 249-252. doi:10.1016/s1995-7645(12)60034-0

.Ramadona, A. et al. "Prediction of Dengue Outbreaks Based on Disease Surveillance and Meteorological Data." 2016, *Plos One*, 11(3). doi:10.1371/journal.pone.0152688

Hyndman's work of course takes us through several ways to approach this problem set, from using multiple linear regression, and ARIMA modeling.

Multiple linear regression involves one dependent variable against more than one predictor variable. Instead of a model being built on just one predictor variable against the dependent variable, multiple linear regression allows for the observed dependent values to be calculated against all the variables at once. While time (especially seasonality) can be dummy variable into the model, it is not easy to model time into the model.

ARIMA modeling is very much built to take into account the requirements of time series modeling, especially with the aim to describe the autocorrelations within the data. Autoregression (where each value carries information about the previous value, lagged to some degree), differencing (between each value), and moving average (again, potentially lagged to different degrees) are ideas that are captured much more readily with ARIMA than with linear regression.

Nedjadi's work explains how dengue is affecting many countries, and to what degree humans are susceptible, along with how the disease does not currently have an effective vaccine. Concluding that the economic burden of dengue is extremely high, and that further work (and money) is required to come up with an effective control. Being able to show the correlations between temperature/rain/vegetation may aide in getting some of that help redirected towards that effort (thus this analysis may help some arguments for money someplace)

Van Buuren's work allowed us to understand how we can approach the missing values from our datasets, to 'backfill' the NA values that exist in the datasets. This is a specific R implementation of Rubin's work (1987, 1996) on incomplete data problems using multiple imputation involving Markov chain Monte Carlo.

Wongkoon and Ramadona discuss that dengue virus is sensitive to changes in the meteorological pattern. Sustainability of dengue virus is highly influenced by precipitation amount, which includes rainfall, and snowfall. The life cycle of a mosquito carrying the dengue virus is associated with precipitation, temperature, and humidity in the atmosphere. Also, vegetation lands, which work as a habitat for dengue mosquitoes, with suitable meteorological conditions impact the number of "total\_cases" of dengue outbreak. After understanding both the studies and how the different factors affect the 'total\_cases' of dengue, we decided to keep below given independent variables in the model. Also, the independent variable validation is done, considering the multicollinearity problem and correlation of independent variable.

## Data Mining/Cleaning

Firstly, we load the following packages into R environment.

```
library(plyr)
library(dplyr)
library(ggplot2)
library(MASS)
library(knitr)
library(rsample)
library(corrplot)
library(mice)
library(caret)
```

This data required some amount of cleaning. Of our 20 predictors that weren't related to time ( $1456 * 20$ ) there were 29120 possible observations. In that set, there were 581 missing values or close to two percent. For a simple time series analysis (not taking into account the predictors, which is what we were able to get through in class) the data does

support 1456 values of the total cases ((936 and 520) dependent variable in weekly captured data.

Summary statistics as follows:

ndvi_ne	ndvi_nw	ndvi_se	ndvi_sw	precipitation_amt_mm
Min. :-0.40625	Min. :-0.45610	Min. :-0.01553	Min. :-0.06346	Min. : 0.00
1st Qu.: 0.04495	1st Qu.: 0.04922	1st Qu.: 0.15509	1st Qu.: 0.14421	1st Qu.: 9.80
Median : 0.12882	Median : 0.12143	Median : 0.19605	Median : 0.18945	Median : 38.34
Mean : 0.14229	Mean : 0.13055	Mean : 0.20378	Mean : 0.20231	Mean : 45.76
3rd Qu.: 0.24848	3rd Qu.: 0.21660	3rd Qu.: 0.24885	3rd Qu.: 0.24698	3rd Qu.: 70.23
Max. : 0.50836	Max. : 0.45443	Max. : 0.53831	Max. : 0.54602	Max. : 390.60
NA's :194	NA's :52	NA's :22	NA's :22	NA's :13
reanalysis_air_temp_k	reanalysis_avg_temp_k	reanalysis_dew_point_temp_k		reanalysis_max_air_temp_k
Min. :294.6	Min. :294.9	Min. :289.6		Min. :297.8
1st Qu.:297.7	1st Qu.:298.3	1st Qu.:294.1		1st Qu.:301.0
Median :298.6	Median :299.3	Median :295.6		Median :302.4
Mean :298.7	Mean :299.2	Mean :295.2		Mean :303.4
3rd Qu.:299.8	3rd Qu.:300.2	3rd Qu.:296.5		3rd Qu.:305.5
Max. :302.2	Max. :302.9	Max. :298.4		Max. :314.0
NA's :10	NA's :10	NA's :10		NA's :10
reanalysis_min_air_temp_k	reanalysis_precip_amt_kg_per_m2	reanalysis_relative_humidity_percent		
Min. :286.9	Min. : 0.00	Min. :57.79		
1st Qu.:293.9	1st Qu.:13.05	1st Qu.:77.18		
Median :296.2	Median :27.25	Median :80.30		
Mean :295.7	Mean :40.15	Mean :82.16		
3rd Qu.:297.9	3rd Qu.:52.20	3rd Qu.:86.36		
Max. :299.9	Max. :570.50	Max. :98.61		
NA's :10	NA's :10	NA's :10		
reanalysis_sat_precip_amt_mm	reanalysis_specific_humidity_g_per_kg	reanalysis_tdtr_k	station_avg_temp_c	
Min. : 0.00	Min. :11.72	Min. :1.357	Min. :21.40	
1st Qu.: 9.80	1st Qu.:15.56	1st Qu.:2.329	1st Qu.:26.30	
Median :38.34	Median :17.09	Median :2.857	Median :27.41	
Mean :45.76	Mean :16.75	Mean :4.904	Mean :27.19	
3rd Qu.:70.23	3rd Qu.:17.98	3rd Qu.:7.625	3rd Qu.:28.16	
Max. :390.60	Max. :20.46	Max. :16.029	Max. :30.80	
NA's :13	NA's :10	NA's :10	NA's :43	
station_diur_temp_rng_c	station_max_temp_c	station_min_temp_c	station_precip_mm	
Min. :4.529	Min. :26.70	Min. :14.7	Min. : 0.00	
1st Qu.:6.514	1st Qu.:31.10	1st Qu.:21.1	1st Qu.: 8.70	
Median :7.300	Median :32.80	Median :22.2	Median :23.85	
Mean :8.059	Mean :32.45	Mean :22.1	Mean :39.33	
3rd Qu.:9.567	3rd Qu.:33.90	3rd Qu.:23.3	3rd Qu.:53.90	
Max. :15.800	Max. :42.20	Max. :25.6	Max. :543.30	
NA's :43	NA's :20	NA's :14	NA's :22	

The two cities "San Juan" and "Iquitos Peru" are located differently. The geographic conditions are different in two cities and their effects on the outbreak of Dengue must be unique. So, we decided to proceed with two different models for two different cities, so that the model's will provide accurate forecasting by capturing the right information from the predictor variables.

Here, we divided the data for two different cities and load the datasets into R

environment.

```
sj_naf_data <- read.csv("C:/Education/PredictiveAnalysis/dengTrain.SJ.5.csv")  
iq_naf_data <- read.csv("C:/Education/PredictiveAnalysis/dengTrain.IQ5.csv")  
train_results <- read.csv("C:/Education/PredictiveAnalysis/DengAI/dengue_labels_train.csv")
```

The above datasets do not include the "total\_cases", which is our dependent variable. For modelling we need the dependent variable in the same dataset which includes the independent variables. Hence, the dependent variable "total\_cases" is included in the datasets as follows:

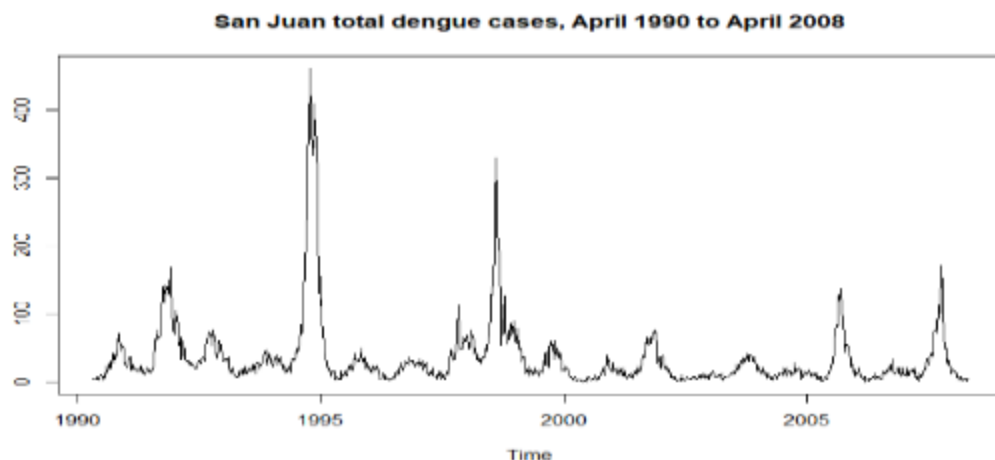
```
train_results_sj <- train_results %>% filter(city == "sj")  
train_results_iq <- train_results %>% filter(city == "iq")  
sj_naf_data <- sj_naf_data %>% mutate(total_cases = train_results_sj[,4])  
iq_naf_data <- iq_naf_data %>% mutate(total_cases = train_results_iq[,4])
```

Utilizing the mice package, we filled in all the NAs using code similar to:

```
mice(sj_naf_data , maxit =50)
```

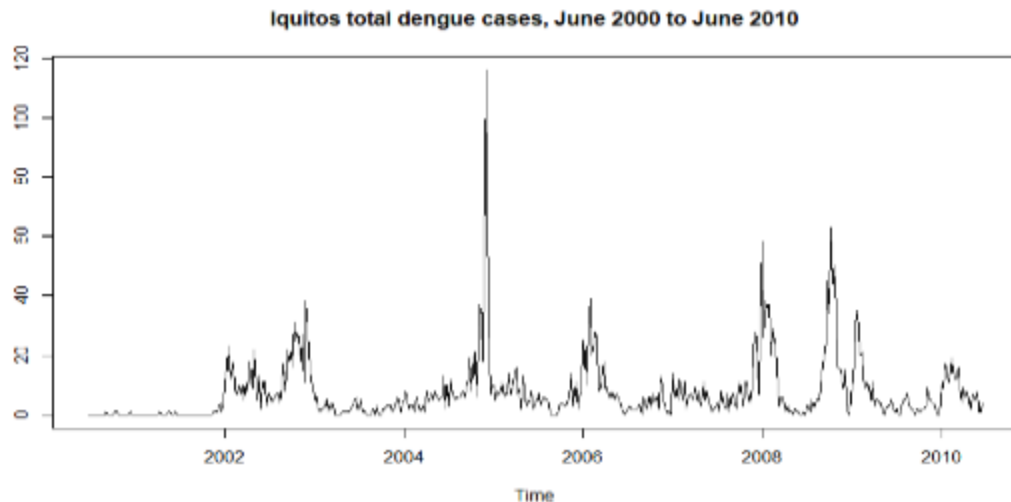
which allowed us to work with a reasonable complete set of predictors.

A visual description of the data:



(Figure 1)

Figure 1 shows the number of dengue cases for the city of San Juan (SJ) for the training-set period. While there might be a downward trend line in the spikes, we did not see any seasonality, cyclicity nor trend.



(Figure 2)

Figure 2 also did not reveal any obvious seasonality, trend or cyclicity for Iquitos (IQ).

## Variable selection for Modelling

As we are aware of the city name for which we are forecasting the number of total cases of dengue, we removed the variable "city" from the datasets. Also, we removed the "week\_start\_date" variable, as we have "weekofyear" which is more relevant to modelling.

## Variable selection for city "San Juan"

Initially, we consider all the independent variables for modelling as follows, using linear regression. Also, we used the forward elimination approach to obtain the best fit model, using most relevant independent variables.

```
fit_sj <- lm(total_cases~., sj_naf_data)
autofit_sj <- stepAIC(fit_sj)

summary(autofit_sj)

##
## Call:
## lm(formula = total_cases ~ year + weekofyear + ndvi_ne + ndvi_nw +
##     ndvi_se + reanalysis_avg_temp_k + reanalysis_dew_point_temp_k +
##     reanalysis_max_air_temp_k + reanalysis_min_air_temp_k + reanalysis_specific_hu
##     midity_g_per_kg,
##     data = sj_naf_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -81.96 -23.83  -8.72   9.28 352.05
##
## Coefficients:
##              Estimate Std. Error t value
## (Intercept)    3811.5814  1621.6529   2.350
## year           -1.3384    0.3688  -3.629
## weekofyear       0.7137    0.1278   5.584
## ndvi_ne        -88.0194   19.4359  -4.529
## ndvi_nw        129.7642   20.8800   6.215
## ndvi_se         87.9211   26.4395   3.325
## reanalysis_avg_temp_k    -13.9060    6.1754  -2.252
## reanalysis_dew_point_temp_k    -7.4894    4.9728  -1.506
## reanalysis_max_air_temp_k    10.5892    3.8196   2.772
## reanalysis_min_air_temp_k     6.4020    3.9753   1.610
## reanalysis_specific_humidity_g_per_kg     8.2522    5.0120   1.646
##
##              Pr(>|t|)
## (Intercept)    0.018961 *
## year           0.000300 ***
## weekofyear     3.08e-08 ***
## ndvi_ne        6.71e-06 ***
## ndvi_nw        7.77e-10 ***
## ndvi_se        0.000918 ***
## reanalysis_avg_temp_k    0.024566 *
## reanalysis_dew_point_temp_k    0.132387
## reanalysis_max_air_temp_k    0.005678 **
## reanalysis_min_air_temp_k    0.107640
## reanalysis_specific_humidity_g_per_kg    0.100003
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 46.61 on 925 degrees of freedom
```



```
## Multiple R-squared:  0.1859, Adjusted R-squared:  0.1771  
## F-statistic: 21.12 on 10 and 925 DF,  p-value: < 2.2e-16
```

We observe that the following independent variables are the most relevant for predicting "total\_cases" of dengue.

year

weekofyear

ndvi\_ne

ndvi\_nw

ndvi\_se

reanalysis\_avg\_temp\_k

reanalysis\_dew\_point\_temp\_k

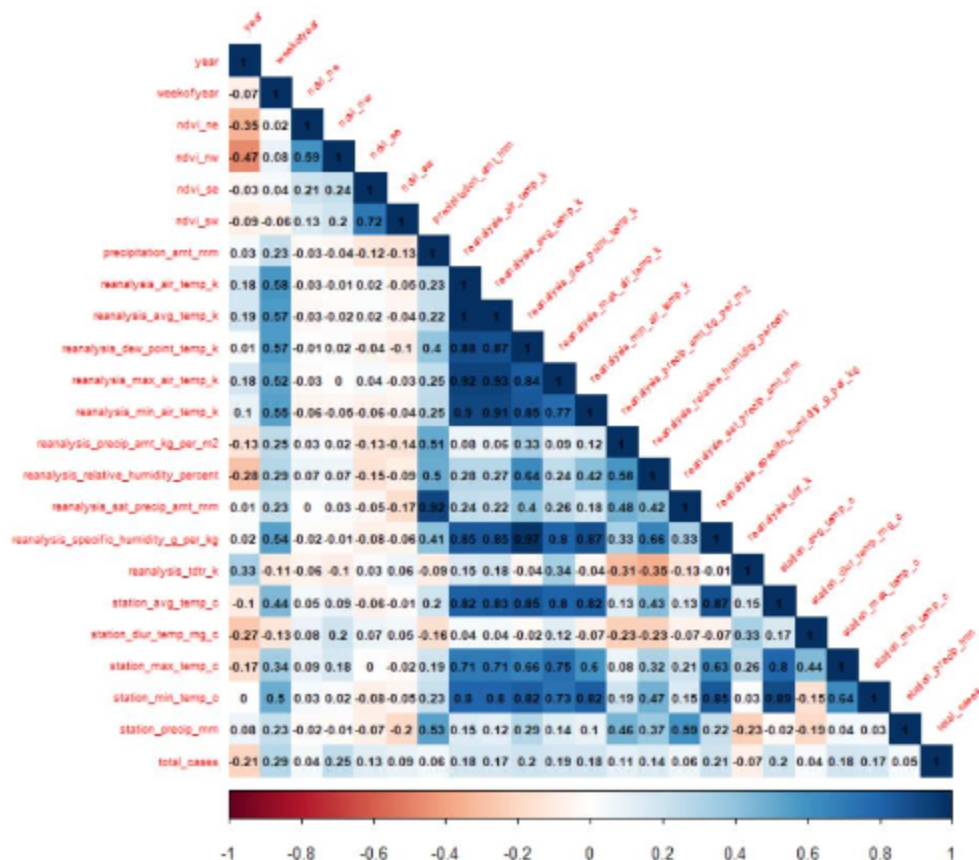
reanalysis\_max\_air\_temp\_k

reanalysis\_min\_air\_temp\_k

reanalysis\_specific\_humidity\_g\_per\_kg

To improve the model forecasting accuracy, we attempted to understand the relationship between the independent variables and their relationship with the dependent variable and plotted the correlation matrix as follows:

```
corrplot(cor(sj_naf_data), method="color", type="lower", number.cex = 0.6, tl.cex = 0.55, tl.srt = 45, addCoef.col = "black")
```



(Figure 3) Correlation matrix plot for all the independent variables of San Juan city

From the above correlation plot in figure 3, we observe many independent variables are highly correlated with each other. So, this raises a problem of multicollinearity.

Multicollinearity is a phenomenon in which one independent variable in a multiple regression model can be linearly predicted from the other independent variables with a substantial degree of accuracy. Because of multicollinearity it is very difficult to assess the effect of independent variables on the dependent variable.

After a thorough analysis, we found that though stepAIC used the relevant independent variables, some among them are still highly correlated. So, to eliminate the highly correlated

independent variables, we decided to keep the following variables in our model for city "San Juan" (SJ):

weekofyear  
ndvi\_nw  
ndvi\_se  
reanalysis\_relative\_humidity\_percent  
reanalysis\_specific\_humidity\_g\_per\_kg  
station\_min\_temp  
station\_max\_temp\_c

To validate the effect of selected independent variables we followed the works of Wonkoon and Ramadona as previously noted.

Variable selection for city "Iquitos Peru"

```
fit_iq <- lm(total_cases~., iq_naf_data)
autofit_iq <- stepAIC(fit_iq)

summary(autofit_iq)

##
## Call:
## lm(formula = total_cases ~ year + ndvi_ne + ndvi_se + reanalysis_max_air_t
emp_k +
##   reanalysis_relative_humidity_percent + reanalysis_specific_humidity_g_
per_kg +
##   station_avg_temp_c + station_diur_temp_rng_c + station_max_temp_c,
##   data = iq_naf_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -17.935  -5.155  -2.816   1.678  100.601
##
## Coefficients:
##              Estimate Std. Error t value
## (Intercept)  -630.7637   352.5764  -1.789
## year           0.4059     0.1636    2.481
## ndvi_ne       12.5802     7.9975    1.573
```

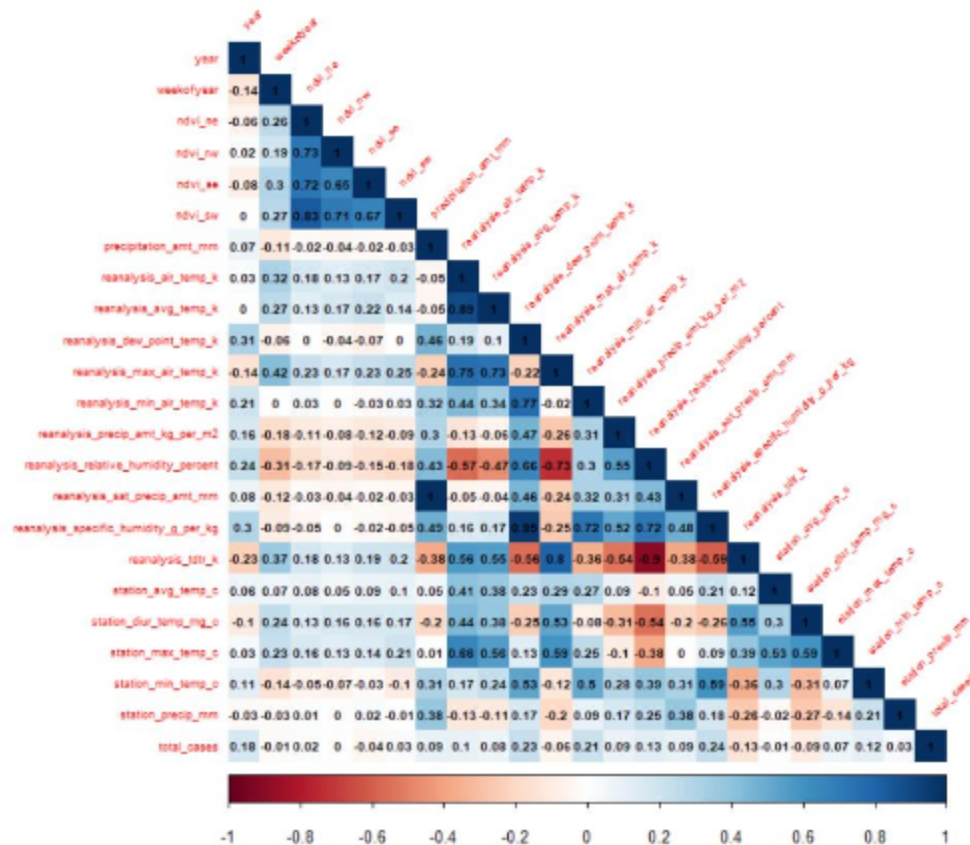
```
## ndvi_se -12.3391 8.4882 -1.454
## reanalysis_max_air_temp_k -0.6174 0.3536 -1.746
## reanalysis_relative_humidity_percent -0.3576 0.1518 -2.356
## reanalysis_specific_humidity_g_per_kg 2.5076 0.5987 4.189
## station_avg_temp_c -0.9291 0.4158 -2.235
## station_diur_temp_rng_c -0.6387 0.3863 -1.654
## station_max_temp_c 1.0253 0.5454 1.880
## Pr(>|t|)
## (Intercept) 0.0742 .
## year 0.0134 *
## ndvi_ne 0.1163
## ndvi_se 0.1467
## reanalysis_max_air_temp_k 0.0814 .
## reanalysis_relative_humidity_percent 0.0188 *
## reanalysis_specific_humidity_g_per_kg 3.3e-05 ***
## station_avg_temp_c 0.0259 *
## station_diur_temp_rng_c 0.0988 .
## station_max_temp_c 0.0607 .
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.3 on 510 degrees of freedom
## Multiple R-squared: 0.09972, Adjusted R-squared: 0.08383
## F-statistic: 6.277 on 9 and 510 DF, p-value: 1.938e-08
```

We observe that the following independent variables are the most relevant for predicting "total\_cases" of dengue.

```
year
ndvi_ne
ndvi_se
reanalysis_max_air_temp_k
reanalysis_relative_humidity_percent
reanalysis_specific_humidity_g_per_kg
station_avg_temp_c
station_diur_temp_rng_c
station_max_temp_c
```

To improve the model forecasting accuracy, we attempted to understand the relationship between the independent variables and their relationship with the dependent variable and plotted the correlation matrix as follows:

```
corrplot(cor(iq_naf_data), method="color", type="lower", number.cex = 0.6, tl.cex = 0.55, tl.srt = 45, addCoef.col = "black")
```



(Figure 3): Correlation matrix plot for all the independent variables of San Juan city  
After a thorough analysis, we found that though stepAIC used the relevant independent

variables, some among them are still highly correlated. From the above correlation plot in figure 3, we observe many independent variables are highly correlated with each other. So, to eliminate the highly correlated independent variables, we decided to keep the following variables in our model for city IQ.

weekofyear

ndvi\_ne

ndvi\_sw

precipitation\_amt\_mm

reanalysis\_avg\_temp\_k

reanalysis\_dew\_point\_temp\_k

reanalysis\_precip\_amt\_kg\_per\_m2

station\_min\_temp\_c

station\_precip\_mm

Iquitos Peru ~ Independent variables:

Weekofyear—This variable tracks the week wise number of total dengue cases according to the meteorological and vegetation conditions.

ndvi\_ne and ndvi\_sw — Both the variables provide the part of the city Iquitos, where the highest amount of the vegetation land is situated.

precipitation\_amt\_mm and station\_precip\_mm — These two variables tell us how much water the earth receives from rainfall and snowfall. (precipitation)

reanalysis\_avg\_temp\_k, reanalysis\_dew\_point\_temp\_k, station\_min\_temp\_c — Both the variables provide the information about the temperature and mean temperature of the city round the year, to understand which temperature range helps the dengue virus to spread around.

reanalysis\_precip\_amt\_kg\_per\_m2— This variable provides information about precipitation in terms of moisture or water vapor present in the air.

## ARIMA and ETS

For our initial modeling approach, we tested out ets (which is the exponential smoothing state space model from Hyndman's forecasting book) and ARIMA on the time series of the total\_cases. Unfortunately, the approaches shown in class did not allow for a very robust ETS or ARIMA model to use in this competition.

After some digging, we did figure out how to get the ARIMA model to take into account the predictors by using the xreg argument. For our first model, we included all of the predictors for each city, and came up with the following auto.arima models in r.

```
Series: dengTotalCases.IQ$total_cases
Regression with ARIMA(1,0,4) errors
sigma^2 estimated as 47.21: log likelihood=-1727.79
AIC=3507.57   AICc=3510.42   BIC=3618.17
```

```
Series: dengTotalCases.SJ$total_cases
Regression with ARIMA(2,0,1) errors
sigma^2 estimated as 170.9: log likelihood=-3723.53
AIC=7497.06   AICc=7498.49   BIC=7618.1
```

Here are the results from using with our reduced predictor sets as described above:

```
Series: dengTotalCases.IQ$total_cases
Regression with ARIMA(1,0,4) errors
sigma^2 estimated as 48.14: log likelihood=-1739.06
AIC=3506.11   AICc=3506.95   BIC=3565.67
```

```
Series: dengTotalCases.SJ$total_cases
Regression with ARIMA(2,0,1) errors
sigma^2 estimated as 171.9: log likelihood=-3733.34
AIC=7488.67   AICc=7488.96   BIC=7541.93
```

(Note that AIC have both improved, and now have lower degrees of freedom.)

We also tried doing BoxCox transforms on both cities total\_cases time series, which ended up with much better AIC values

```
Series: dengTotalCases.IQ$total_cases
Regression with ARIMA(0,1,1) errors
```

```
Box Cox transformation: lambda= 0.3112776  
sigma^2 estimated as 2.337: log likelihood=-946.07  
AIC=1936.15 AICc=1938.19 BIC=2029.69
```

```
Series: dengTotalCases.SJ$total_cases  
Regression with ARIMA(2,1,2) errors  
Box Cox transformation: lambda= 0.3649882  
sigma^2 estimated as 1.353: log likelihood=-1455.88  
AIC=2961.76 AICc=2963.19 BIC=3082.78
```

Upon on our submission to driven data, the results were not as good as the non-transformed values. Also, the Arima models with the xreg formulas as determined by our analysis of multi-collinearity did not perform as well on the competition site as the ARIMA model with all predictors included in the model. This is not in line with our expectation with the better AICs in all cases.

Submission of ARIMA with all predictors, non-transformed:

## Submissions

BEST SCORE	CURRENT RANK	# COMPETITORS	SUBS. TODAY
26.8510	688	2371	1 / 3
EVALUATION METRIC			

$$MAE = \frac{1}{n} \sum_{i=1}^n |f_i - y_i|$$

## Linear Regression:

Using the above mentioned independent variables for two cities, we developed the linear regression model as mentioned below.

We also predicted the "total\_cases" for the test data of both the cities.

We found that some of the predicted values are negative, which is impossible in real-world, so, we replaced the negative values with zero.



```

#SJ city
lm_fit_sj <- lm(total_cases ~ weekofyear+
                ndvi_nw+ndvi_se+
                reanalysis_relative_humidity_percent+reanalysis_specific_humid
ity_g_per_kg+
                station_min_temp_c+station_max_temp_c,
                data = sj_naf_data)

x <- round(predict(lm_fit_sj,sj_test_data),digits=0)

for(i in 1:length(x)){
  x[i] <- ifelse(x[i]<0,0,x[i])
}

write.csv(x,"sj.csv")

#IQ city
lm_fit_iq <- lm(total_cases ~ weekofyear+
                ndvi_ne+ndvi_sw+precipitation_amt_mm+reanalysis_avg_temp_k+
                reanalysis_dew_point_temp_k+
                reanalysis_precip_amt_kg_per_m2+
                station_min_temp_c+station_precip_mm,
                data=iq_naf_data)

y <- round(predict(lm_fit_iq,iq_test_data), digits=0)

for(i in 1:length(y)){
  y[i] <- ifelse(y[i]<0,0,y[i])
}

write.csv(y,"iq.csv")

```

We submitted the forecasted "total\_cases" to drivendata competition and obtained the following score:

## Submissions

BEST SCORE	CURRENT RANK	# COMPETITORS	SUBS. TODAY
28.4135	821	2380	1 / 3

## Random Forest

To improve the accuracy of predictions, we have used advanced modelling technique "RandomForest" from the caret package.

Random Forests are trained via the bagging method. Bagging or Bootstrap Aggregating, consists of randomly sampling subsets of the training data, fitting a model to these smaller data sets, and aggregating the predictions. This method allows several instances to be used repeatedly for the training stage given that we are sampling with replacement. Tree bagging consists of sampling subsets of the training set, fitting a Decision Tree to each, and aggregating their result.

The Random Forest method introduces more randomness and diversity by applying the bagging method to the feature space.

Reference: (Reinstein, %. (2017, October). KDnuggets. Retrieved December 11, 2017, from <https://www.kdnuggets.com/2017/10/random-forests-explained.html>)

```
formula_sj<- total_cases ~ weekofyear+
              ndvi_nw+ndvi_se+
              reanalysis_relative_humidity_percent+reanalysis_specific_humid
ity_g_per_kg+
              station_min_temp_c+station_max_temp_c

newm_sj <- train(formula_sj, data = sj_naf_data, method = "rf",
                 preProc = c("center", "scale"))

print(newm_sj)

## Random Forest
##
## 936 samples
##   7 predictor
##
## Pre-processing: centered (7), scaled (7)
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 936, 936, 936, 936, 936, 936, ...
## Resampling results across tuning parameters:
```

```
##
## mtry RMSE Rsquared MAE
## 2 44.53738 0.2705477 24.36430
## 4 45.85386 0.2460935 24.72674
## 7 47.14277 0.2249627 25.09987
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was mtry = 2.

x <- round(predict(newm_sj,sj_test_data),digits=0)

for(i in 1:length(x)){
  x[i] <- ifelse(x[i]<0,0,x[i])
}

write.csv(x,"sj.csv")

formula_iq <- total_cases ~ weekofyear+
  ndvi_ne+ndvi_sw+precipitation_amt_mm+reanalysis_avg_temp_k+
  reanalysis_dew_point_temp_k+
  reanalysis_precip_amt_kg_per_m2+
  station_min_temp_c+station_precip_mm

newm_iq <- train(formula_iq, data = iq_naf_data, method = "rf",
  preProc = c("center", "scale"))
print(newm_iq)

## Random Forest
##
## 520 samples
## 9 predictor
##
## Pre-processing: centered (9), scaled (9)
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 520, 520, 520, 520, 520, 520, ...
## Resampling results across tuning parameters:
##
## mtry RMSE Rsquared MAE
## 2 10.46662 0.07252879 6.465190
## 5 10.60382 0.07443037 6.552994
## 9 10.80570 0.06633057 6.634360
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was mtry = 2.

y <- round(predict(newm_iq,iq_test_data),digits=0)

for(i in 1:length(y)){
  y[i] <- ifelse(y[i]<0,0,y[i])
}
```

```
write.csv(y,"iq.csv")
]
```

## DengAI: Predicting Disease Spread

HOSTED BY DRIVENDATA

### Submissions






BEST SCORE	CURRENT RANK	# COMPETITORS	SUBS. TODAY
25.4760	416	2381	0 / 3

#### EVALUATION METRIC

$$MAE = \frac{1}{n} \sum_{i=1}^n |f_i - w_i|$$

The metric used for this competition is mean absolute error. The absolute error is calculated for each label in the submission and then averaged across the labels. For more information on how to calculate MAE, see [wikipedia](#), [sklearn](#) in Python, or the [Metrics](#) package in R. A lower score is better. The goal is to minimize MAE.

#### SUBMISSIONS

Score	Submitted by	Timestamp
28.4135	Romy 	Dec. 9, 2017, 10:52 p.m.
25.4760	Romy 	Dec. 9, 2017, 11:45 p.m.
25.6058	Romy 	Dec. 10, 2017, 12:05 a.m.
26.0769	Romy 	Dec. 10, 2017, 12:17 a.m.
25.7308	Romy 	Dec. 10, 2017, 12:29 a.m.

Make new submission

## Model Comparison of Linear Multiple Regression and Random Forest

To compare linear regression and random forest models, we split the data into 80-20 ratio for train and test data and used the RMSE and MAE metrics.

```
#Split of SJ dataset into train and test
set.seed(100)
train_test_split_sj <- initial_split(sj_naf_data, prop = 0.8)
train_sj <- training(train_test_split_sj)
test_sj <- testing(train_test_split_sj)
#Split of IQ dataset into train and test
set.seed(100)
train_test_split_iq <- initial_split(iq_naf_data, prop = 0.8)
train_iq <- training(train_test_split_iq)
test_iq <- testing(train_test_split_iq)

#Linear regression model for SJ
train_lm_fit_sj <- lm(total_cases ~ weekofyear+
                      ndvi_nw+ndvi_se+
                      reanalysis_relative_humidity_percent+reanalysis_specific_humid
ity_g_per_kg+
                      station_min_temp_c+station_max_temp_c,
                      data = train_sj)

x <- round(predict(train_lm_fit_sj,test_sj),digits=0)

for(i in 1:length(x)){
  x[i] <- ifelse(x[i]<0,0,x[i])
}

rmse(test_sj$total_cases,x)

## [1] 34.94618

mae(test_sj$total_cases,x)

## [1] 26.76471

#Linear regression model for IQ
train_lm_fit_iq <- lm(total_cases ~ weekofyear+
                      ndvi_ne+ndvi_sw+precipitation_amt_mm+reanalysis_avg_temp_k+
                      reanalysis_dew_point_temp_k+
                      reanalysis_precip_amt_kg_per_m2+
                      station_min_temp_c+station_precip_mm,
                      data=train_iq)

x <- round(predict(train_lm_fit_iq,test_iq),digits=0)
```

```

for(i in 1:length(x)){
  x[i] <- ifelse(x[i]<0,0,x[i])
}

rmse(test_iq$total_cases,x)

## [1] 10.09854

mae(test_iq$total_cases,x)

## [1] 6.097087

#RandomForest Model for SJ

formula_sj<- total_cases ~ weekofyear+
               ndvi_nw+ndvi_se+
               reanalysis_relative_humidity_percent+reanalysis_specific_humid
ity_g_per_kg+
               station_min_temp_c+station_max_temp_c

train_newm_sj <- train(formula_sj, data = train_sj, method = "rf",
                       preProc = c("center", "scale"))

x <- round(predict(train_newm_sj,test_sj),digits=0)

for(i in 1:length(x)){
  x[i] <- ifelse(x[i]<0,0,x[i])
}

rmse(test_sj$total_cases,x)

## [1] 34.10506

mae(test_sj$total_cases,x)

## [1] 22.55615

#RandomForest Model for IQ

formula_iq <- total_cases ~ weekofyear+
               ndvi_ne+ndvi_sw+precipitation_amt_mm+reanalysis_avg_temp_k+
               reanalysis_dew_point_temp_k+
               reanalysis_precip_amt_kg_per_m2+
               station_min_temp_c+station_precip_mm

newm_iq <- train(formula_iq, data = train_iq, method = "rf",
                 preProc = c("center", "scale"))

y <- round(predict(newm_iq,test_iq),digits=0)

for(i in 1:length(y)){
  y[i] <- ifelse(y[i]<0,0,y[i])
}

```

```

}

rmse(test_iq$total_cases,y)
## [1] 9.989801

mae(test_iq$total_cases,y)
## [1] 5.990291

```

**Table 1: Models comparison metrics table**

	San Juan city		Iquitos Peru city	
	RMSE	MAE	RMSE	MAE
Linear Regression Model	34.96	26.76	10.09	6.09
Random Forest Model	34.10	22.55	9.98	5.99

As we have already validated our models with given test data set, we analyzed the model's accuracy. For both the cities San Juan and Iquitos Random Forest method provides more accurate prediction. The RMSE and MAE using Random Forest method for San Juan city are 34.10 and 22.55, which are lower than 34.96 and 26.76 for RMSE and MAE accordingly, using the linear regression method. The RMSE and MAE using Random Forest method for Iquitos city are 9.98 and 5.99, which are lower than 10.09 and 6.09 for RMSE and MAE accordingly, using the linear regression method. Also same is confirmed after submitting the final models results at drivendata.com DengAI competition. Thus the model built on variables following the above mentioned journals and using the Random Forest method provides the best predictions in our case.

## Discussion and Recommendations

The analysis of the variables definitely shows correlation between the number of cases of dengue and vegetation, air temperature and moisture conditions. The optimal range for dengue outbreak is approximately 28 °C, 80% relative humidity, and .1 to .2 NVDI (which is the normalized and differenced pixels from the satellite imagery).

With these two particular cities, when those conditions are forecast to occur, it is advised that actions be taken to minimize or prevent mosquito breeding activity, so as to minimize the impact on dengue outbreaks. Also, any cities that at risk of developing conditions like this, it would be advised to ensure health officials are aware that dengue may be more likely to occur than previously in those regions, and take actions that will help in prevention.

Producing accurate and actionable forecasts of total cases of a disease outbreak in short term or long term will improve public health response to disease out breaks. Real time forecast of disease out breaks will help to understand the cause and effect of outbreaks and improve the intervention and prevention strategies.