SPATIOTEMPORAL PREDICTION MODELING OF DENGUE FEVER IN COLOMBIA, SOUTH AMERICA BASED ON TEMPERATURE AND PRECIPITATION

by

Megan Elaine Sirbaugh

A thesis submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Master of Science in
Earth Sciences

Charlotte

2017

Approved by:

____
Dr. Matthew Eastin

____
Dr. Eric Delmelle

____
Dr. Wenwu Tang

ABSTRACT

MEGAN ELAINE SIRBAUGH. Spatiotemporal prediction modeling of dengue fever in
Colombia, South America based on temperature and precipitation. (Under the direction of
DR. MATTHEW EASTIN)

Dengue fever is a prominent mosquito-borne viral disease in the tropics that is estimated to infect as many as 400 million people per year. Dengue is endemic to Colombia, South America and it is crucial to be able to predict when outbreaks may occur so that preventative measures may be taken. The primary vector for the virus, the *Aedes aegypti* mosquito, requires warm temperatures and standing water to live, breed and incubate the virus. Therefore, weather variables such as temperature and precipitation correlate to dengue incidence and can be used to predict the timing and location of dengue outbreaks. While most of the previous research on this topic has focused on temporal prediction of dengue outbreaks in a small area, a spatiotemporal prediction model for the entire country of Colombia was developed for this study using correlations between weather variables and dengue fever incidence data. Temperature and precipitation data from weather stations across Colombia was interpolated via Inverse Distance Weighting, Kriging and Cokriging. Then a prediction model based on the auto-regressive moving average model was developed to compare dengue incidence to each weather variable and to itself at different time lags then to predict future dengue incidence. The accuracy of the prediction model depended on which variables were incorporated into the model, but the most accurate model was the model that only took historical dengue incidence into account. The model performed better in cities than over the country as a whole, which is notable because the majority of cases occur in highly populated areas. Model prediction errors were high, and it is important to note that socio-economic factors, as well as environmental factors, need to be taken into account to create an accurate prediction model.

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Matthew Eastin, for developing the idea for this thesis and for his support and endless assistance in every step along the way. I would also like to thank my committee members, Dr. Eric Delmelle and Dr. Wenwu Tang for their invaluable input and their guidance in the geographical aspects of the project. Dr. Irene Casas provided the dengue incidence data, without which this project would not have been possible. Thanks to my fellow graduate students, Matt Gropp and Warren Pettee, who generously lent their superior programming knowledge to teach me skills I needed to successfully create my dengue prediction model.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

INTRODUCTION

Dengue fever is a mosquito-borne viral disease prominent in the tropics. The Centers for Disease Control and Prevention (CDC) reports that dengue may infect as many as 400 million people each year. Dengue is endemic to over 100 countries, putting nearly half of the world's population at risk. Dengue causes flu-like symptoms such as high fever, severe headaches and body aches, nausea, vomiting and rash. Severe dengue is a potentially deadly form of the disease which causes plasma leaking, fluid accumulation, severe bleeding or organ impairment, and infects as many as 500,000 people each year (Durbin 2016). The four serotypes of the dengue virus, known as DENV-1, DENV-2, DENV-3 and DENV-4, are passed between humans by female mosquitoes of the *Aedes* genus (primarily *Aedes aegypti)*.

The occurrence and spread of dengue is affected by environmental factors because the dengue vector, the female *Ae. aegypti* mosquito, can only survive and incubate the virus under certain environmental conditions. *Ae. aegypti* requires standing water to breed, thus ample precipitation is a necessary condition for dengue to spread. Warm temperatures enable the virus to incubate faster in the mosquito and increase oviposition rates of the mosquito (Rueda et al. 1990, Watts et al. 1987; Yang et al. 2009). These conditions are omnipresent throughout the tropical latitudes, whereas cooler temperatures and drier air in poleward locations hinder the ability of *Ae. aegypti* to spread the virus, hence why dengue is largely confined to the tropics.

Until late 2015, there was no approved vaccine to prevent dengue (Durbin 2016; Pitisuttithum and Bouckenooghe 2016). Dengvaxia, the first licensed vaccine for preventing dengue, was developed by Sanofi Pasteur, the vaccines division of Sanofi, a leading global healthcare company. The vaccine targets all four dengue strains, and was found to reduce incidence of confirmed dengue by 65.6% in trial participants during the first 25 months after injection. Three years after the first injection, the risk of hospitalization was reduced by 50%. There were no safety concerns identified for trial participants over 9 years old. Dengvaxia is currently approved for persons ages 9-45 in eleven countries: Mexico, Brazil, the Philippines, El Salvador, Paraguay, Guatemala, Peru, Costa Rica, Indonesia, Thailand and Singapore. The vaccine is still not licensed in Colombia, so other preventative measures must be taken such as the use of mosquito nets, elimination of mosquito breeding grounds and pesticide spraying. In order to implement such preventative measures in time for them to be effective at preventing outbreaks, we must understand how environmental factors influence the occurrence and spread of dengue fever so that the timing of future outbreaks can be accurately predicted.

All four serotypes of dengue are endemic to Colombia and dengue poses a health risk to much of the country, so it is important to research the correlations between weather and dengue incidence across Colombia. The problem this research addresses is to assess spatiotemporal correlations between temperature and precipitation with the documented cases of dengue fever at the department (equivalent to U.S. states) level in Colombia, South America for the years 2005-2012. Then these spatiotemporal correlations will be used to create a prediction model that will forecast outbreaks of dengue fever across Colombia for the years 2013-2015. Most previous studies created models to temporally predict dengue outbreaks for a single city, so this research will have the benefit of a spatial prediction component in the model. Such a prediction model could be incorporated into an Early Warning System (EWS) for Colombia, so that the time and place of outbreaks of dengue could be accurately predicted, and steps could be taken well ahead of time to prevent outbreaks from occurring.

An additional motivation for this research is that with global climate change and the increasing temperature trends, there is a concern that dengue may be able to spread to new regions as the suitable environment for *Ae. aegypti* expands latitudinally and altitudinally. Some studies have shown that the range of *Ae. aegypti* could expand into the southern United States (Hopp and Foley, 2003; Jetten and Focks 1997). Developing an EWS for an endemic area, such as Colombia, will have the added benefit of allowing non-endemic regions to be better prepared for any spatial expansion of tropical vector-borne diseases into new areas.

History

Dengue became a pandemic following World War II when epidemic transmission of dengue in Southeast Asia increased and the first epidemics of severe dengue (then known as dengue hemorrhagic fever, or DHF) were recorded in Manila, Philippines in 1953 (Gubler 1998). Dengue and DHF proceeded to spread throughout Asia. Dengue was reintroduced to the Pacific Islands in the 1970s, from where it quickly spread to the tropical latitudes of the Americas. Mosquito eradication programs that had been in place in the American region were discontinued in the 1970s because of the dangers associated with pesticide use, allowing *Ae. aegypti* to reinvade many countries in Central and South America. By the 1980s dengue epidemics began to sweep through countries that had been free of the disease for decades, prompting a surge of studies on the factors that influenced dengue outbreaks and how outbreaks could be prevented.

The Dengue Vector: The Female *Aedes aegypti* Mosquito

The principal dengue vector, *Ae. aegypti* is a domesticated tropical mosquito that feeds primarily on humans in daylight hours (Gubler 1998). It lays its eggs in sitting water in artificial containers such as tires, rainwater buckets, flower vases and containers used for water storage such as 55 gallon drums and septic tanks (Barrera 2011; Carbajo 2012). This enables the *Ae. aegypti* to become highly populous in tropical urban settings and near human homes. The virus is transmitted when the female *Ae. aegypti* mosquito takes a blood meal from an infected human, then takes another blood meal from an uninfected human after the virus incubates within the mosquito. The extrinsic incubation period of the mosquito can vary based on temperature, as well as the mosquito's willingness to bite.

Correlations between Temperature and Mosquito Life Cycle

Many laboratory studies have shown important correlations between temperature and mosquito life cycle, as well as temperature and the ability of the mosquitoes to spread the dengue virus. In a controlled laboratory environment, the optimal temperature range for *Ae. aegypti* to survive in the aquatic stage is 15-35 °C, and 15-30 °C for adults (Yang et al. 2009). The highest rates of survival to the adult stage occurs between 20 and 27 °C, with major drop-offs in survival at 15 °C and slight drop-offs above 30 °C (Rueda et al. 1990). The temperature at which oviposition rates are the highest is at 26 °C (Yang et al. 2009). More recently, it has been studied whether diurnal temperature range has an effect on the life cycle of *Ae. aegypti* and the transmission of the dengue virus (Lambrechts et al. 2011; Eastin et al. 2014). Average daily temperatures may not accurately predict the individual temperatures that occurred on a day. In 2011, Lambrechts et al. conducted a laboratory study in which mosquitoes were held in conditions that changed temperature by varying degrees throughout the day. No temperature change was used as the control group, and all groups had the same average temperature. Larger diurnal temperature range reduced the probability of *Aedes aegypti* survival through the time it takes for the virus to spread through the mosquito to its salivary glands so that it can infect humans, known as the extrinsic incubation period, and led to a lower rate of infection within the mosquitoes. It may be more important to assess how daily temperature ranges, rather than average daily temperatures, affect dengue transmission.

Vector efficiency after exposure to different temperatures has also been tested in the lab. *Ae. aegypti* is less mobile and struggles to take blood meals when temperatures are below 15 °C (Christophers 1960; Yang et al. 2009). *Ae. aegypti* is more likely to bite when temperatures are higher (Christophers 1960; Scott et al. 2000). In 1987 Watts et al. assessed the effect of temperature on the vector efficiency of the *Ae. aegypti* mosquito at spreading the DEN-2 virus. *Ae. aegypti* mosquitoes obtained from Bangkok, Thailand were allowed to feed on viremic adult rhesus monkeys and were subsequently grouped and kept at set temperatures that approximated the hot-dry, rainy and cool-dry seasons in Bangkok. For differing number of days after the blood meal, DEN-2 virus transmission was attempted at room temperature by allowing the potentially infected mosquitoes to feed on uninfected monkeys. Transmission was attempted at several different set amounts of days after feeding on the viremic (infected) monkeys. In turn, blood from the monkeys was tested for the presence of dengue virus. The extrinsic incubation period, was 7 days for mosquitoes kept at higher temperatures (32 or 35°C). Conversely, the extrinsic incubation period was longer (12 days) for mosquitoes kept at 30°C, and no transmission occurred from mosquitoes kept at temperatures below 30 °C. At higher temperatures when the incubation period is shorter, transmission to humans can occur faster. This indicates that in the world outside of the laboratory, several days with temperatures in the 32-35°C range could be a factor in the onset of a dengue outbreak. Laboratory tests have been important to understanding the conditions under which *Ae. aegypti* can survive, breed and incubate the dengue virus and their results have been supported by correlations between reported dengue cases and recorded temperature and precipitation.

Correlations between Weather and Dengue Incidence

The results of idealized laboratory studies have been confirmed in subsequent studies of *Ae. aegypti* and dengue incidence in the field. Significant associations between dengue incidence and weather variables, such as several temperature and rainfall parameters, have been consistently reported in the literature (Descloux et al. 2012; Eastin et al 2014; Hii et al. 2012; Wu et al. 2007).

Many field studies have been conducted in Southeast Asia, where dengue is endemic to the majority of countries. Wu et al. applied an autoregressive integrated moving average (ARIMA) model to Kaohsiung, in southern Taiwan to model how dengue fever incidence correlated to temperature and humidity over time between the years 1998-2003. ARIMA is a common way to make short term forecasts by analyzing historical data, then using the correlations found to make predictions. Significant associations between maximum monthly temperature, minimum monthly temperature, relative humidity, monthly rainfall and dengue incidence were found. The strongest correlations were between temperature and dengue incidence and relative humidity and dengue incidence at a time lag of two months (Wu et al. 2007). A similar study by Hii et al. (2012) predicted dengue cases based on weather variables in Singapore for the years 2000-2011. A time series Poisson multivariate regression model was used to analyze the correlations between weekly dengue cases and daily mean temperature and rainfall data. These correlations were used to make a predictive model for 2011-2012 which proved to well-predict dengue outbreaks that were actually observed during that time period. The model accurately predicted 5 out of 5 outbreak weeks in 2011 using a 16 week time lag, which, in practice, would be ample lead time for health agencies and citizens to prepare and try to prevent outbreaks. Descloux et al.'s 2012 study on dengue in Noumea, New Caledonia, a French territorial island off the east coast of Australia, reported that the peak of epidemics occurred 1-2 months after the warmest temperature and simultaneously with the maximum precipitation and relative humidity.

More recent studies have taken diurnal temperature range into account. In 2014, Eastin et al. utilized two unprecedented weather variables in weather-dengue research, mean daily temperature range and number of days with measurable rainfall. One goal of the study was to determine whether these parameters were more relevant to vector mosquito survival than daily maximum and minimum temperatures and total rainfall, respectively. Temporal dengue incidence prediction was completed using an ARIMA model. Mean daily temperature range was found to be the most accurate predictor of dengue incidence from 2001-2011 (Eastin et al. 2014). Studies conducted using observed dengue and weather data in multiple tropical locations found that the most significant correlations have been found between temperature variables and dengue incidence, and agree with the laboratory studies that warmer temperatures lead to higher dengue incidence. Correlations between precipitation and dengue incidence have also been found. This study will extend the results of previous work to a spatiotemporal context.

Dengue Incidence and Climate Change

As global climate change progresses and the increasing temperature trend continues, there is a concern that dengue may be able to spread to new regions as the suitable environment for *Ae. aegypti* expands latitudinally and altitudinally. Several studies have attempted to quantify the effects of climate warming on the range of *Ae. aegypti* (Hopp et al. 2003, Jetten & Focks 1997). By taking into account the ideal temperature range for the mosquito to live, it has been found that a global temperature increase of 2°C would increase dengue rates in some parts of southern North America, southern Europe, South America, Sub-Saharan Africa, India, Southeast Asia and Australia by two to ten times (Jetten & Focks 1997). This increase in temperature would also introduce new locations where dengue transmission would be possible in the previously listed regions (Jetten & Focks 1997). A temperature change of 4°C increases dengue transmission rates and new areas of potential transmission even more (Jetten & Focks 1997). With increased temperature, the potential area for dengue transmission would expand in latitude and altitude and the length of the transmission season would be longer.

Hopp et al. conducted a similar study in which a numerical model was used to simulate how *Aedes aegypti* may have responded to climate changes that were observed between 1958 and 1995 and to see how these potential mosquito environments compare to the locations of dengue cases. Precipitation, relative humidity, temperature, and solar radiation changes from 1958 to 1995 were analyzed on a global scale and the model analyzed areas that were potential vector habitats at each time step. Throughout smaller countries in Southeast Asia, and the Americas, there were statistically significant correlations between mosquito density, as predicted by the model, and observed dengue cases, thus, this model, when applied to climate predictions could be used as a predictor of vector habitat expansion. The correlation was not significant for larger countries such as the United States, China, Brazil and Australia because outbreaks are typically contained in only small portions of these countries.

Climate change has caused growing concern of dengue spreading to new areas. This threat increases the need for research into predictive models so that outbreaks can be prevented in areas that will become at-risk for becoming endemic dengue regions. Predictive models that have a spatial component will be especially important in analyzing where dengue outbreaks are likely to occur as climate change continues.

Temporal and Spatiotemporal Analysis

Most dengue modeling studies have focused on temporal variations in dengue incidence rather than spatial or spatiotemporal variations. The correlations found in the aforementioned field studies helped the researchers to develop prediction models that can be integrated into early warning systems (EWS) for tropical areas. Most current research on the topic is geared towards creating temporal prediction models of dengue outbreaks for endemic cities (Descloux et al. 2012, Eastin et al. 2014, Hii et al. 2012, Wu et al. 2007). In these studies, first, analytical models were developed to detect correlations between weather variables and dengue cases. Then these correlations were applied to predictive models, which analyzed weather data and predicted when dengue outbreaks would occur. All of these previous studies were all successful in predicting dengue outbreaks. Descloux et al.'s predictive model has been integrated into an EWS for Noumea, New Caledonia.

A new avenue of research in this area is taking into account the spatial variation of dengue outbreaks in addition to temporal variation. One such study was conducted by Delmelle et al. for Cali, Colombia and published in 2013. Temporal variation in dengue incidence during the first eight months of 2010 was evaluated using a cumulative distribution function (CDF), which represents the probability that an event will happen on or before a certain date and a probability distribution function (PDF) which represents the probability that an event occurs at a particular time. A high prevalence in February 2010 was found and attributed to lack of rain, high temperature, which reduced incubation time of the virus. K-functions were used to complete the spatial and spatiotemporal analysis and it was found that clusters of dengue cases occurred in areas where there was a concentration of vulnerable populations and areas with favorable mosquito breeding conditions (Delmelle et al. 2013). This study did not use weather variables as predictors of dengue outbreaks, but rather used weather as an explanation for why dengue outbreaks may have occurred when they did.

Because spatial prediction of dengue outbreaks has not been studied extensively, especially in relation to weather variables, the research proposed in this paper aims to fill that gap. This will be achieved by analyzing weather variables (specifically, monthly mean and extreme temperatures and total precipitation) across the entire country of Colombia and calculating correlations between such weather variables and dengue incidence data. With these correlations, a spatiotemporal prediction model will be created that will predict dengue outbreaks across Colombia and over time. Such a model could be integrated into an EWS for Colombia to predict when dengue outbreaks may be likely well ahead of time so that preventative measures can be taken to prevent epidemics.

METHODS

Study Area

Colombia, South America is a large (with an area of 439,737 square miles) and physically diverse (with varying elevations and climate zone) tropical country that straddles the Equator. Along the Pacific coast, to the north is the Caribbean coastal region and to the south is the Pacific coastal region, both of which are characterized by low elevations. Just inland from the Pacific coast is the Andean region, dominated by the Andes Mountains, where elevations reach over 5000 meters. To the east of the Andes lays the Amazon rainforest to the south and the Orinoquia savanna to the north at low elevations. Figure 1 shows a map of land cover in Colombia, where the rainforest region is easily seen in dark green and savanna region in orange. The Andean region and northwest coastal region are dominated by croplands. Land cover data was obtained from the GlobCover 2009, obtained and compiled by the European Space Agency.

Due to its proximity to the equator, Colombia's temperatures do not vary extremely latitudinally or seasonally. However, temperatures do vary altitudinally, with cooler temperatures in the high mountains of the Andes (Ishida and Kawashima 1993). Generally, warm-dry seasons occur from about December-February and June-September and cool-wet seasons occur from about March-May and October-November (Eastin et al. 2014).

Colombia is subdivided into 32 departments (equivalent to U.S. states) and a Capital District, Bogota. Colombia's estimated population as of October 2016 was 48,900,000 people. The population has been steadily increasing over the last 60 years. However, Colombia's population is unevenly distributed throughout the country. Figure 2 shows a map of Colombia’s population distribution. The majority of the population is concentrated in the western half of the country, with very few people living in the Amazon rainforest or Orinoquia savanna regions, which make up the eastern half of the country. The majority of the population is concentrated in the Andean region. Population density likely has an influence on the number of dengue cases that occur in an area. This is in part because *Ae. aegypti* is a highly urbanized mosquito, and because there are more opportunities for blood meals and to spread the virus.

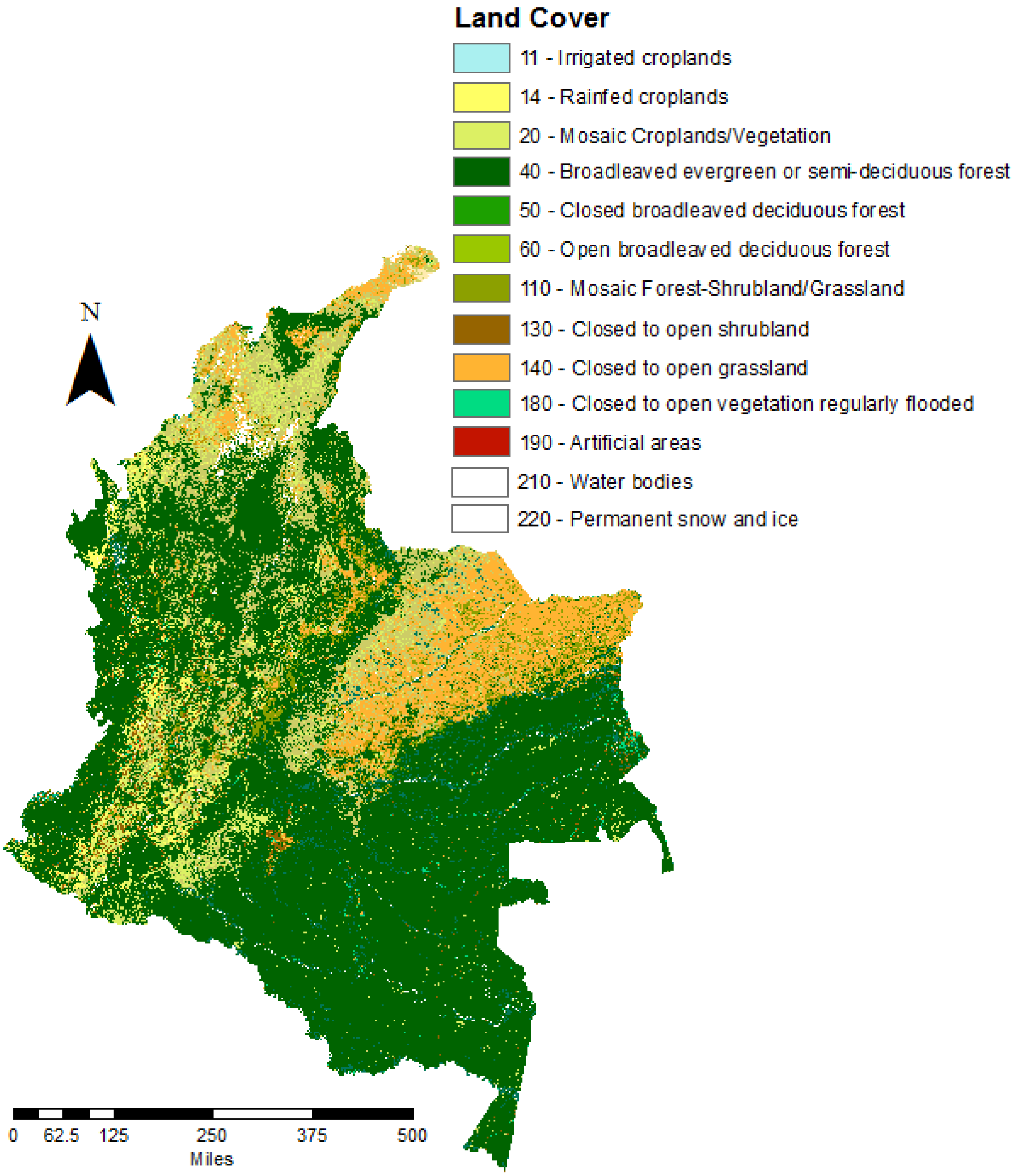


Figure 1. A land cover map of Colombia shows that the eastern portion of the country is dominated by the Amazon Rainforest and Orinoquia Savanna. The majority of the population, however, resides in the built-up and cropland areas of the Andes Mountains and the northwest coast.

Population Data

Population data was obtained from the Socioeconomic Data and Applications Center (SEDAC), a data center in NASA's Earth Observing System Data and Information System. The dataset used was the Gridded Population of the World (GPW), version 4. A GeoTiff of the estimated 2005 population of Colombia at 30" resolution was downloaded and is shown in Figure 3. The population data will be used as a proxy for where dengue cases occur, under the assumption that more cases occur where there are more people. This is to account for the resolution difference between the interpolated weather data being on a grid and the dengue cases being aggregated into departments.

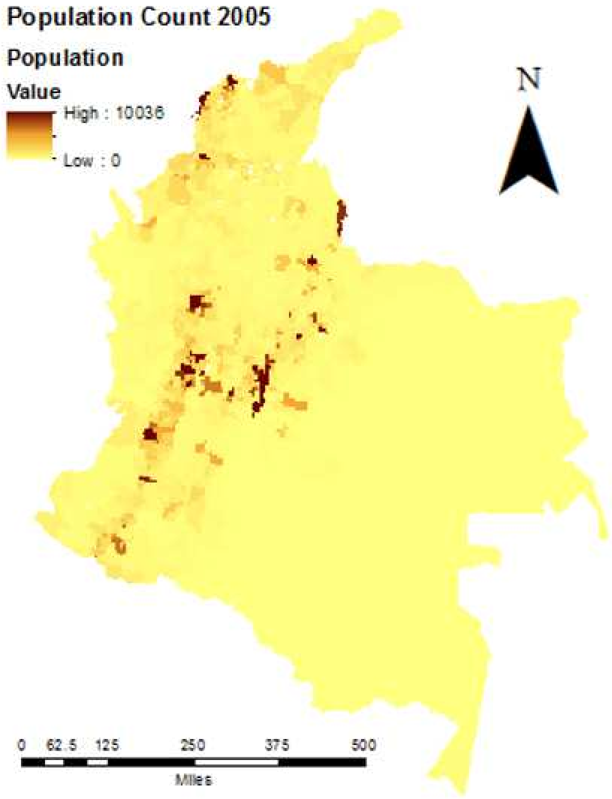


Figure 2. Map of population count in Colombia. The majority of the population is concentrated in the Andean Region and the western coast, whereas very few people live in the eastern portion of the country in the savanna and rainforest regions.

Meteorological Data

The weather data used for this study came from the Monthly Summaries of Global Historical Climatology Network (GHCN)- Daily, which is a dataset maintained by the National Oceanic and Atmospheric Administration (NOAA)'s National Centers for Environmental Information (NCEI) (formerly National Climatic Data Center, NCDC). The GHCN is a database of climate summaries from land surface observation stations around the world (GHCN 2015). Monthly summaries are derived from the GHCN-Daily dataset. The monthly summary provides simple averages or monthly accumulations of daily observations. The variables downloaded for this study were monthly means of maximum temperature, minimum temperature and mean temperature, and total monthly precipitation. The monthly climate summary can be obtained from the NCDC's Climate Data Online (CDO) system. The monthly climate summary was chosen as the dataset for this study because it was the shortest time span for which data was available for Colombian weather stations from CDO. The data can be downloaded as a comma-separated value (CSV) file and manipulated in Microsoft Excel.

Four types of weather data were downloaded from the GHCN, monthly averages of maximum temperature, minimum temperature and mean temperature, and total monthly precipitation. Data from 29 weather stations in Colombia and surrounding countries was collected for each month of each year of the study period, 2000-2015. The location of weather stations is Colombia is not evenly distributed. See Figure 3 for a map of the locations of weather stations used, as well as elevation. Generally, more weather stations exist where there are more people. There are very few weather stations in the heavily forested areas of the country. (Refer to Figure 1 for land cover and Figure 2 for population distribution). Because of the uneven distribution of available weather data, the interpolation of weather data throughout the country will be highly influenced by the location of weather stations. See Table 1 below for the list of weather stations used and their location.

In the downloaded file, the values of precipitation were in tenths of a millimeter and temperature values are in tenths of a degree Celsius. Precipitation was converted to millimeters and temperature was converted to degrees Celsius. Elevation, latitude and longitude information for each station was also included in the downloaded file. Most stations had missing data for some variables in some months. The missing data were excluded from further analysis.

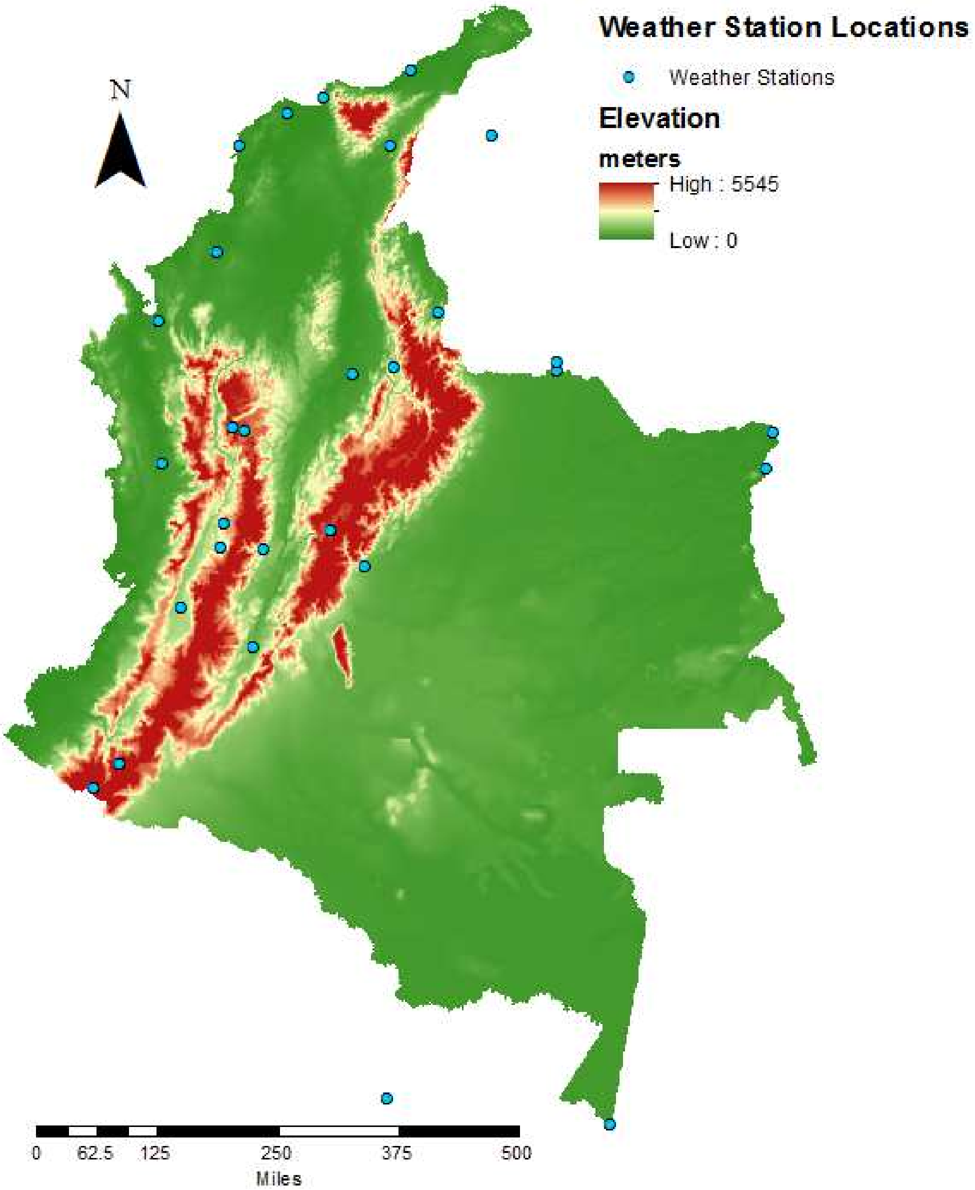


Figure 3. Locations of the 29 weather stations used to gather temperature and precipitation data overlain on a digital elevation model of Colombia. Twenty-five stations were located in Colombia, three in Venezuela and one in Peru. The majority of weather stations are located near population centers in the Andean and coastal regions. The lack of weather stations in the eastern portion of the country is due to lack of population.

Table 1. List of weather stations used in the interpolation, with their location and elevation. For country, CO is Colombia, VE is Venezuela and PE is Peru.

Station Name	Latitude	Longitude	Elevation	Country
APTO A BONILLA AUT	3.54	-76.38	961	CO
APTO ALFONSO LOPEZ	10.4	-73.25	138	CO
APTO ALM PADILLA	11.5	-72.92	4	CO
APTO ANTONIO NARIN	1.2	-77.29	1816	CO
APTO BENITO SALAS	2.95	-75.29	439	CO
APTO CAMILO DAZA	7.93	-72.51	250	CO
APTO EL CARANO	5.69	-76.65	53	CO
APTO EL DORADO	4.7	-74.14	2547	CO
APTO EL EDEN	4.45	-75.77	1229	CO
APTO J M CORDOVA	6.17	-75.43	2073	CO
APTO LOS GARZONES	8.82	-75.82	20	CO
APTO MATECANA	4.81	-75.74	1342	CO
APTO OLAYA HERRERA	6.22	-75.59	1490	CO
APTO PALONEGRO	7.12	-73.18	1189	CO
APTO PERALES	4.42	-75.13	928	CO
APTO RAFAEL NUNEZ	10.4	-75.51	2	CO
APTO SAN LUIS	0.86	-77.67	2961	CO
APTO SIMON BOLIVAR	11.1	-74.23	4	CO
APTO VANGUARDIA	4.16	-73.62	423	CO
APTO VASQUEZ COBO	-4.19	-69.94	84	CO
APTO YARIGUIES	7.02	-73.81	126	CO
ANTONIO ROLDAN BETANCOURT	7.812	-76.716	14	CO
ERNESTO CORTISSOZ	10.89	-74.781	29.9	CO
PUERTO CARRENO A.GU	6.1	-67.49	50	CO
SANTIAGO PEREZ	7.069	-70.737	128	CO
CASIQUE ARAMARE	5.62	-67.606	74.7	VE
GUASDUALITO	7.211	-70.756	129.8	VE
IQUITOS	-3.783	-73.3	126	PE
LA CHINITA INTERNATIONAL	10.558	-71.728	71.6	VE

Epidemiological Data

Dengue incidence data was obtained from Colombia’s Sistema de Vigilancia en Salud Publica (SIVIGILA; English—Public Health Surveillance System) for the study period 2000-2015. SIVIGILA is the system in place in Colombia to which all lab-confirmed dengue cases are reported. Information for each dengue case includes patient information such as sex, age race and neighborhood, dates of diagnosis, first symptoms and hospitalization, final condition and reporting institution (Delmelle et al. 2013; Eastin et al. 2014). SIVIGILA is a reliable source of dengue information that has been in place since the 1960s. However, some bias may be presented in the date of diagnosis because of the days that people are actually able to go to the hospital are limited. This can be accounted for by taking monthly totals of the number of dengue cases (Eastin et al. 2014). Additionally, dengue is typically under-reported because many people who don’t exhibit severe symptoms don’t seek professional medical treatment (Suaya et al. 2007). Dengue incidence data was aggregated into the number of cases per department per month of the study period. See Figure 4 for a map of dengue incidence totals per department in January 2005.

Epidemiological data needed to be assessed at a finer scale than the department level in order to obtain a more precise location where dengue outbreaks may occur. Population-weighted dengue density maps were created and employed to address this issue. The dengue incidence data was overlain on the population data (resolution = 6600 m). First the departmental dengue data was converted to a raster of cell size 6600 m where each cell was given the value of the total dengue count for the department in which that cell belonged. Then the population of each cell was divided by the total population of its department then multiplied by the amount of dengue for that department. This resulted in a 6600 m resolution grid of dengue incidence. Figure 5 shows an example of a population weighted dengue raster from January 2005. Correlations between dengue and weather variables were calculated from this raster.

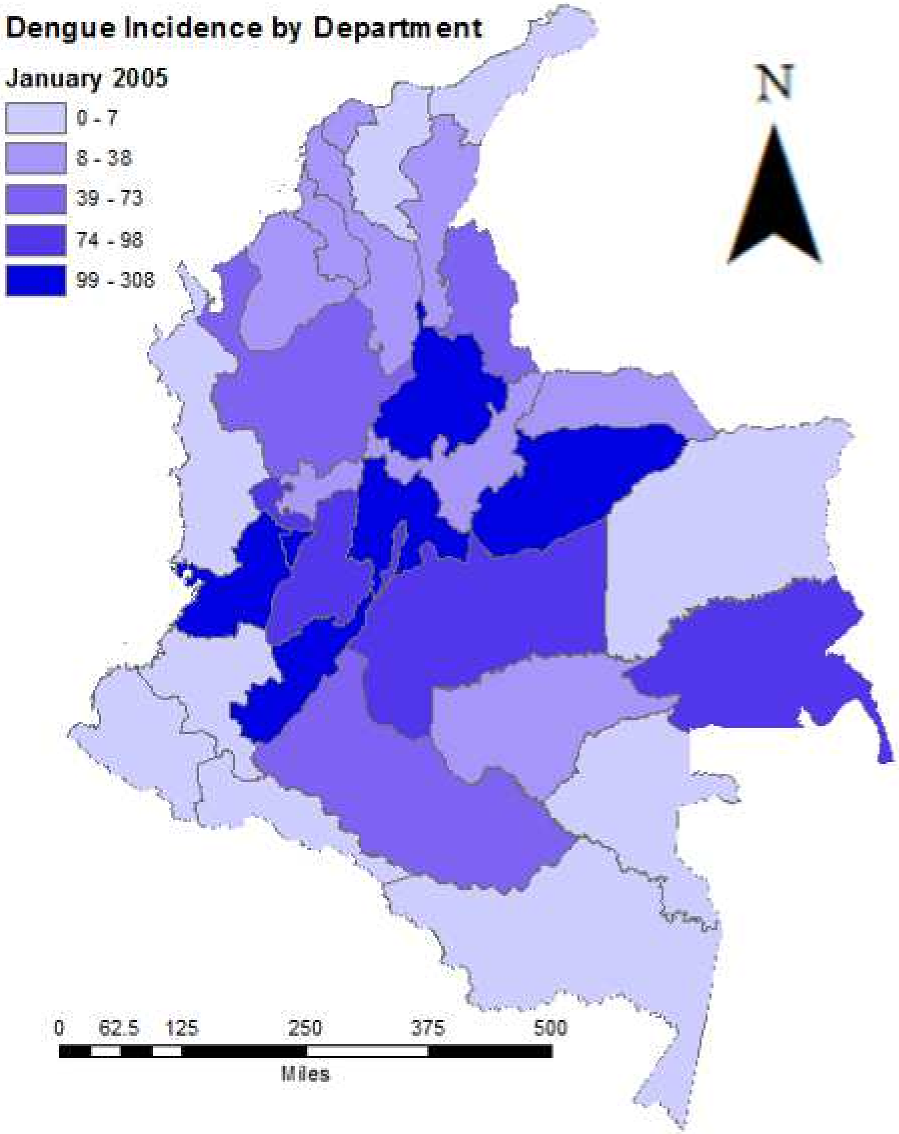


Figure 4. Total number of dengue cases per department in January 2005. Dengue is more concentrated where the population is higher. The eastern portion of the country, in the savanna and rainforest, has few dengue cases.

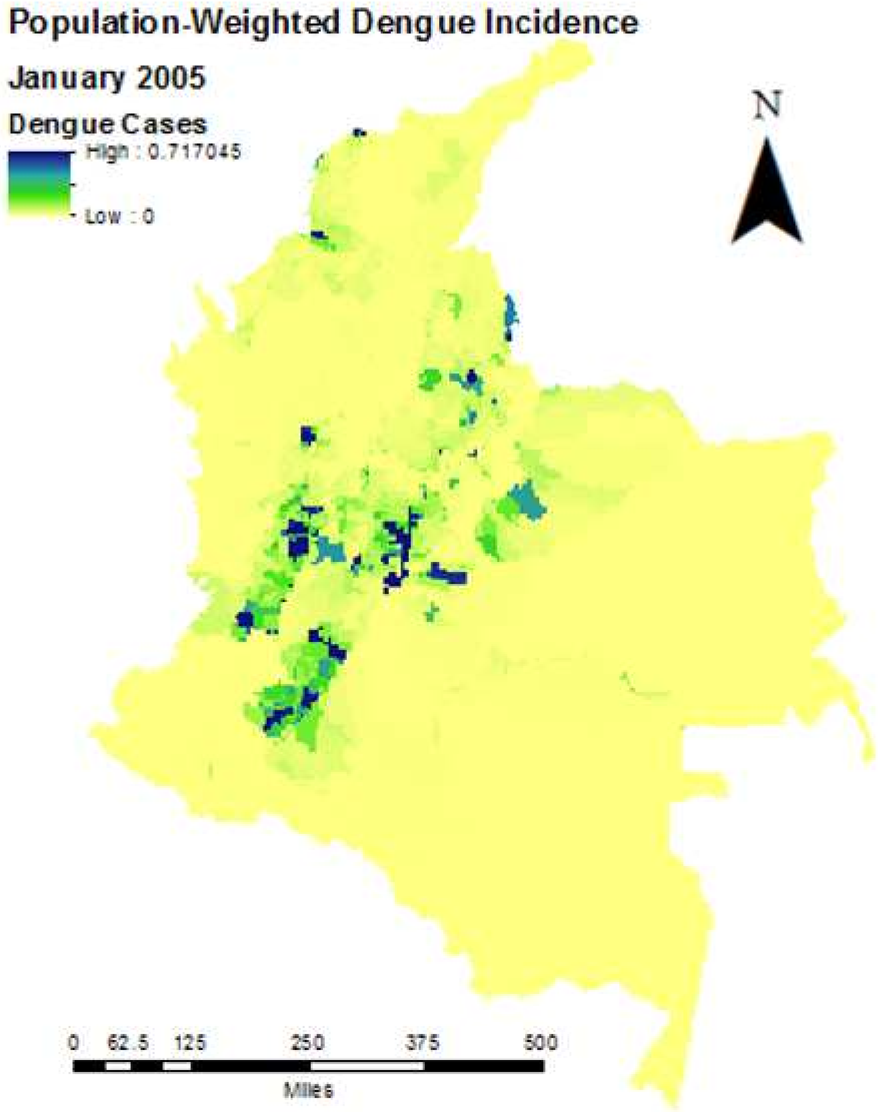


Figure 5. Population-weighted dengue count for January 2005. The number of cases of dengue per 6600 m grid cell is reported.

Interpolation of Weather Data

Spatial interpolation is the procedure by which values of a variable are estimated for an entire area from values observed at sample locations. Because weather stations are not located at every point in Colombia, the observations recorded by the present weather stations must be interpolated throughout the entire country to get an idea of what the temperature and precipitation patterns are like, which can then be compared to dengue incidence. Three common methods of interpolation are inverse distance weighting, kriging and cokriging.

Inverse distance weighting (IDW) is a simple interpolation technique which estimates the value of a variable at a point between two sampled locations by averaging the observed values from surrounding points. Observed values are weighted so that observations that are closer to the estimation point are weighted inversely by distance so that observation closer to the estimation point are weighted more than observations further away. To estimate the temperature at an unsampled location, the observed temperatures from surrounding weather stations were averaged, with the temperatures from the closer weather stations weighing more in the average and temperatures from further stations weighing less. One disadvantage of IDW is that since it only takes averages of surrounding points, an estimated point cannot be lower or higher than any of the surrounding points, as could be the case. Another disadvantage is that if the sampling points are sparse or unevenly distributed throughout the study area, the results of the estimation may not sufficiently represent reality (Watson & Philip 1985). Therefore, the best results from IDW are achieved when the sampled locations are dense and evenly distributed in the study area, which as shown by Figures 6-9, is not the case for this study.

The main advantages of IDW are that it is easy to program and is not computationally expensive.

Kriging is often touted as the most accurate method of spatial interpolation (Li & Heap, 2011). Kriging and cokriging are two common methods of spatial interpolation. Kriging interpolates a variable across space, and co-kriging does the same, while also using a highly correlated auxiliary variable to improve the accuracy of the interpolation. Though more computationally expensive (compared to IDW), kriging interpolation can be completed in programs such as R and ArcGIS.

Ordinary kriging is an interpolation technique which predicts the value of a variable for a location by assigning weights to surrounding measured locations then averaging the surrounding observations and taking their weights into account (Webster, Oliver & Gerard 1989). For every point in an area, a semivariogram is applied, which assigns weights to each surrounding observation (sampled location) by summarizing spatial dependence of the observations at different distances from the unsampled point of interest (Zimmerman 1999). The semivariogram is a function that indicates spatial correlation in observations measured at sample locations, which is commonly represented as a graph that shows the variance in the observed values with distance between all pairs of sampled locations (Clark 1979). Therefore, in ordinary kriging, not only is the distance between the observed point and the unsampled point considered, but the distance between the observed points, the prediction locations and the overall spatial arrangement among the observed points are considered (Webster, Oliver & Gerard 1989). This allows for individual observations within clusters to be weighted less than independent observations.

Ordinary kriging assumes that the mean value of a variable is constant across the study area and that the semivariogram is known. Kriging is considered the optimal interpolator because its estimates are unbiased, have minimum variances and it is an exact interpolator, meaning the kriged value at a sampled point is the measured value there, so variance is zero (Webster, Oliver & Gerard 1989).

Equation (1) shows the equation for ordinary kriging:

$$\hat{z}(B) = \sum_{i=1}^{n} \lambda_i \, z(x_i), \tag{1}$$

where z is the observed value of a property at *n* sites, x_i and λ_i are the weights associated with the sample points, which are summed to one to ensure that the estimates are unbiased (Webster, Oliver & Gerard 1989). Equation (2) shows the estimation variance for the kriging prediction ẑ(B):

$$\boldsymbol{\sigma^2(B) = E[\{\hat{z}(B) - z(B)\}^2] = 2\sum_{i=1}^{n}} \lambda_{\boldsymbol{i}} \boldsymbol{\bar{\gamma}}(x_i, B) - \sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} \sum_{\boldsymbol{j=1}}^{\boldsymbol{n}} \lambda_i \lambda_j \gamma(x_i, x_j) - \bar{\gamma}(B, B)\,, \tag{2}$$

where $\gamma(x_i,x_j)$ represents the semivariance between the *i*th and *j*th sampling points, $\bar{\gamma}(x_i,B)$ is the average semivariance between the block B and the *i*th sampling point, and $\bar{\gamma}(B,B)$ is the average semivariance within the block (Webster, Oliver & Gerard 1989). The estimation variance is minimized when

$$\sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} \lambda_{\boldsymbol{i}} \boldsymbol{\gamma}(x_i, x_j) + \varphi = \boldsymbol{\bar{\gamma}}(x_j, B) \tag{3}$$

where Ψ is a Lagrange multiplier, which achieves minimization of variance (Webster, Oliver & Gerard 1989).

Cokriging is a kriging approach which uses a second variable to provide interpolation estimates, which is especially useful when the auxiliary data is highly correlated to the primary data source (Goovaerts 2000). Temperature is highly correlated with elevation, so elevation from a digital elevation model is used as auxiliary data in this for this study. Cokriging can be especially useful because comprehensive and areal elevation data is more easily obtainable than comprehensive temperature data. Cokriging works well in areas with large elevation variations, so using elevation as auxiliary data in cokriging is common (Yang et al. 2004).

In this study, once the raw weather data was obtained from CDO and organized into a usable format, the files were imported into ArcMap. Within the python environment in ArcMap, shapefiles were created for each weather variable for each month in the study period, from 2005-2015. The shapefiles were sets of points representing each weather station, containing the value of a given weather variable at that station for a given month. These shapefiles were used to complete IDW, ordinary kriging and cokriging of each weather variable for each month.

IDW was completed in R for each of the weather variables, T_{MAX}, T_{MIN}, T_{MEAN} and P_{TOT}, for each month of the eleven-year study period. The IDW function from the R library ‘gstat’ was used to complete the interpolation. The interpolation resulted in maps of each variable for each month which spanned all of Colombia.

Ordinary kriging was completed similarly in R. The autokrige function, part of the *automap* package available in R, was used to automatically generate a variogram and complete the kriging interpolation for each weather variable for every month of the study period. The automatic fitting of the variogram estimates the sill, range and model for the variogram based on the data. Then the data is automatically kriged based on the generated variogram.

Cokriging was completed for each month in ArcMap's Geostatistical Wizard with the weather variables as the primary datasets and the digital elevation model of Colombia as the secondary dataset. Simple cokriging with a prediction output surface was selected for the interpolation. Otherwise, the Geostatistical Wizard settings were kept as the default, because it automatically determines the best semivariogram and cross-variogram.

The accuracy of the IDW, ordinary kriging and cokriging models was assessed using leave-one-out cross validation, in which one sample point is removed from the kriging interpolation and the interpolated value is compared to the observed value at the sample point. This is done repeatedly so that each sample point is removed one at a time and compared to the kriged value at that point. Three measures of error, mean error (ME), root mean squared error (RMSE) and root mean squared standardized (RMSSE) error were calculated to assess the accuracy of the kriging models for each T_{MAX}, T_{MIN}, T_{AVG} and P_{TOT}. Cokriging interpolation was found to have the lowest root-mean-square error, so the cokriged weather maps were used in further analysis.

Dengue Prediction Modeling

Autoregressive moving average (ARMA) models have proven to be a successful approach to forecasting time series based on correlations between weather variables and dengue cases (Eastin et al. 2014; Wu et at. 2007). The ARMA model analyzes time series data then predicts a value in a forecast time series as a linear combination of its own past values and past errors (SAS Institute Inc. 2014). There are three stages to ARMA modeling, which were outlined by Box and Jenkins (1976). First, in the identification stage, autocorrelations and cross-correlations between predictor variables are computed. Second, in the estimations and diagnostic testing stage, the optimized ARMA models are developed along with the required estimation coefficients and diagnostic statistics to assess the adequacy of each model. Third, in the prediction stage future values are predicted and confidence intervals are calculated for each forecast. A Space-Time Autoregressive Integrated Moving Average (STARIMA) model has also been developed, which is valuable for forecasting purposes when a system exhibits spatial autocorrelation (Pfeifer and Deutsch 1980). The STARIMA model is an extension of the univariate ARIMA model into the spatial domain (Pfeifer and Deutsch 1980).

The procedure used in this research was based on and adapted from the ARMA and STARIMA models. The variables used in the prediction models and their abbreviations are summarized in Table 2.

Correlations between dengue and each weather variable were calculated for time lags of one to six months at each pixel (26,360 pixels). For example, at a 1 month time lag, D_{POP} in July was compared to T_{MAX} in June, D_{POP} in August to T_{MAX} in July, and so on throughout the model training period of 2005-2012. At a 6 month time lag, D_{POP} in July was compared to T_{MAX} in January, D_{POP} in August to T_{MAX} in February, and so on throughout the model training period. Then D_{POP} was autocorrelated at time lags of one to six months. At each pixel, the ideal time lag for each weather variable was identified as the lag with the highest absolute value of the correlation coefficient (R). The ideal time lag between the autocorrelated dengue data was also identified (and was always at a time lag of one month).

Once the ideal time lag for each predictor was identified at each pixel, stepwise regression was employed to select which predictors would create a successful prediction model at each pixel without overfitting. Stepwise regression selects the best predictors from a pool of potential predictors based on a set of criteria (Wilks 1995). The criteria used to select predictors for this model were that the predictor increased the proportion of variance explained (R^2) by the model by more than 1% and the p-value of that predictor was less than 0.1. R^2 was chosen to evaluate model performance because it shows the proportion of the variance in dengue incidence that is explained by the model. A cutoff of a 1% increase in R^2 was chosen because if the proportion of variance explained by the model did not increase by at least 1% when adding a new predictor, there is no need to include that variable in the model. A cutoff of $p < 0.1$ ensured the significance of each variable included in the model to the 10% level. R^2, p-value, and regression coefficient (β) were calculated via ordinary least squares regression for each variable at each pixel. An intercept of the regression model and its associated p-value was also calculated at each pixel. The intercept was only included in the model if it was deemed significant in the same fashion as the predictors.

D_{POP} at a one month time lag was always included as the first predictor because in the autoregressive dengue correlations, R was the highest at a one month time lag for every pixel. The p-value of D_{POP} at a one month time lag was always much less than 0.1, and therefore considered significant. Then the weather variables were selected based on their improvement of the correlation coefficient of the model, and the significance of that variable (Wilks 1995). If adding, for example, T_{MAX} at its ideal time lag, to the model improved R^2 by more than 1% and the p-value of T_{MAX} was less than 0.1, then T_{MAX} at its ideal time lag was added to the model. Then it was tested whether adding another weather variable, for example, P_{TOT}, would again meet these criteria. This would continue until adding an additional variable did not improve R^2 by more than 1% or if the p-value of the variable was higher than 0.1. If adding one weather variable did not meet these criteria, it was excluded from the model and another weather variable was tested. When adding one of any of the weather variables did not improve R^2 by more than 1% or the p-value was less than 0.1, the only predictor for that pixel would be dengue at a one month time lag. It is important to note that the time lag associated with each meteorological predictor varied from pixel to pixel. Table 3 shows the 16 potential predictor combinations that were assessed.

Once the ideal predictors were selected at each pixel, the optimal prediction model was developed. An individual optimal prediction model was applied to each pixel. The general form of the prediction model was a regression containing all *N* potential predictors,

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_N x_N, \tag{4}$$

where β_0 is the intercept, β_N is the regression coefficient of the variable and x_N is the value of the variable. Typically, not all predictors were used in each pixel's model. For example, if model 7 was deemed to be the optimal predictor for a pixel, the model for that pixel was:

$$\hat{y} = \beta_0 + \beta_{Dpop} D_{POP} + \beta_{Tmax} T_{MAX} + \beta_{Ptot} P_{TOT}. \tag{5}$$

An optimal prediction equation was applied to each pixel for each month of the prediction period, 2013-2015. The results of the prediction model were compared to the observed dengue incidence at each pixel for each month of 2013-2015 and model error was assessed. It is also important to note that the intercept was not always significant to the $p < 0.1$ level, and therefore not every pixel's model included the β_0 term. Once the model predictions were complete at each pixel for each of the 36 months of the prediction period, model error was assessed.

Table 2. Summary of predictors used in the dengue prediction models and their abbreviations.

Predictors	
Dengue	D_{POP}
Maximum Temperature	T_{MAX}
Mean Temperature	T_{MEAN}
Minimum Temperature	T_{MIN}
Total Precipitation	P_{TOT}

Table 3. Summary of potential predictor combinations. The combination that was deemed most effective at each pixel individually, via correlation coefficient and predictor p-value analysis, was used as the basis of the prediction model at each pixel.

Model #	Variables
0	D_{POP}
1	$D_{POP} + T_{MAX}$
2	$D_{POP} + T_{MEAN}$
3	$D_{POP} + T_{MIN}$
4	$D_{POP} + P_{TOT}$
5	$D_{POP} + T_{MAX} + T_{MEAN}$
6	$D_{POP} + T_{MAX} + T_{MIN}$
7	$D_{POP} + T_{MAX} + P_{TOT}$
8	$D_{POP} + T_{MEAN} + T_{MIN}$
9	$D_{POP} + T_{MEAN} + P_{TOT}$
10	$D_{POP} + T_{MIN} + P_{TOT}$
11	$D_{POP} + T_{MAX} + T_{MEAN} + T_{MIN}$
12	$D_{POP} + T_{MAX} + T_{MEAN} + P_{TOT}$
13	$D_{POP} + T_{MAX} + T_{MIN} + P_{TOT}$
14	$D_{POP} + T_{MEAN} + T_{MIN} + P_{TOT}$
15	$D_{POP} + T_{MAX} + T_{MEAN} + T_{MIN} + P_{TOT}$

RESULTS

Interpolation Results

The results of the inverse distance weighting, kriging and cokriging interpolations were compared. The results of the IDW interpolation are highly influenced by the weather station locations. Because IDW only takes into account the distance an estimated location is from a sample location, the areas immediately surrounding the weather stations are strongly influenced by the value of the weather variable at that station. In areas that are far from any weather stations, there may be large errors in prediction. The results of the Ordinary Kriging interpolation are similar, but less extreme. Again, the location of the weather stations strongly influences the accuracy of the interpolation. Due to the low number and clustered nature of weather stations with available data in Colombia, IDW and kriging did not provide realistic representations of temperature and precipitation trends across the country. Cokriging helped to eliminate the issue of sparse weather data availability by utilizing elevation as a secondary variable, because it is highly correlated to temperature and precipitation. Additionally, the interpolation will be most accurate near the weather stations themselves, and the weather stations tend to be in areas of high population. These are the areas of main concern for this prediction model. Very few people live in areas where there are not weather stations nearby and those areas see very few annual dengue cases, for example, in the Amazon region. The results of the different interpolation methods for T_{MAX}, T_{MEAN}, T_{MIN}, and P_{TOT} can be compared in Figures 6, 7, 8 and 9, respectively. For Figures 6-9, the IDW and kriging interpolation maps were created by importing the results from doing the interpolation in R, then constructing maps in ArcMap. The cokriging interpolation and map construction was completed entirely in ArcMap.

Table 4 shows a comparison of the errors of the interpolation methods. These were average errors for each month of all eleven years of the study period. Three measures of error were employed: mean absolute error (MAE), root mean square error (RMSE) and root mean square standardized error (RMSSE). The equations for these three error metrics are as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\widehat{x_\iota} - x_i| \tag{6}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\widehat{x_\iota} - x_i)^2} \tag{7}$$

$$RMSSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\frac{\widehat{x_\iota} - x_i}{\hat{\sigma}})^2} \tag{8}$$

where $\widehat{x_\iota}$ is the estimated value of the interpolation and x_i is the actual observed value of each weather station and σ is the standard deviation. MAE is the average absolute difference between the predicted and observed values, RMSE is the average of the squared differences, and RMSSE standardizes RMSE by taking standard deviation into account. For MAE and RMSE, a smaller value indicates lower error (Chai & Draxler 2014). The closer the value of RMSSE to 1, the better. If it is greater than 1, as is the case for all three interpolation methods, this indicates an underestimation of variability in the prediction.

RMSE assumes a normal error distribution (i.e. few large errors, with small errors more frequent), whereas MAE assumes a uniform error distribution, which would indicate the same number of large and small errors (Chai & Draxler 2014). A normal error distribution was more likely for this study and therefore RMSE was considered as the most important measure of error. Additionally, RMSE accounts for large errors, so a lower RMSE indicates fewer large errors. Cokriging had by far the lowest RMSE, so had much fewer large errors than IDW or ordinary kriging. Therefore, the cokriged weather data was used in the correlations with dengue incidence data to create a prediction model. In Figure 9, precipitation is very low in the Amazon region of Colombia, where one would assume precipitation values would be high. It is likely that in some months, precipitations errors may be large in the Amazon region, but these errors are overlooked because of the very low population and low dengue occurrence in this area.

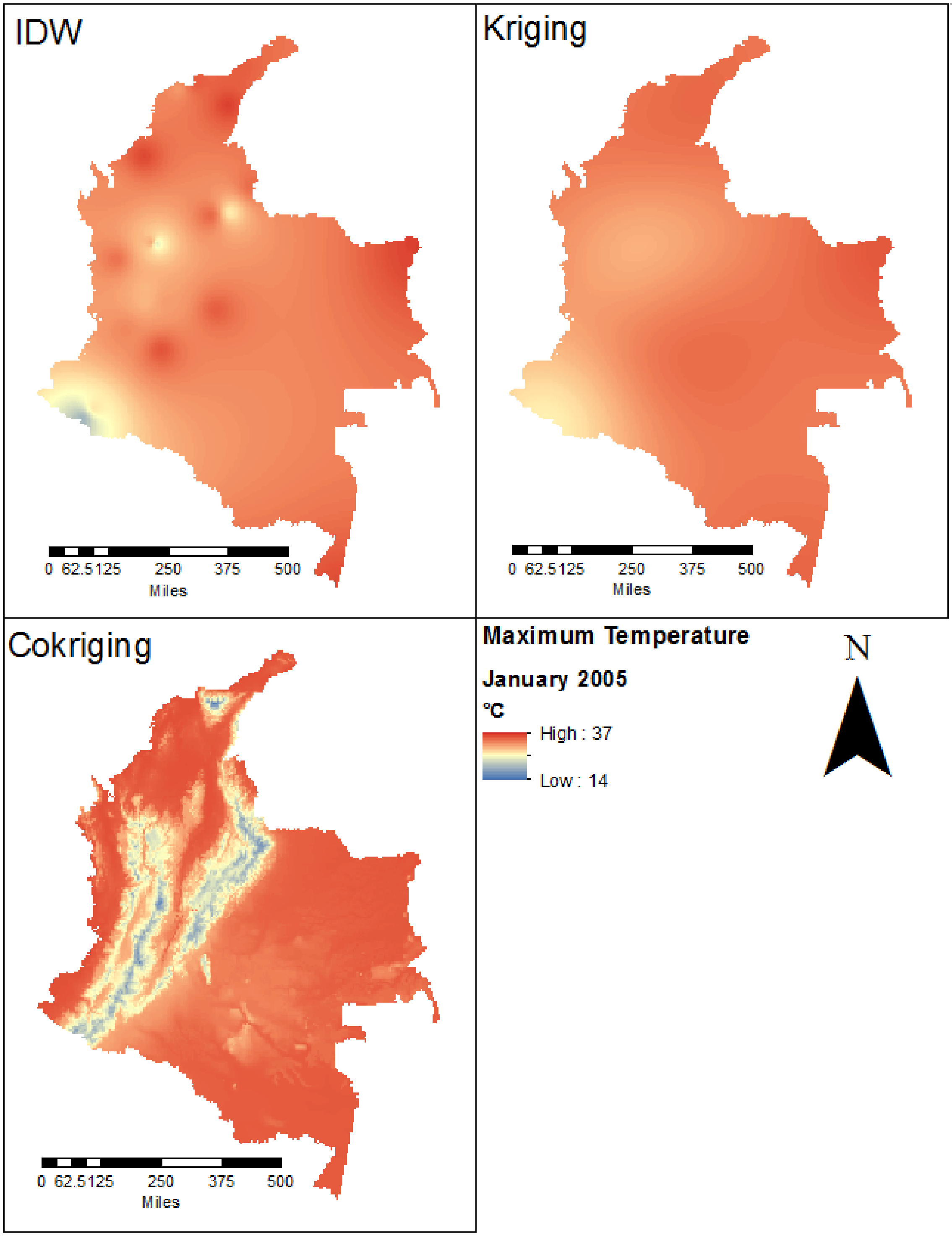


Figure 6. Results of IDW, ordinary kriging and cokriging interpolations of maximum temperature for January 2005.

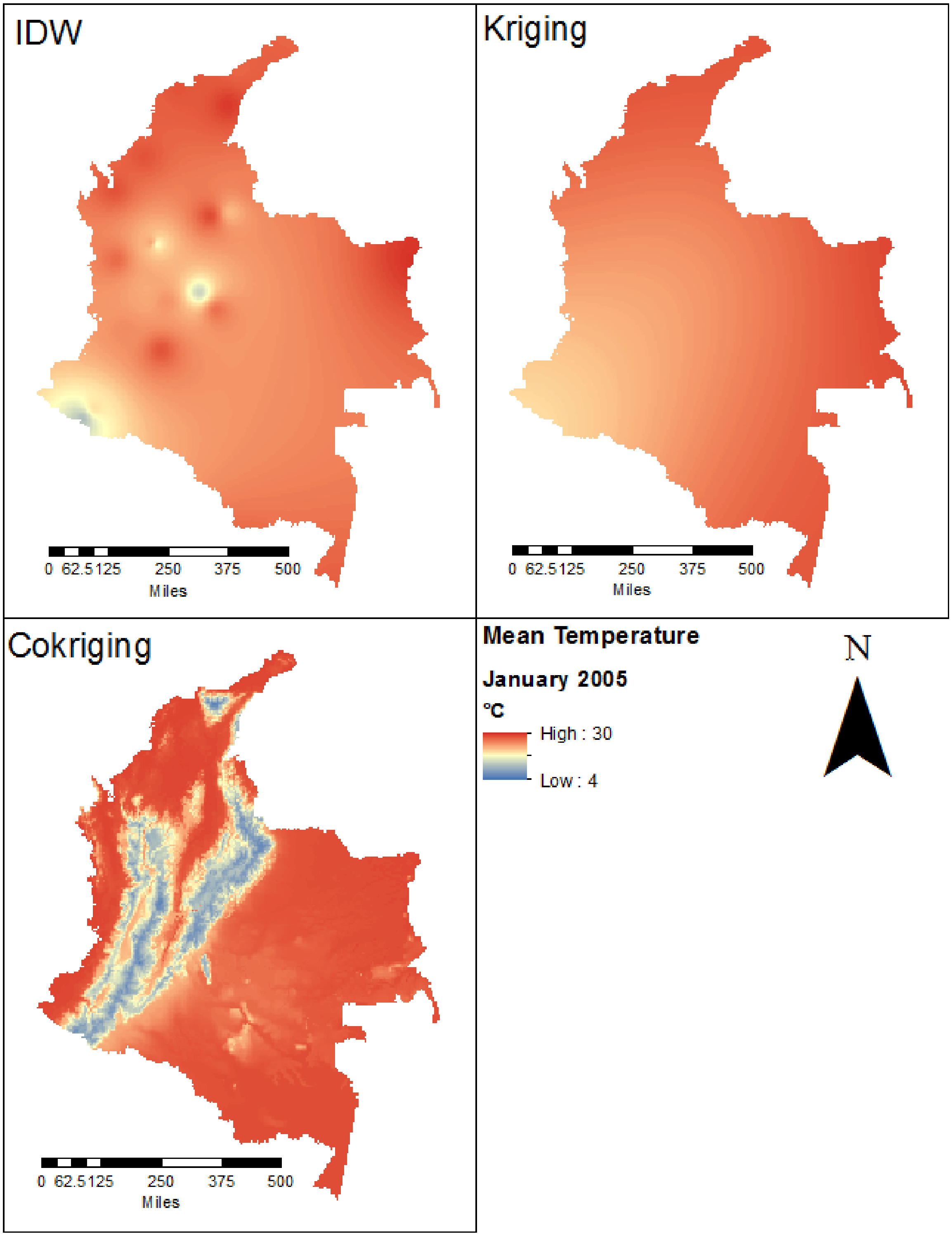


Figure 7. Results of IDW, ordinary kriging and cokriging interpolations of mean temperature for January 2005.

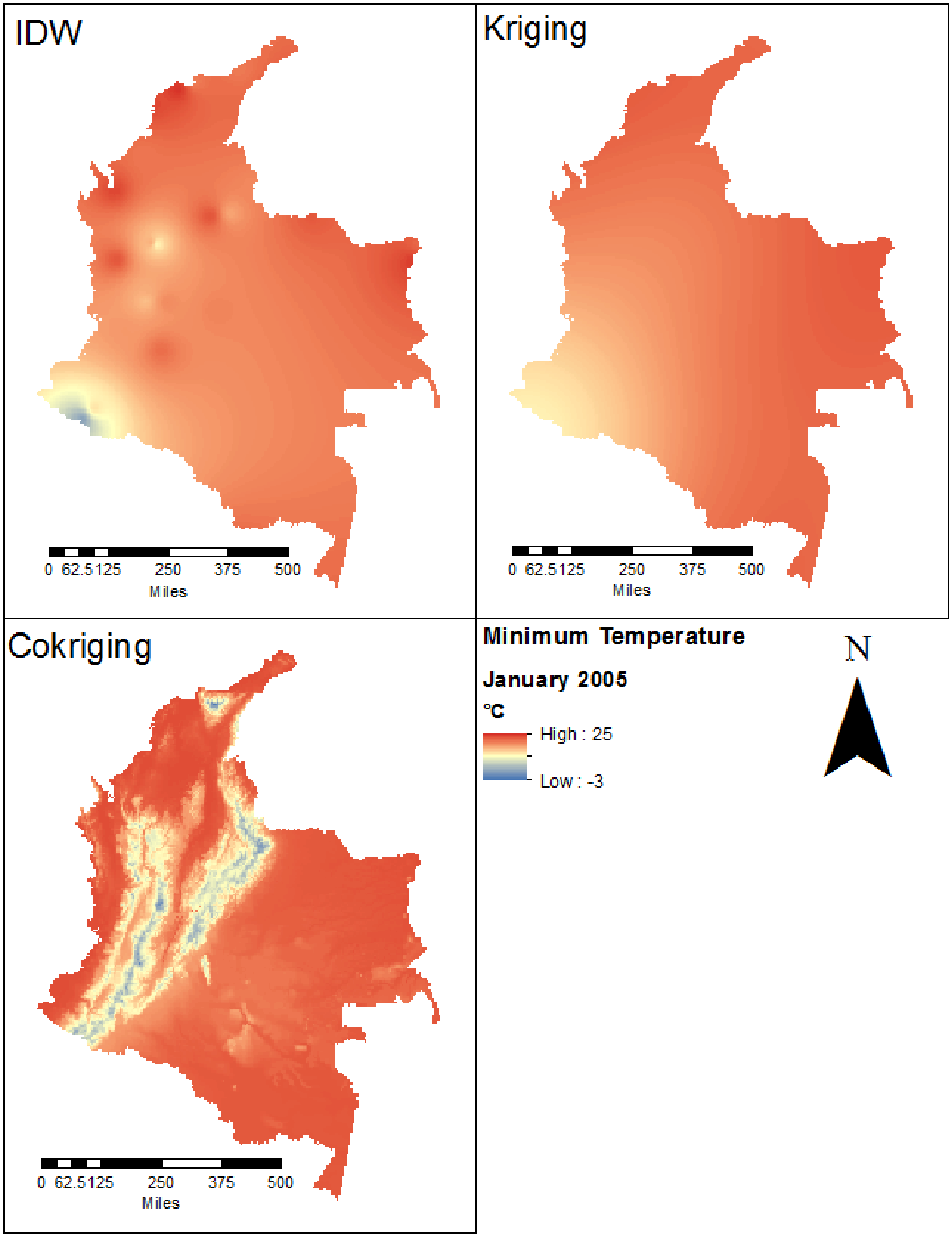


Figure 8. Results of IDW, ordinary kriging and cokriging interpolations of minimum temperature for January 2005.

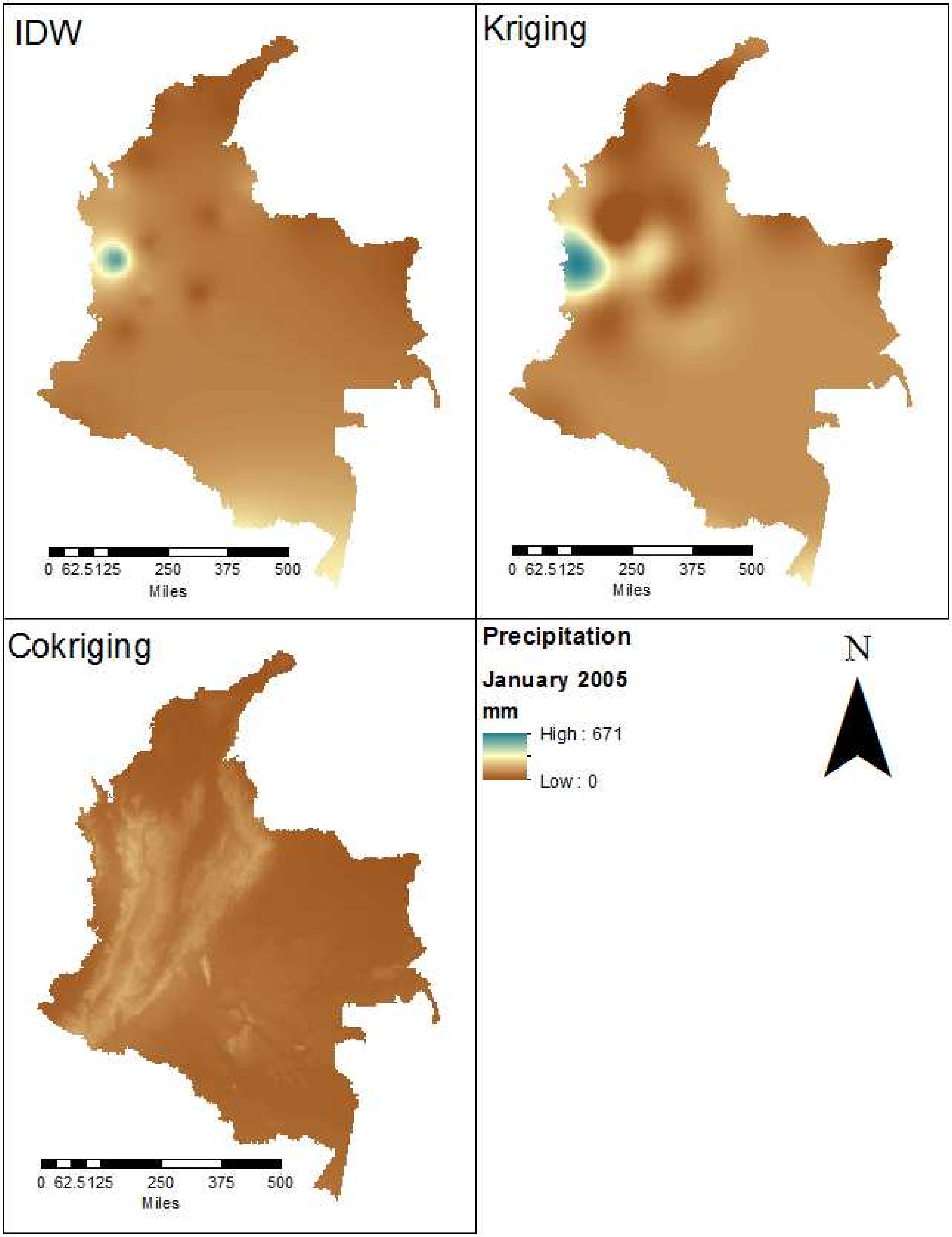


Figure 9. Results of IDW, ordinary kriging and cokriging interpolations of precipitation for January 2005.

Table 4. Error comparison for the three interpolation methods. The three measures of error assessed were mean error (ME), root mean squared error (RMSE), and root mean squared standardized error (RMSSE). Because RMSE was by far the lowest for cokriging, the cokriged weather data was used for further analysis.

Error	**IDW**			**Ordinary Kriging**			**Simple Cokriging**		
Averages	ME	RMSE	RMSSE	ME	RMSE	RMSSE	ME	RMSE	RMSSE
MaxT	0.61581	47.08864	NA	0.05548	45.16173	0.99747	-0.59642	3.61886	0.82667
MeanT	0.66744	15.74018	NA	0.05208	16.02554	1.11479	-1.01906	2.63523	5.61023
MinT	0.79762	22.36692	NA	0.06024	22.47877	1.02585	-1.68076	3.73030	3.80652
Precip	9.47522	25227.19754	NA	6.85695	28385.50790	1.55134	9.62189	169.19742	1.58470

Correlations between Weather and Dengue

Each weather variable was compared to dengue incidence at different time lags at each pixel for each month for the years 2005-2012 to serve as model training data. During model training, correlations between each weather variable and dengue incidence were identified at time lags from one to six months. These correlations were used to create the dengue prediction model. An illustration of the correlations that were identified on a pixel by pixel basis can be found in Figures 10-13. The ideal time lag for each variable at each pixel was identified as the time lag with the highest R value. Maps of the ideal time lags for each weather variable are presented in Figure 14. When comparing these figures, some patterns become apparent. For maximum temperature, correlations at high elevations (in the Andes region) were much higher at a 6-month lag than at a 1 month lag. Along the coast and in the valley, a 1 month lag between maximum temperature and dengue had a much higher R. Distinct patterns are not as apparent for mean temperature, but the majority of the country experienced positive correlations between mean temperature and dengue incidence at both 1- and 6- month lags. Correlations between minimum temperature and dengue at a 1 month lag were strongly positive at low elevations (along the coast and in the valley) and neutral at high elevations. At a 6- month lag, R was moderately to strongly negative at high elevations and neutral at low elevations. Therefore, in figure 12 it is apparent that at these high elevations, 6- month lag was used for minimum temperature. Precipitation at both a 1 and 6- month lag was negatively correlated to dengue incidence most of Colombia. Typically, stronger negative correlations were found at longer lag times.

Based on these optimal lags, the ideal predictive model was identified at each pixel via stepwise regression (see Table 3 for the list of potential models). Figure 15 shows the model that was used to predict dengue at each pixel. Model 0 is the model that incorporated only the autoregressive dengue correlation. Model 0 was used for those areas in which no dengue was previously present (in grey in Figure 15). Ten of the sixteen potential models were found to be the best predictive models for individual pixels. The models that were used are summarized in Table 5.

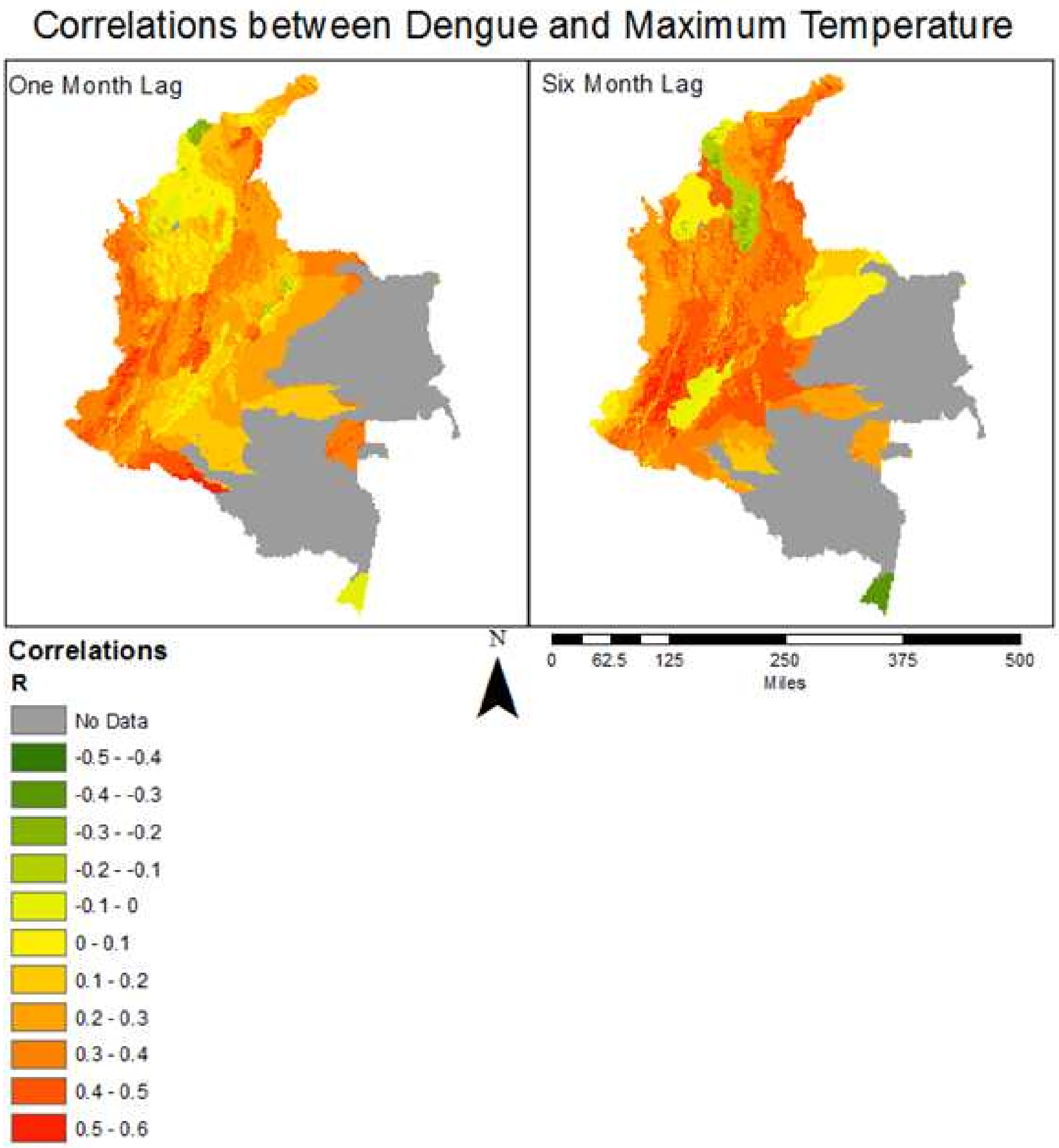


Figure 10. A comparison of correlation coefficient (R) values between dengue incidence and maximum temperature at a one-month and six-month time lag. Some pixels experienced stronger correlations with a shorter lag time, and some experience a stronger correlation with a longer lag time. Negative correlations were stronger at a longer time lag.

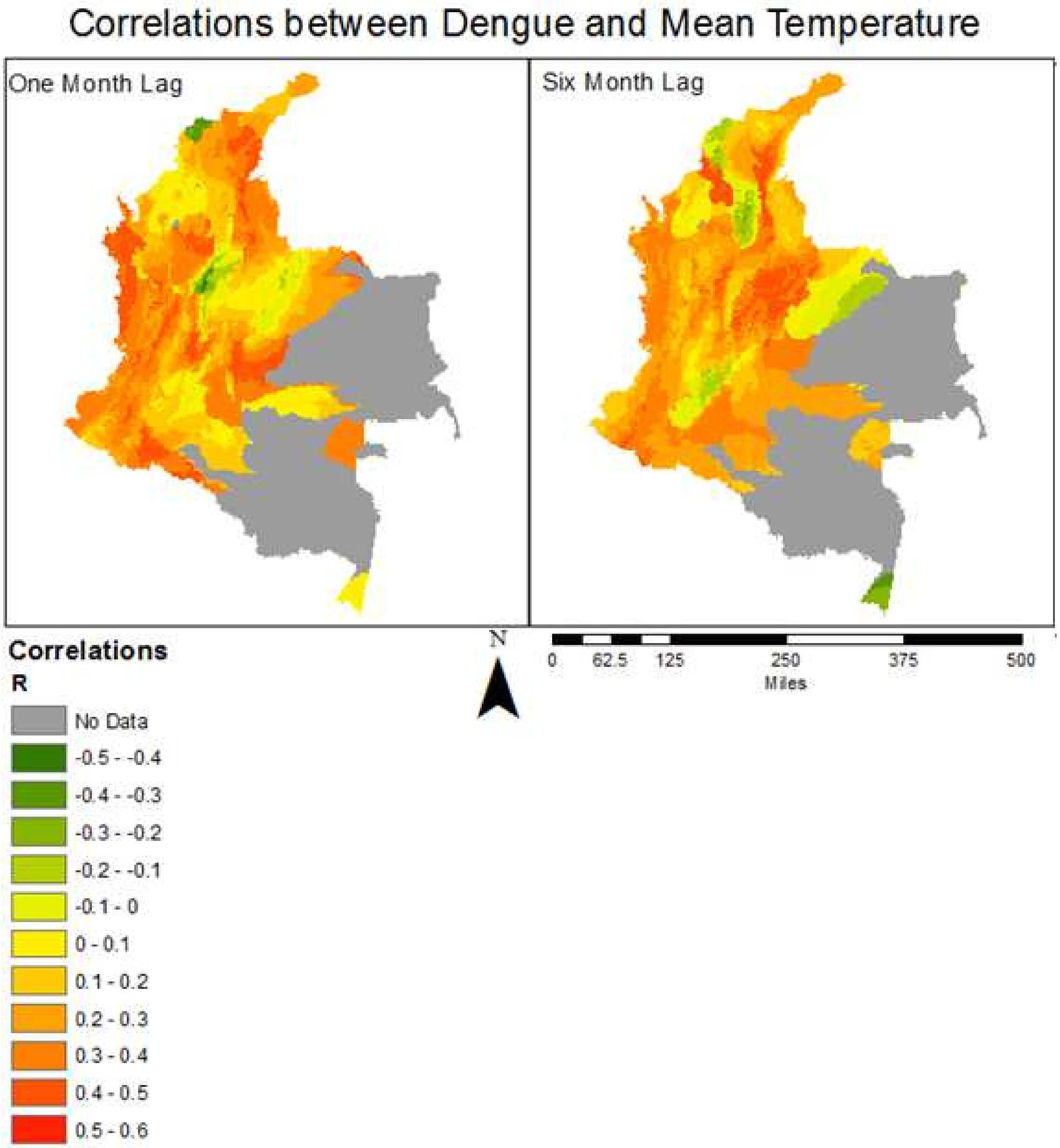
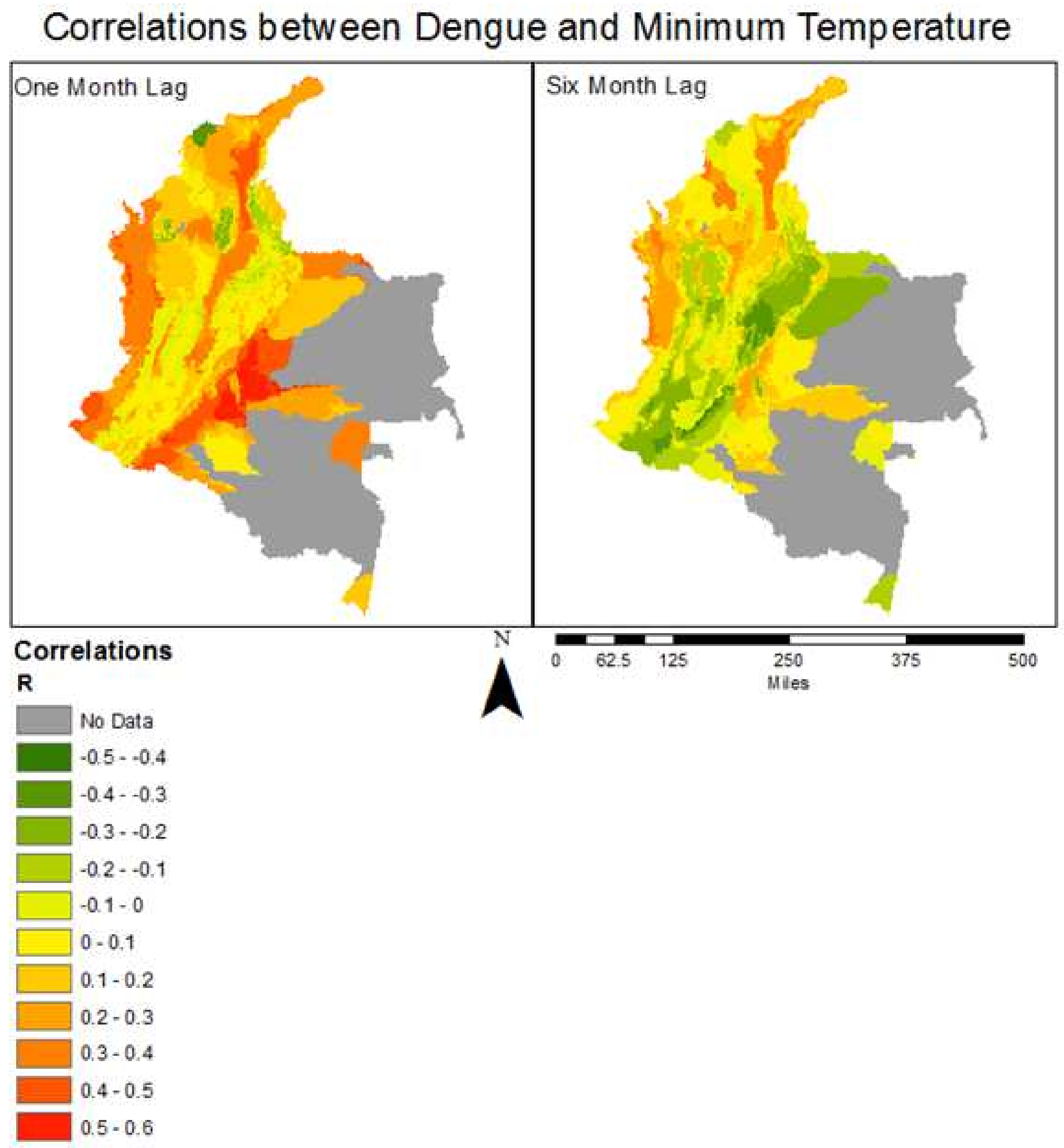


Figure 11. A comparison of correlation coefficient (R) values between dengue incidence and mean temperature at a one-month and six-month time lag. There are more pixels with strong positive correlations at a one month time lag.



Correlations between Dengue and Minimum Temperature

One Month Lag

Six Month Lag

Figure 12. A comparison of correlation coefficient (R) values between dengue incidence and minimum temperature at a one-month and six-month time lag. At a shorter time lag, strong positive correlations are found at low elevations and weak correlations exist at high elevations. At a longer time lag, strong negative correlations exist at high elevations.

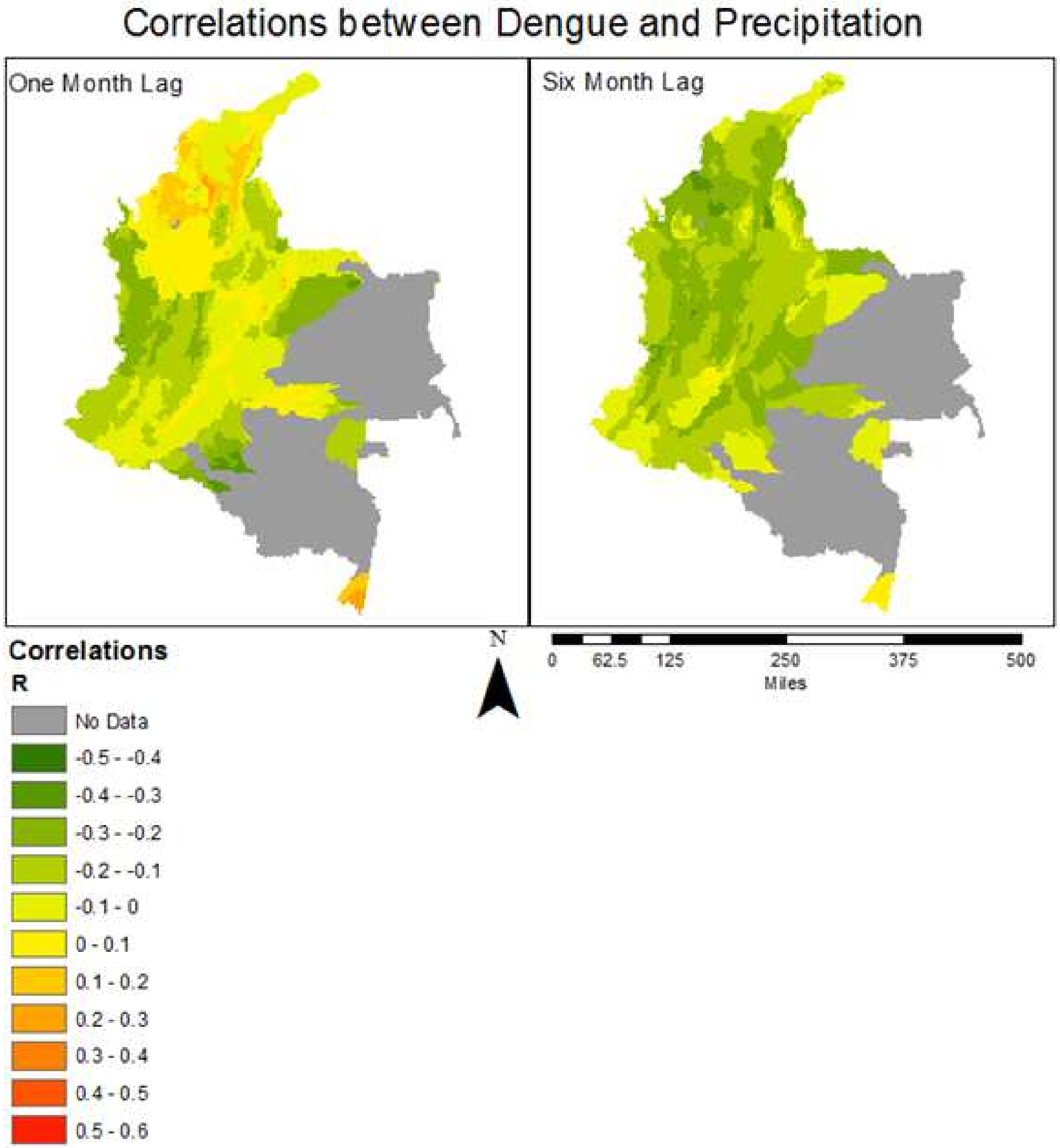


Figure 13. A comparison of correlation coefficient (R) values between dengue incidence and precipitation at a one-month and six-month time lag. At both time lags, the country is dominated by negative correlations.

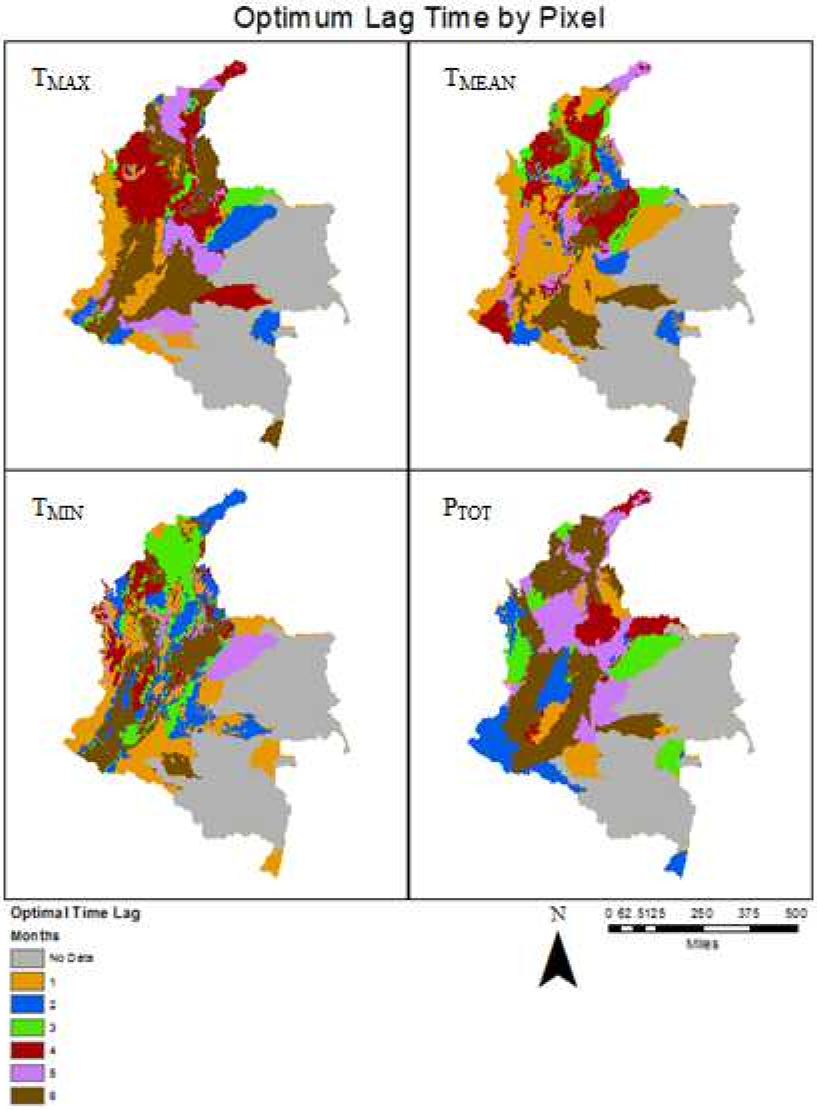


Figure 14. Maps of the optimal lag time for each predictor at each pixel. The optimal lag was the lag time with the highest R value. The grey areas represent areas where no dengue occurred. The optimal time lag for these areas for each predictor was one month.

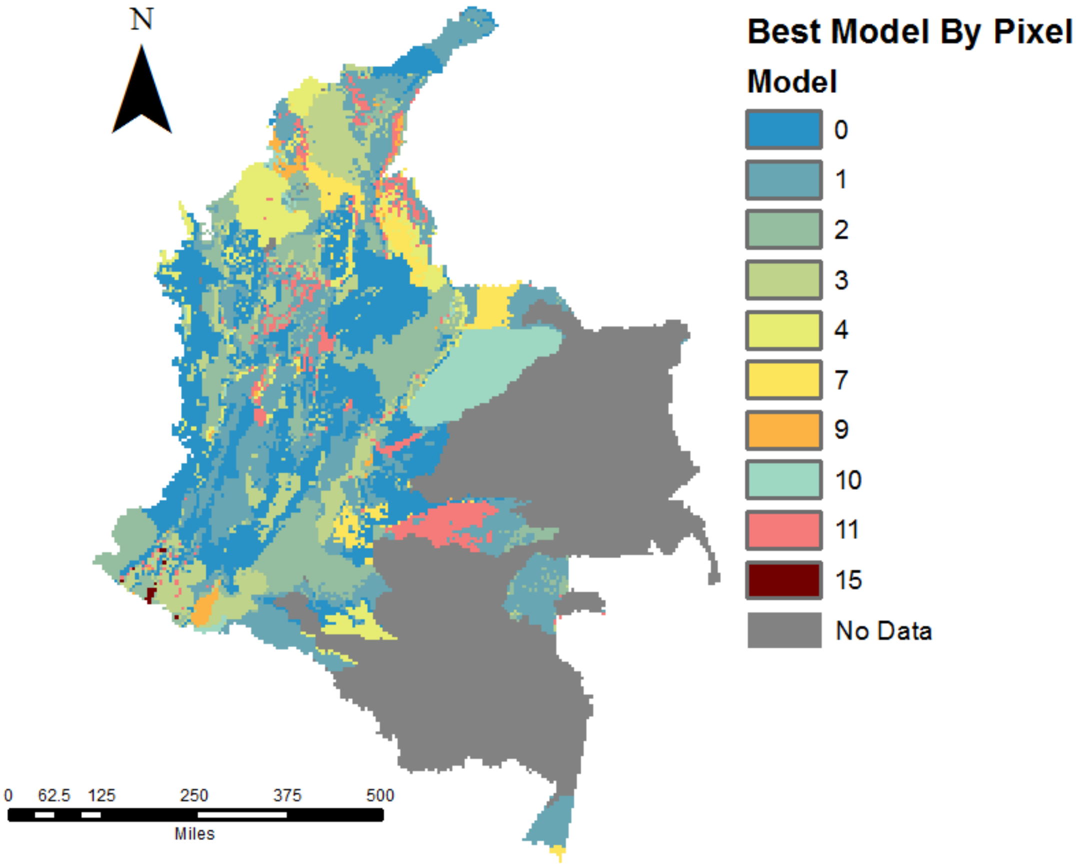


Figure 15. Map of the ideal model at each pixel as determined by stepwise regression. The grey areas represent the areas where no dengue cases were reported during the model training years. For these areas, Model 0 was always used.

Table 5. Prediction models that were used and the variables associated with each model.

Model #	Variables
0	D_{POP}
1	$D_{POP} + T_{MAX}$
2	$D_{POP} + T_{MEAN}$
3	$D_{POP} + T_{MIN}$
4	$D_{POP} + P_{TOT}$
7	$D_{POP} + T_{MAX} + P_{TOT}$
9	$D_{POP} + T_{MEAN} + P_{TOT}$
10	$D_{POP} + T_{MIN} + P_{TOT}$
11	$D_{POP} + T_{MAX} + T_{MEAN} + T_{MIN}$
15	$D_{POP} + T_{MAX} + T_{MEAN} + T_{MIN} + P_{TOT}$

Model Prediction Results

Once correlations between each weather variable and dengue incidence were identified from the model training data, dengue was predicted at each pixel for each month of 2013-2015. Figures 16 and 17, respectively show maps of predicted versus actual dengue amounts by pixel for May 2013 and September 2015 as an illustration of the model output. These two months were chosen because May 2013 had the lowest mean absolute percent error (MAPE) for the prediction period, and September 2015 had the highest. Pixel size was small for this analysis (6.6 km x 6.6 km), so the number of dengue cases per pixel was very low, typically less than 0.01 cases per pixel per month. Figure 18 shows the total number of predicted versus actual cases by month for the entire country, and Figure 19 shows the monthly percent error.

The model consistently over-predicted the total number of dengue cases for the entire country. As shown in figures 16 and 17 in reality, high dengue incidence only occurred in a few confined areas, specifically, the cities of Medellin, Pereira, Cali, Bucaramanga and Barranquilla. The model predicted more widespread dengue incidence. In examining Figure 18, the prediction model consistently overpredicts total monthly dengue incidence by about 100 cases. With 50-150 cases being typical monthly country-wide totals, this lead to high percent errors each month.

Error by model was also assessed to understand which prediction equations, and therefore, which combinations of variables, were accurate predictors of dengue for Colombia. These results are summarized in Table 6. Model 0, the model that incorporated only the previous month’s dengue total, had the lowest percent error. Model 15, which incorporated all the potential predictors, had the second lowest error, but was only used by 17 pixels. Model 7, which incorporated a combination of dengue, maximum temperature and precipitation, had the highest percent error.

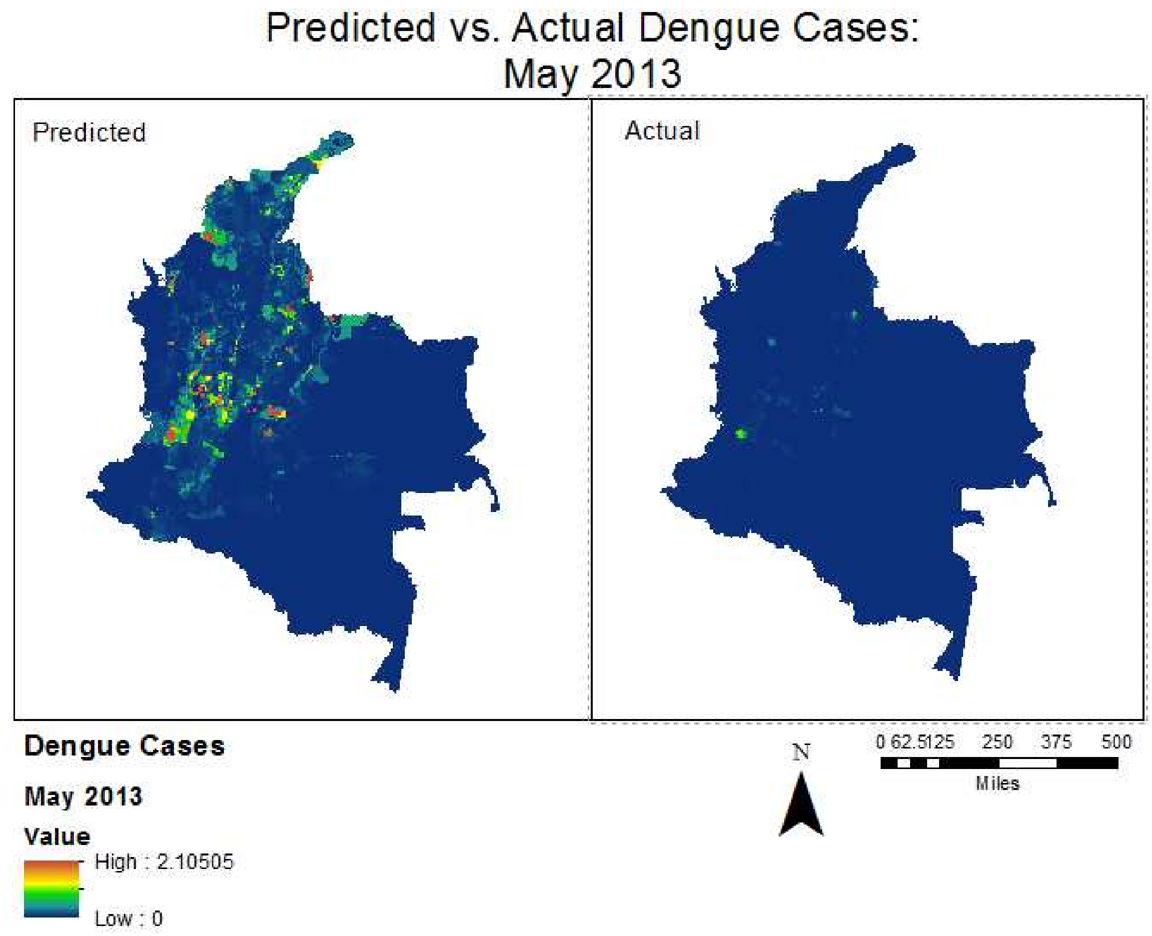


Figure 16. The number of predicted vs. actual dengue cases per pixel in May 2013. This was the month with the lowest overall percent error of the prediction period.

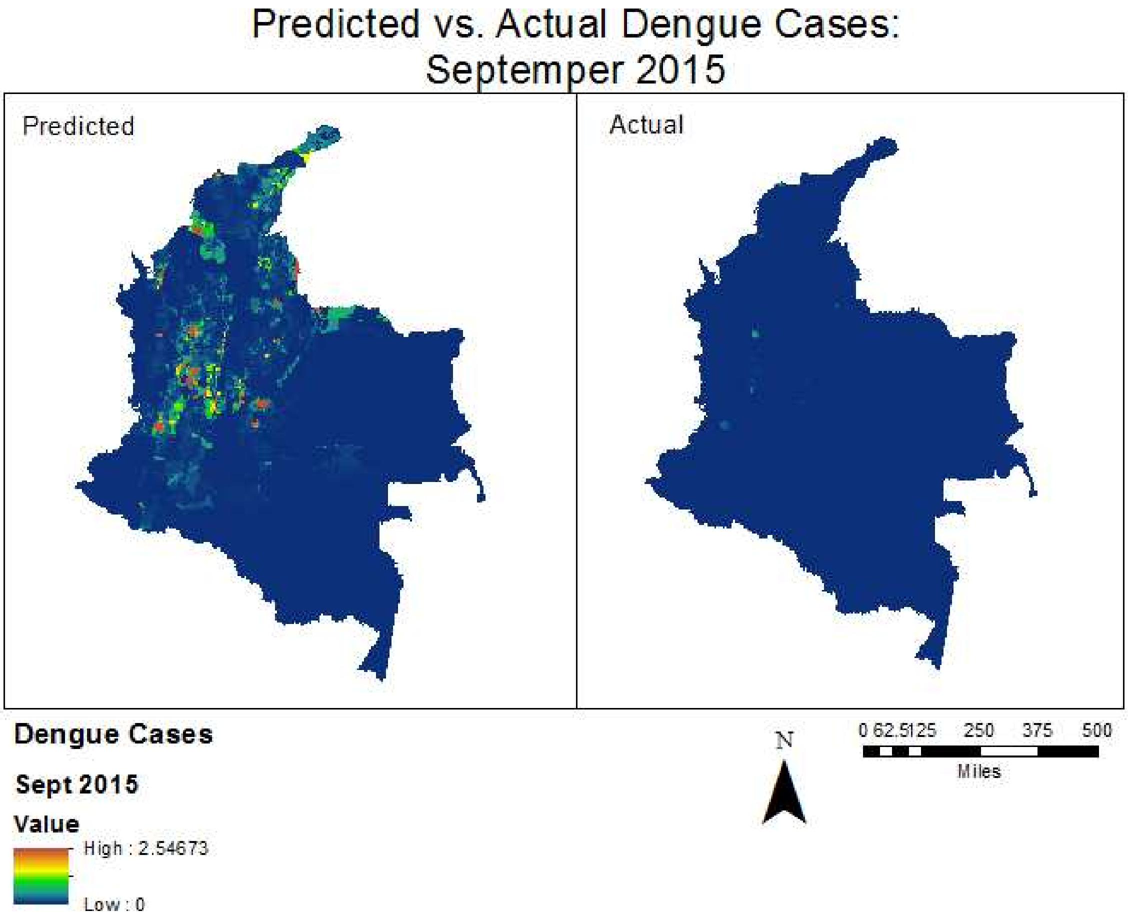


Figure 17. The number of predicted vs. actual dengue cases per pixel in September 2015. This was the month with the highest overall percent error of the prediction period.

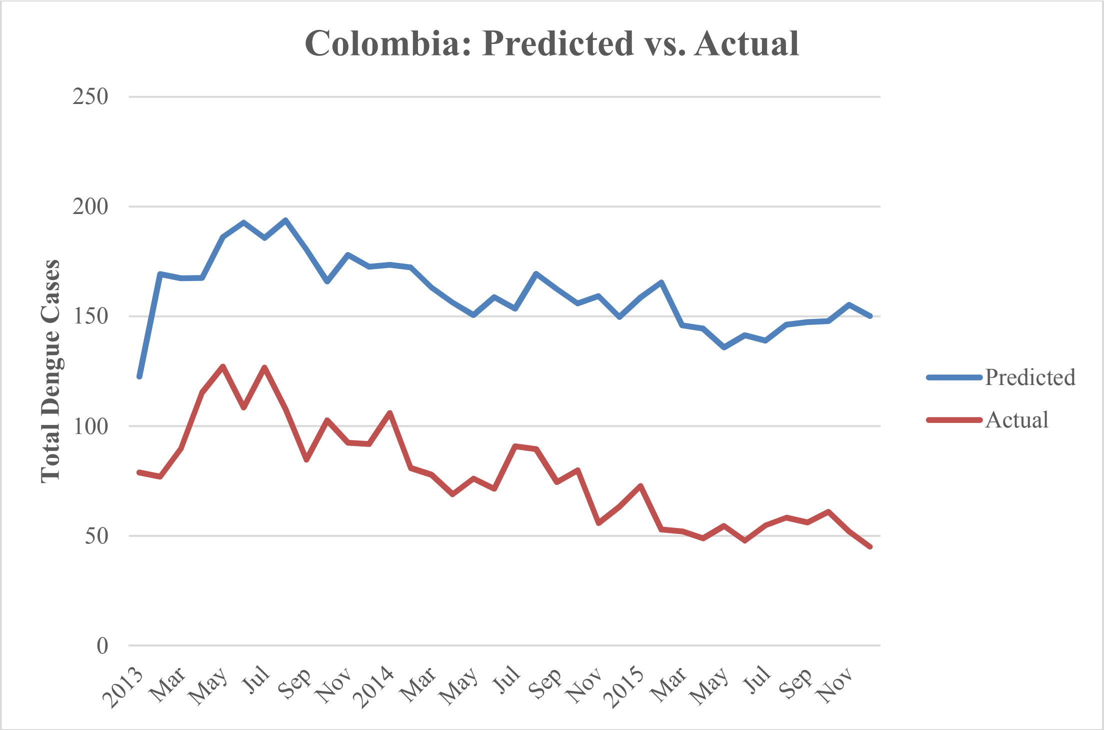


Figure 18. The total number of predicted vs. actual dengue cases by month for the entire country of Colombia. The model consistently overpredicted the number of cases.

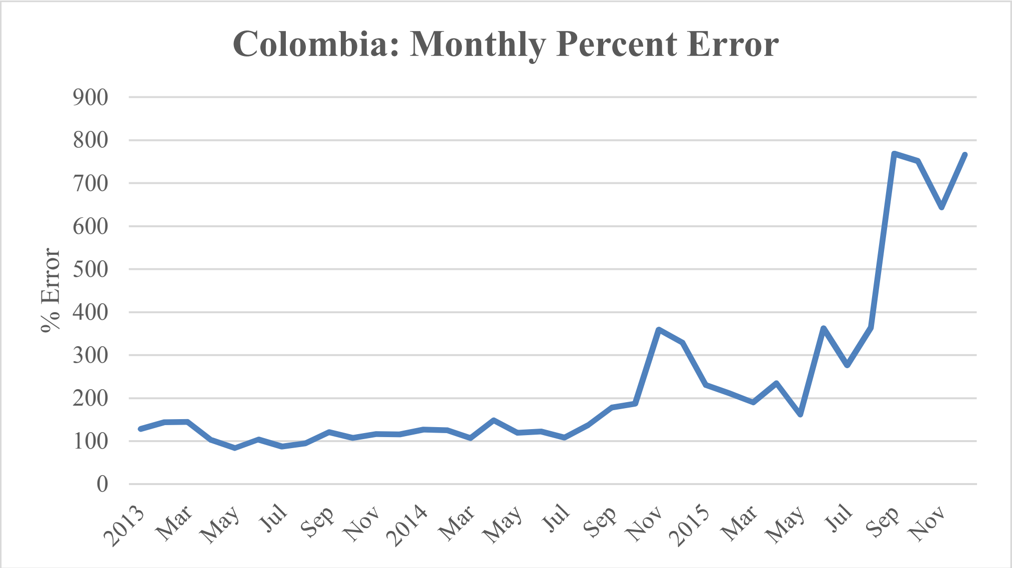


Figure 19. The percent error of the dengue prediction model for the entire country. Percent error increased drastically at the end of the prediction period.

Table 6. The errors associated with each model. Model 0, the model incorporating only the autoregressive dengue term, had the lowest average percent error. Model 7 had the highest percent error.

Model	MAE	MAPE	RMSE	# of pixels
0	0.000431	8.232241	0.000597	14,028
1	0.013232	589.5222	0.013459	4126
2	0.006543	402.1369	0.006752	2765
3	0.004361	340.8134	0.004657	1763
4	0.005962	88.10499	0.007182	1029
7	0.008126	1752.991	0.008295	645
9	0.025529	642.566	0.025988	182
10	0.002697	358.1733	0.003656	960
11	0.014338	211.321	0.01497	845
15	0.000299	62.7444	0.000342	17

Because the pixel size was very small, pixels were aggregated for the ten most populous cities in Colombia to further examine the results and success of the model in these highly-populated areas. It is important to assess model performance in this way because most of the population lives in these areas and *Ae. aegypti* is an urban mosquito, so these are the areas with a high risk of dengue. The pixels that fell within the municipal boundaries of these ten cities were aggregated to get an idea of how many dengue cases occurred in each city each month. Table 7 shows the cities that were analyzed and their populations and Figure 20 shows their locations. Table 8 shows the prediction model(s) that were used for the pixels that made up each city. Table 9 and Figures 21-40 summarize the model results and errors in each of the ten cities. The graphs in Figures 21-40 show the number of predicted dengue cases vs. actual dengue cases in each city per month, and the percent error of the model for each city per month.

Like the entire country, at the city level, the prediction model tended to overpredict the number of dengue cases that occurred. The main exceptions were Cartagena, Barranquilla and Santa Marta, the three cities along the northwest coast. The model typically underpredicted dengue incidence in most months for these three cities. Predicted and actual cases were the most similar for Bucaramanga and that city typically experienced the lowest percent errors. Dengue incidence was highly overpredicted in Pereira, leading to the highest monthly percent errors of all the cities. The monthly percent error of the prediction model for each of the cities can be compared in Figures 41 and 42. In Figure 42, the percent error graph for Pereira is removed so differences in monthly percent error for the other cities is seen more clearly. Percent errors tend to be lower at the beginning of the prediction period (2013) than at the end (2015).

A summary of the model output for the ten cities is shown in Table 10. The elevation and model associated with each city is provided. The average number of monthly predicted and observed dengue cases is provided. The correlation between predicted and observed cases was calculated, which indicates the accuracy of temporal variability. Monthly average of mean bias was calculated to indicate the accuracy of overall model magnitude and show whether the model provided an over- or under-prediction for that city. The correlation between predicted and observed cases was generally strongly positive, except for Bogota and Pereira which had very weak positive correlations. Pereira had the highest overall error, so it makes sense that the correlation between observations and predictions were low. Mean bias was low and positive for most cities. However, Medellin and Pereira had high mean bias. Medellin's average prediction was much higher than what was observed, as was Pereira's. As expected, Cartagena and Barranquilla both had a negative mean bias, indicating under-prediction. (Santa Marta's was very low and positive). Bucaramanga, the city with the least overall error, also had a slightly negative mean bias.

It is important to know how well the model performs when dengue incidence is high because that is when an accurate prediction model would be crucial. This can be assessed at both the country-wide and city scales. At both scales, dengue incidence was higher at the beginning of the prediction period (2013) than the end (2015), however large fluctuations occurred from month to month. Correlations between the number of monthly dengue cases and the monthly error of the model were identified for the entire country and for each of the cities. These results are summarized in Table 11. The consistent trend at both scales is that the number of dengue cases is negatively correlated with both monthly absolute error and monthly percent error. This means the higher the dengue incidence, the lower the error of the model.

Table 7. Ten most populous cities in Colombia as of 2017, used to analyze model performance in areas of high population.

City	Population (2017)	Longitude	Latitude
Bogota	7,674,366	-74.07194	4.710833
Cali	2,392,877	-76.53194	3.451667
Medellin	1,999,979	-75.58111	6.244167
Barranquilla	1,380,425	-74.80694	11.00389
Cartagena	952,024	-75.47917	10.39083
Cucuta	721,398	-72.49667	7.888889
Bucaramanga	571,820	-73.37417	7.119167
Pereira	440,118	-75.69056	4.808611
Santa Marta	431,781	-74.21083	11.24028
Ibague	421,685	-75.24222	4.444444

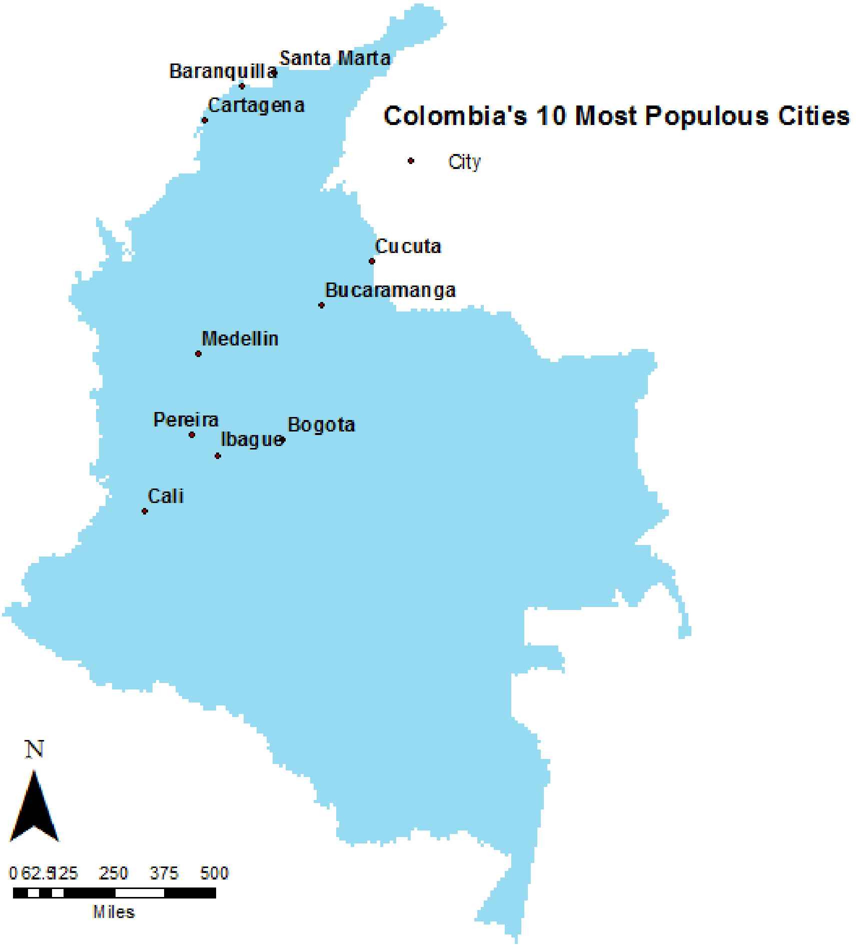


Figure 20. Geographic locations of the 10 most populous cities in Colombia, which were used to analyze model performance in highly populated areas.

Table 8. The prediction model(s) associated with each city.

City	**Models**
Bogota	1, 0; urban center: 11, 9
Cali	0,1
Medellin	11,2
Barranquilla	4
Cartagena	1, 0
Cucuta	1, 7, 4, 11
Bucaramanga	0
Pereira	1, 11, 4
Santa Marta	1,3,11
Ibague	0, 2

	2013			2014			2015			Overall		
City	**Predicted**	**Actual**	**Difference**	**Predicted**	**Actual**	**Difference**	**Predicted**	**Actual**	**Difference**	**RMSE**	**MAE**	**MAPE**
Bogota	16.93189086	8.69050058	8.241390282	15.99886012	8.88593813	7.112921993	17.35888907	4.91564449	12.44324458	4.642414403	0.773735734	159.3953339
Cali	108.0414057	89.05846	18.98294573	75.09581123	43.51731	31.57850123	73.28646064	42.32598	30.96048064	14.32196119	2.386993531	62.20273847
Medellin	134.6958092	35.381	99.31480924	146.43241	43.5306	102.90181	146.8385172	37.988	108.8505172	51.84452274	8.640753789	314.5460168
Baranquila	26.55340333	56.47363	-29.92022667	11.10840079	17.703424	-6.595023208	7.759564327	15.780181	-8.020616673	7.448069884	1.241344981	45.10765225
Cartagena	8.477156275	14.62151	-6.144353725	1.885434246	4.14387	-2.258435754	0.956034536	3.031895	-2.075860464	1.822487564	0.303747927	56.97742128
Cucuta	99.96038038	28.46723	71.49315038	99.7637262	28.9212	70.8425262	91.37041939	14.312597	77.05782239	36.56558316	6.094263861	350.3221604
Bucaramanga	23.80154086	27.0721	-3.270559138	21.19797167	21.7037	-0.505728333	10.16802735	10.09534	0.072687347	1.810491652	0.301748609	19.91897799
Pereira	56.36784361	3.405884	52.96195961	56.7926536	5.71761	51.0750436	60.93983774	6.888413	54.05142474	26.34807132	4.391345221	1194.244014
Santa Marta	11.37793999	12.82775	-1.449810015	5.497608785	2.349772	3.147836785	5.990418871	4.092466	1.897952871	1.289893705	0.214982284	102.7885739
Ibague	31.46223369	19.46448	11.99775369	25.72666099	11.69566	14.03100099	26.53079941	11.2887	15.24209941	6.889605979	1.148267663	120.7563805

Table 10. Model results by city summary. Geographic information about each city is provided, along with the mean monthly observed and predicted dengue incidence, the correlation between them and mean bias.

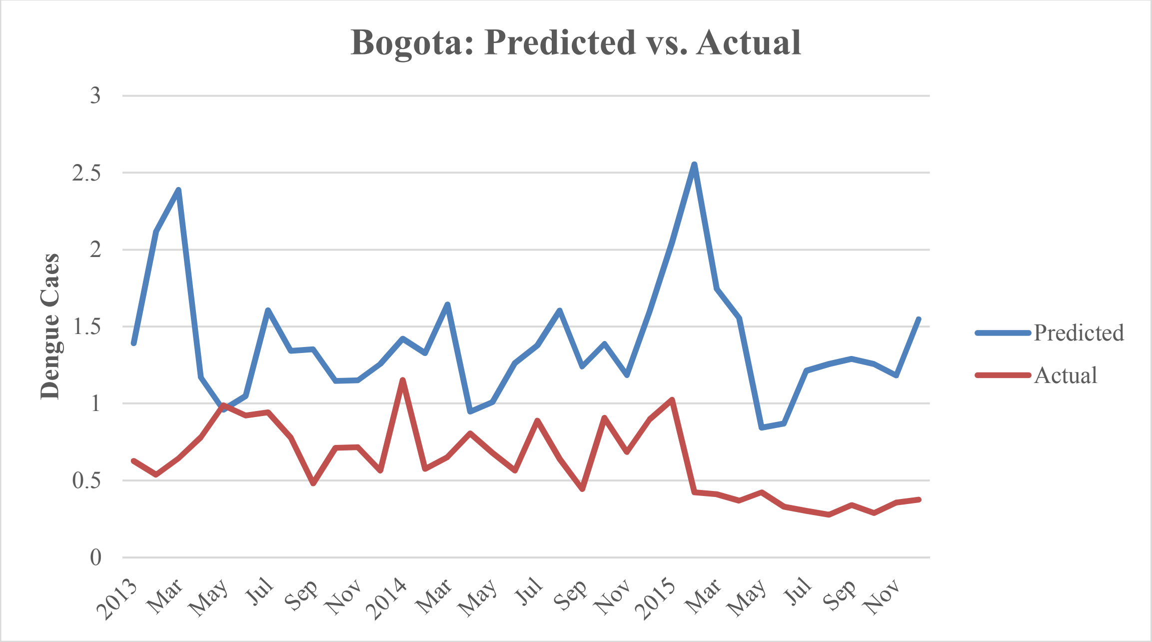


Figure 21. Graph of the number of predicted and actual dengue cases each month of the model prediction period for Bogota.

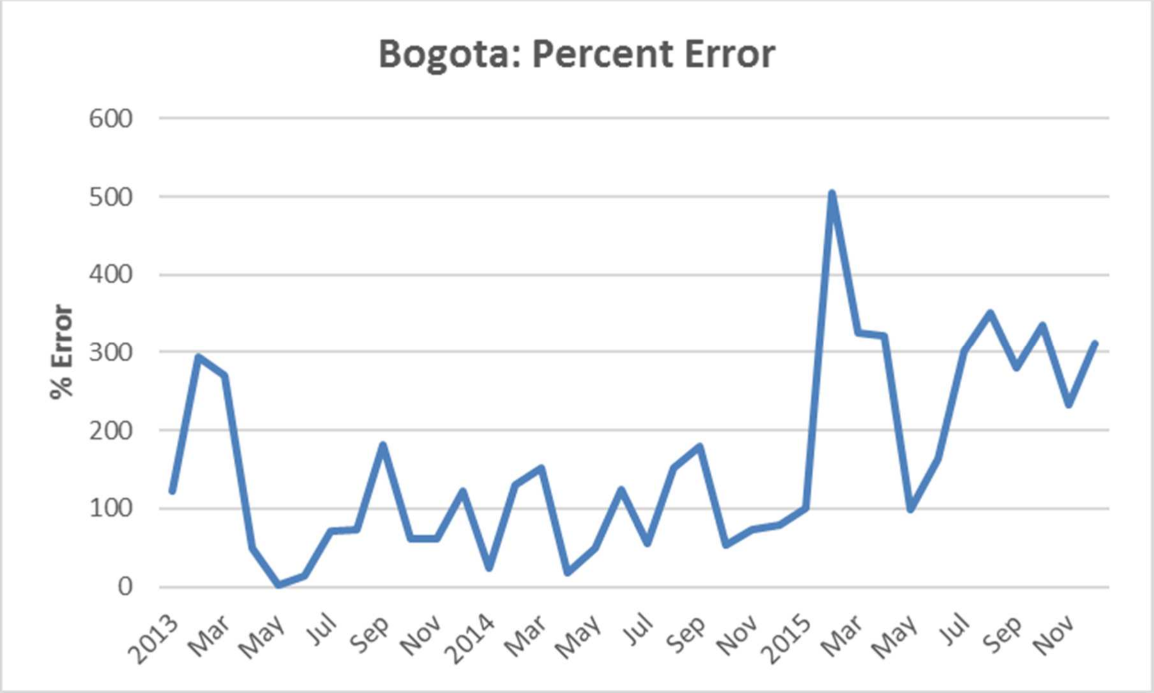


Figure 22. Graph of the percent error of the prediction model for each month of the model prediction period for Bogota.

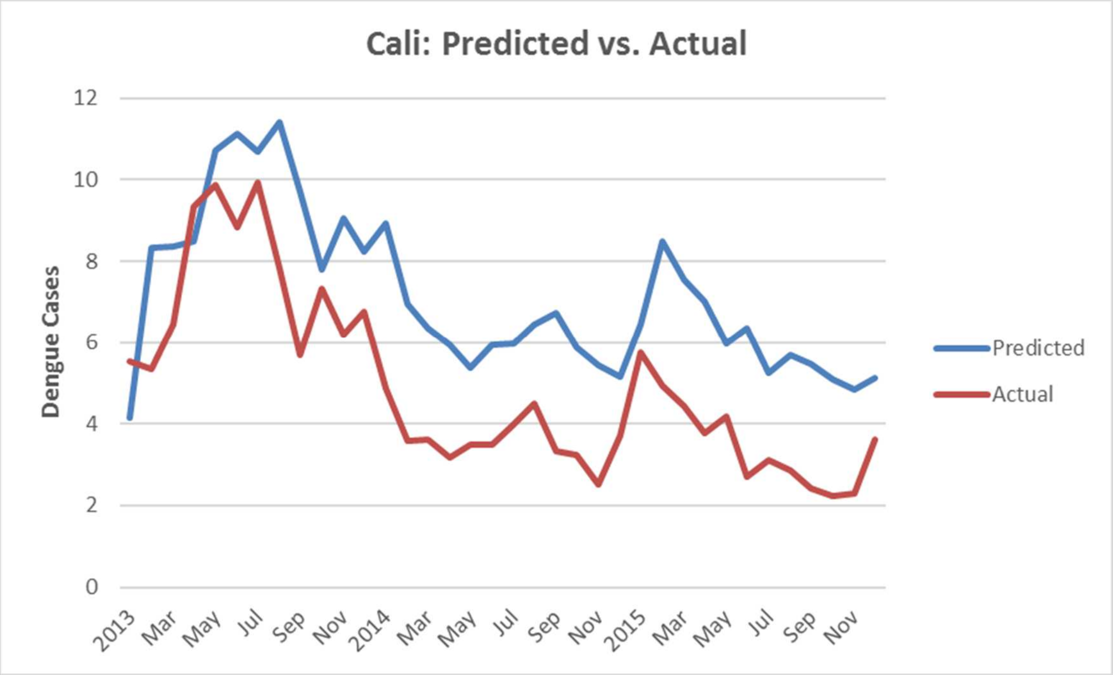


Figure 23. Graph of the number of predicted and actual dengue cases each month of the model prediction period for Cali.

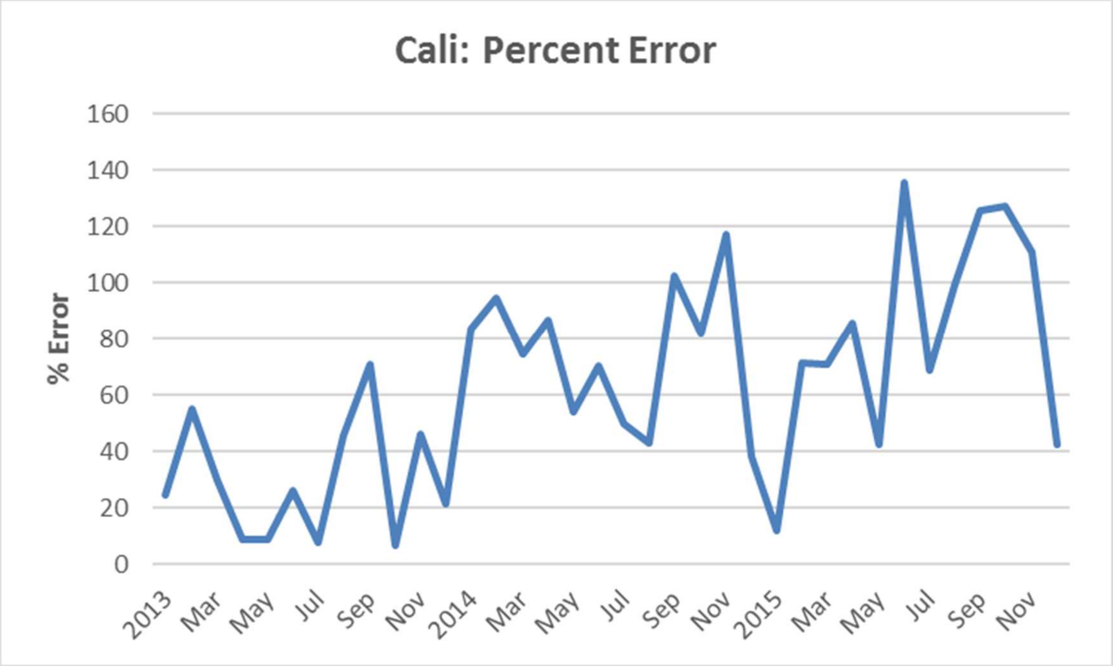


Figure 24. Graph of the percent error of the prediction model for each month of the model prediction period for Cali.

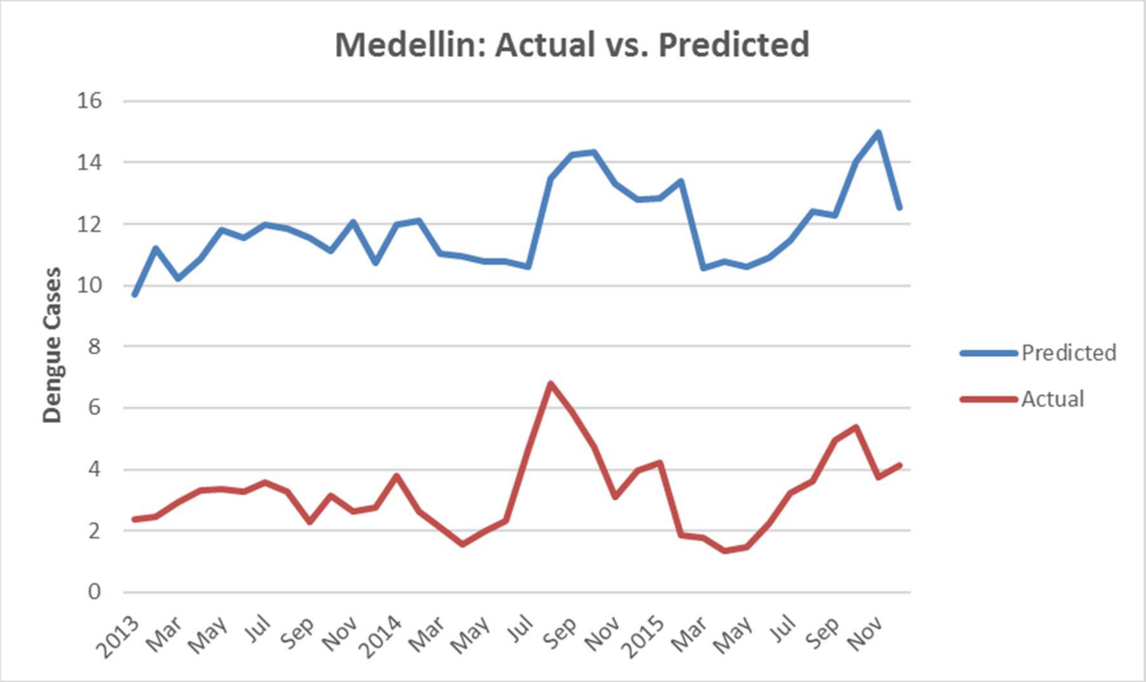


Figure 25. Graph of the number of predicted and actual dengue cases each month of the model prediction period for Medellin.

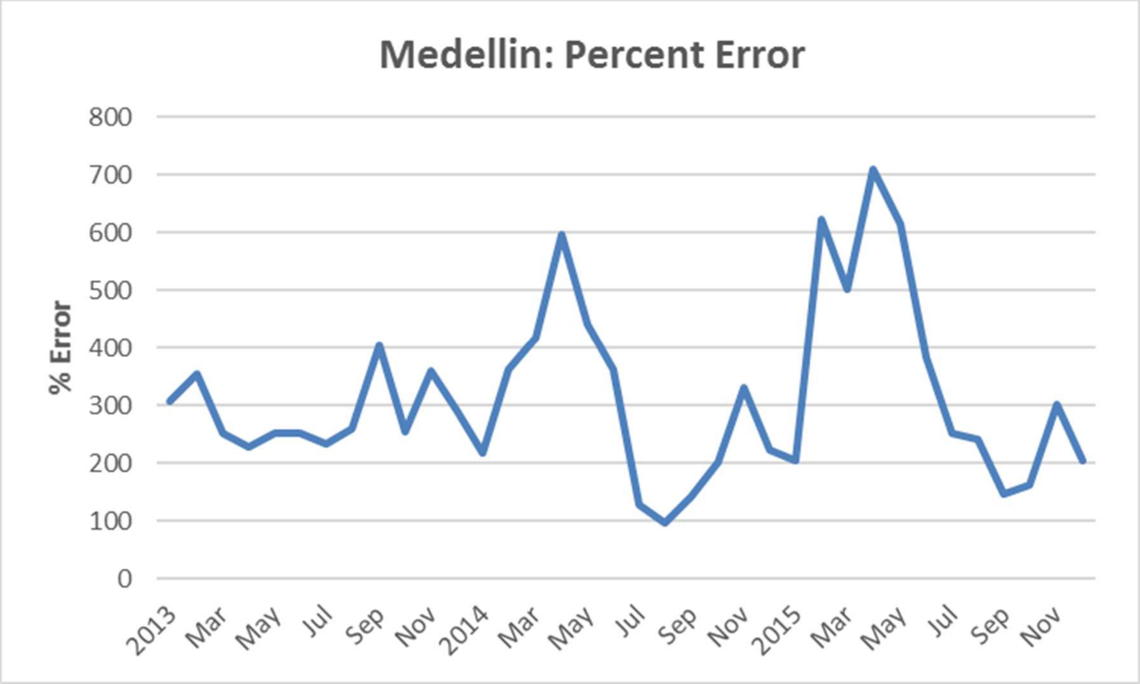


Figure 26. Graph of the percent error of the prediction model for each month of the model prediction period for Medellin.

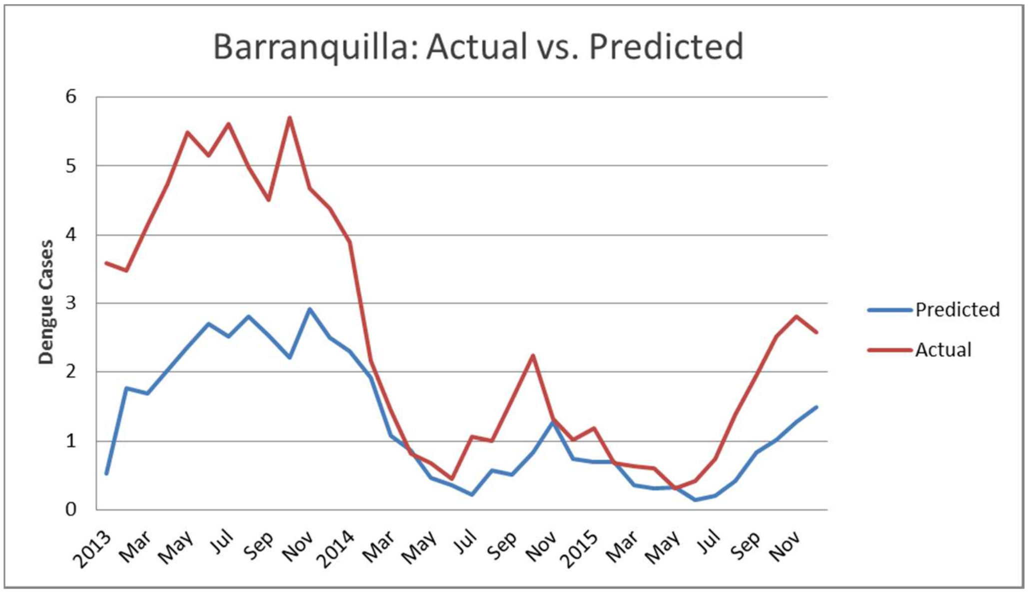


Figure 27. Graph the number of predicted and actual dengue cases each month of the model prediction period for Barranquilla.

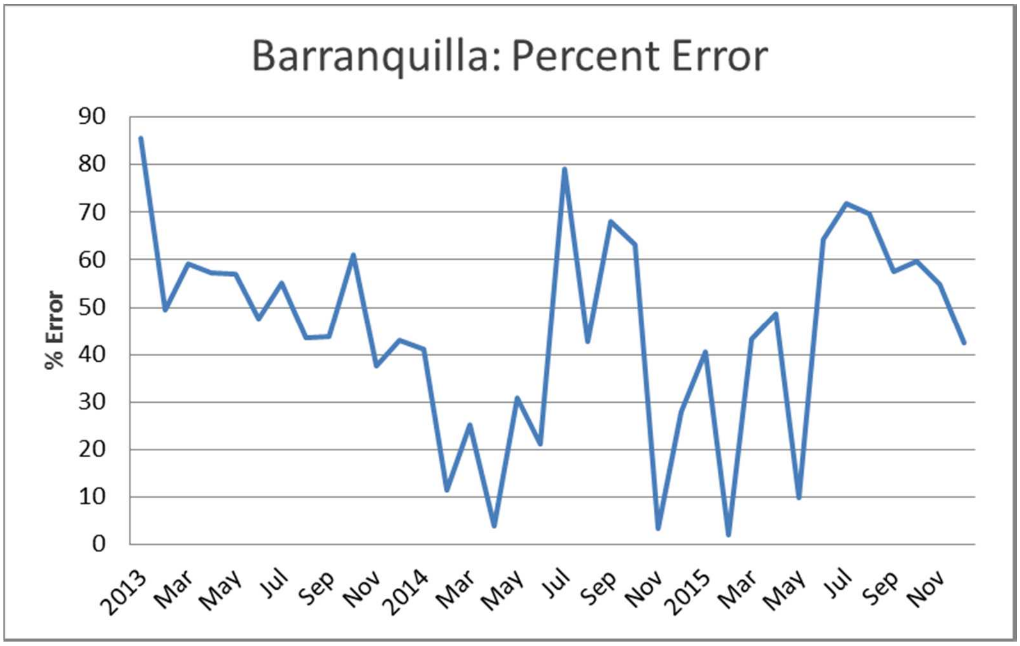


Figure 28. Graph of the percent error of the prediction model for each month of the model prediction period for Barranquilla.

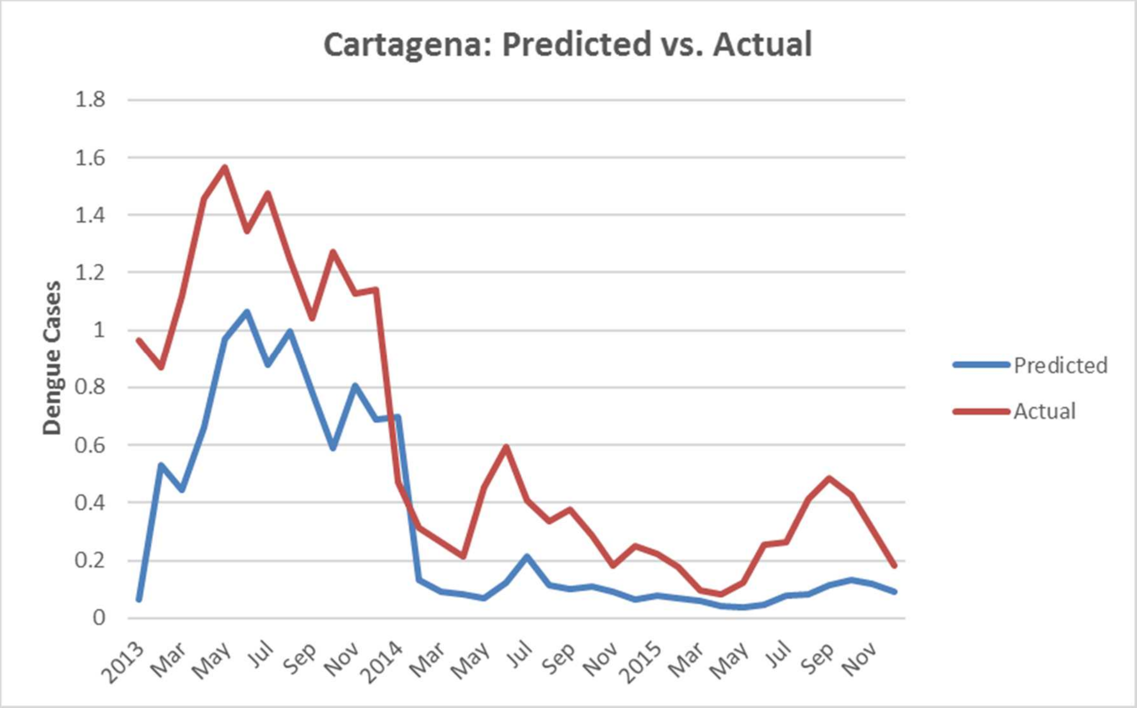


Figure 29. Graph of the number of predicted and actual dengue cases each month of the model prediction period for Cartagena.

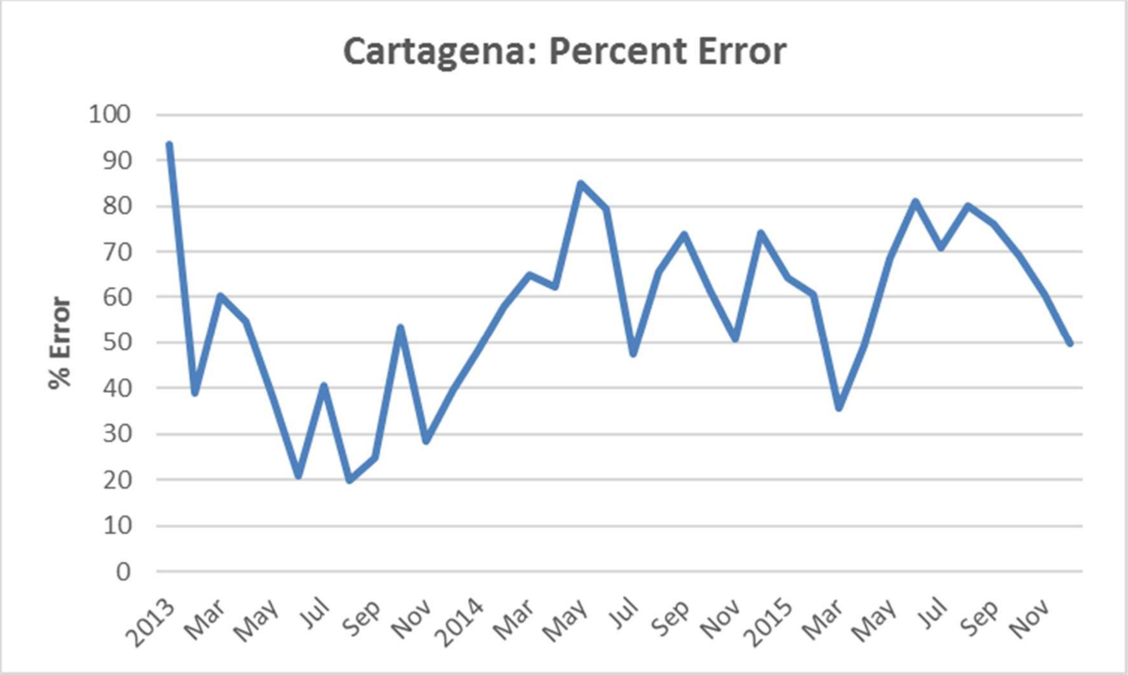


Figure 30. Graph of the percent error of the prediction model for each month of the model prediction period for Cartagena.

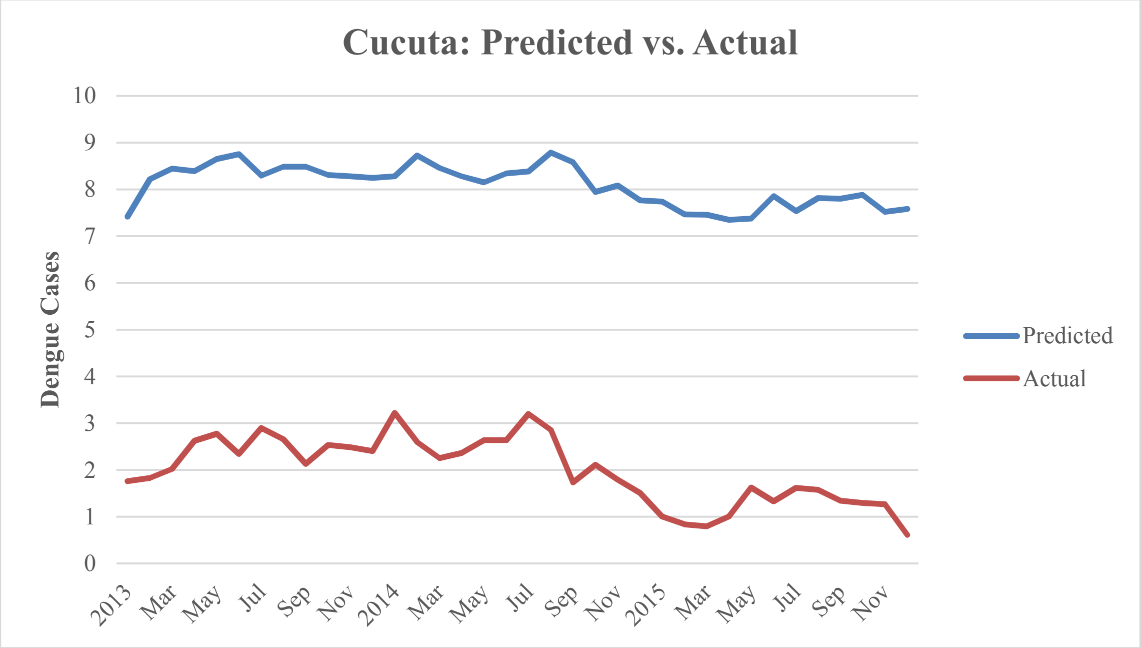


Figure 31. Graph of the number of predicted and actual dengue cases each month of the model prediction period for Cucuta.

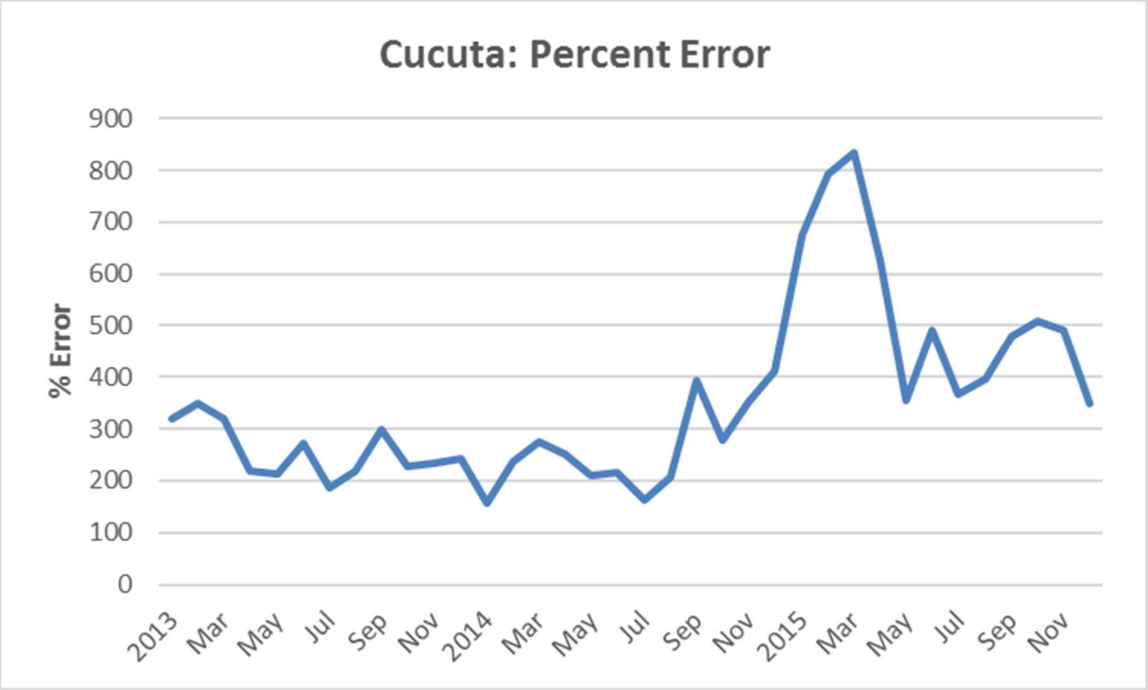


Figure 32. Graph of the percent error of the prediction model for each month of the model prediction period for Cucuta.

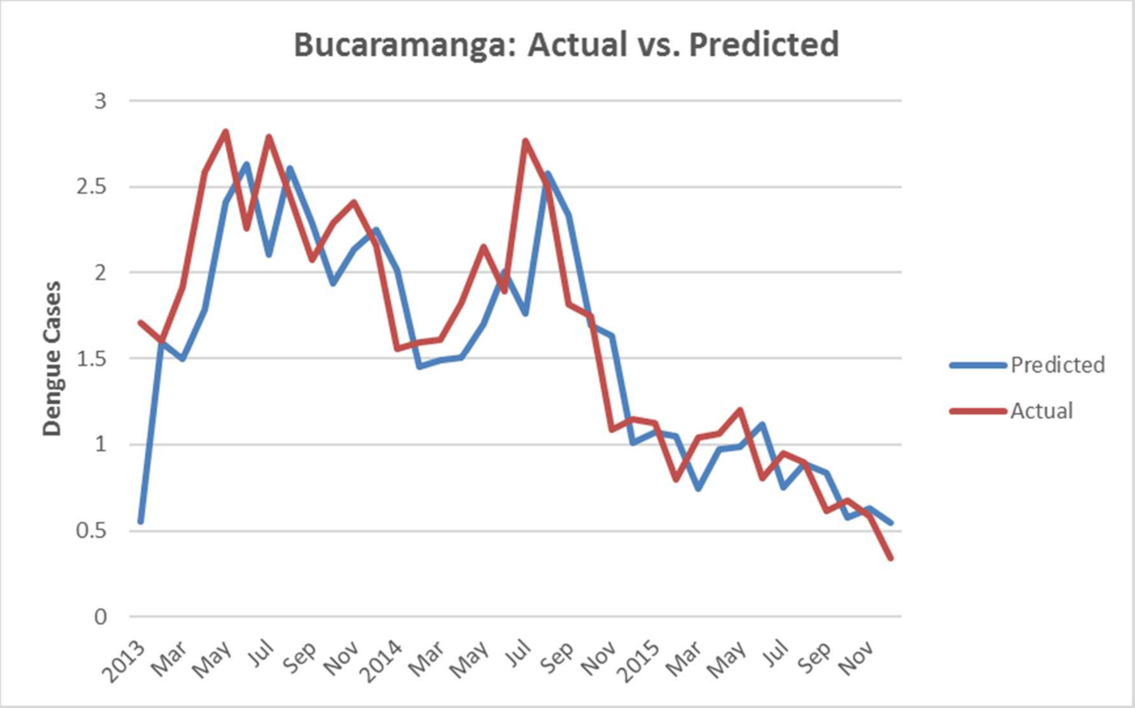


Figure 33. Graph of the number of predicted and actual dengue cases each month of the model prediction period for Bucaramanga.

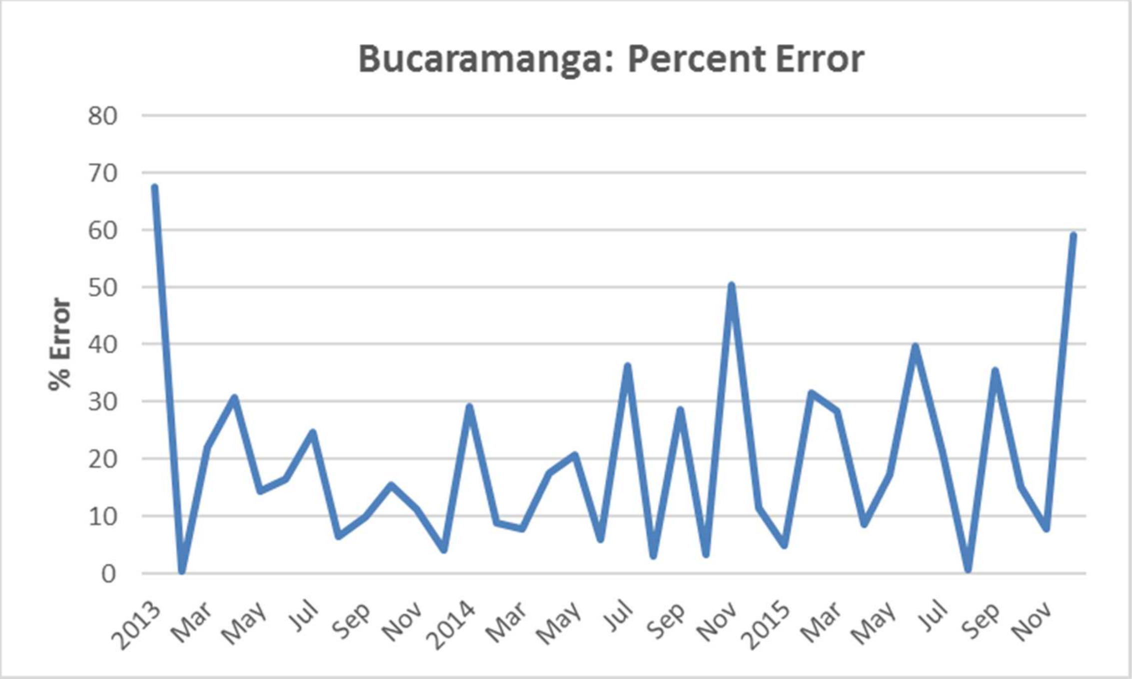


Figure 34. Graph of the percent error of the prediction model for each month of the model prediction period for Bucaramanga.

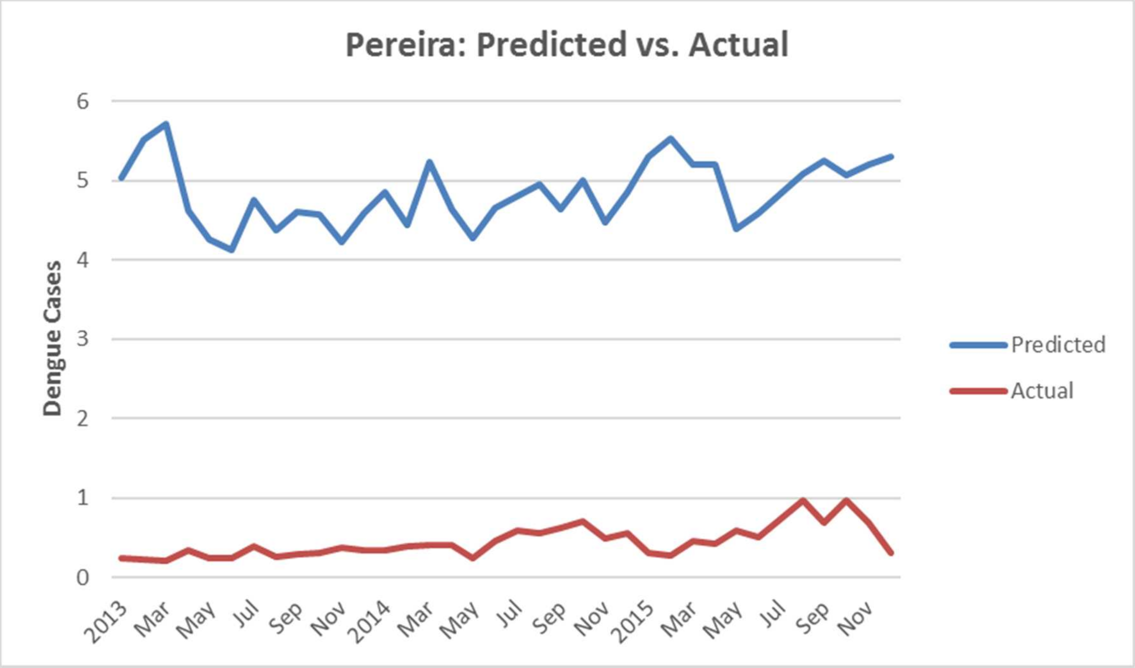


Figure 35. Graph of the number of predicted and actual dengue cases each month of the model prediction period for Pereira.

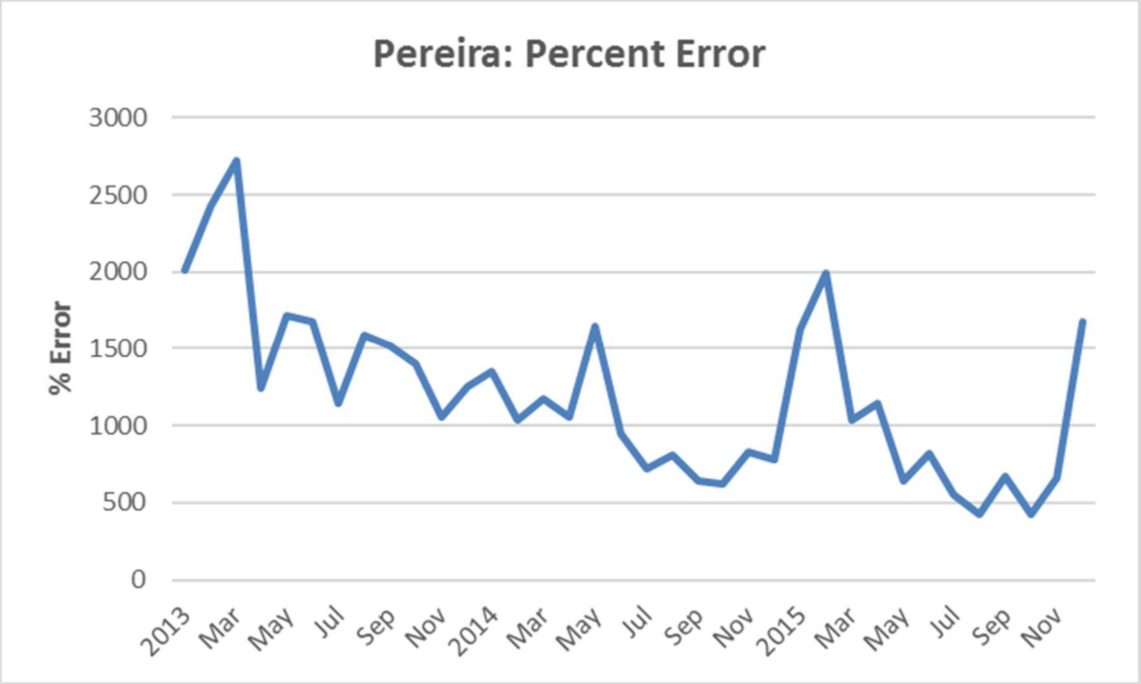


Figure 36. Graph of the percent error of the prediction model for each month of the model prediction period for Pereira.

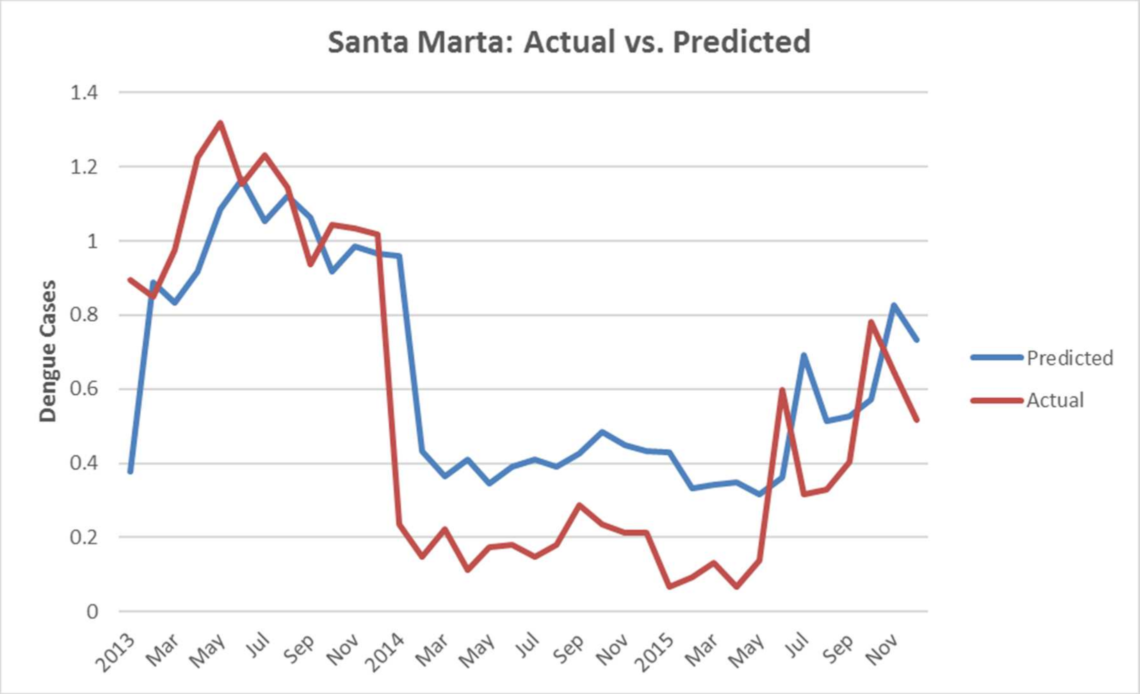


Figure 37. Graph of the number of predicted and actual dengue cases each month of the model prediction period for Santa Marta.

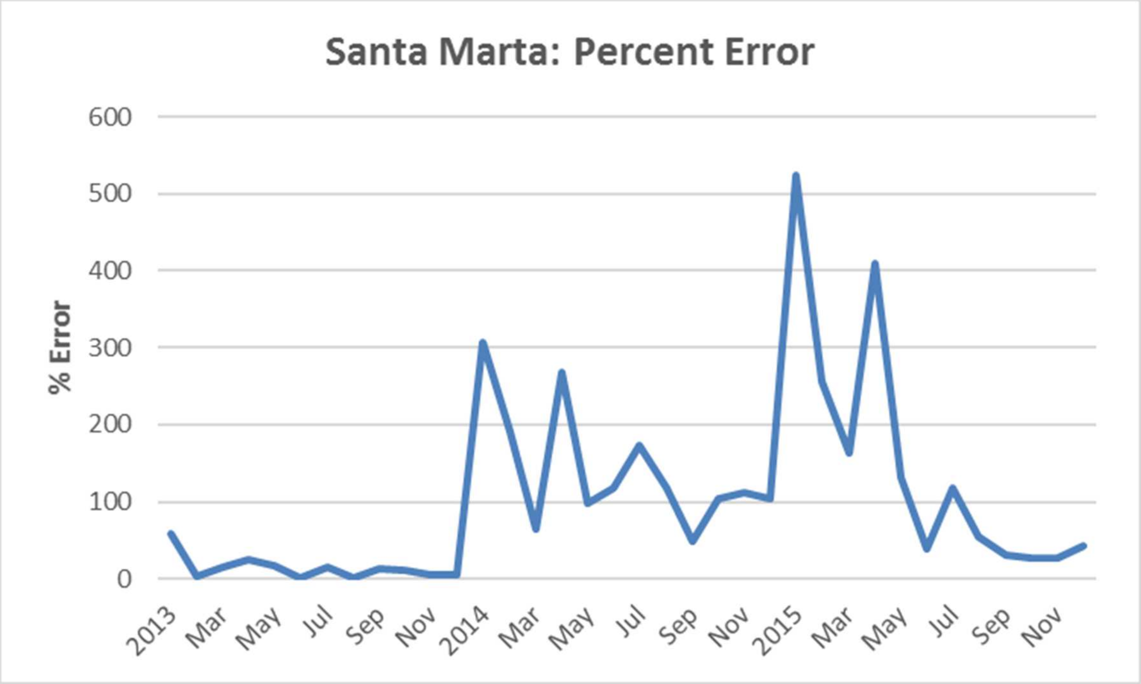


Figure 38. Graph of the percent error of the prediction model for each month of the model prediction period for Santa Marta.

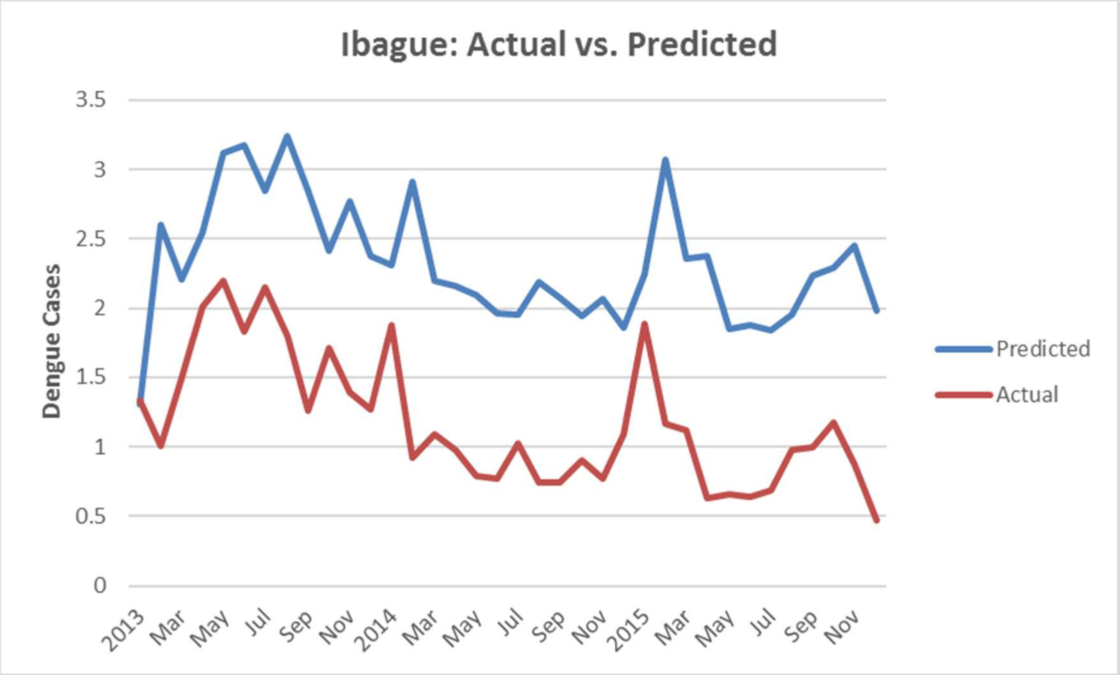


Figure 39. Graph of the number of predicted and actual dengue cases each month of the model prediction period for Ibague.

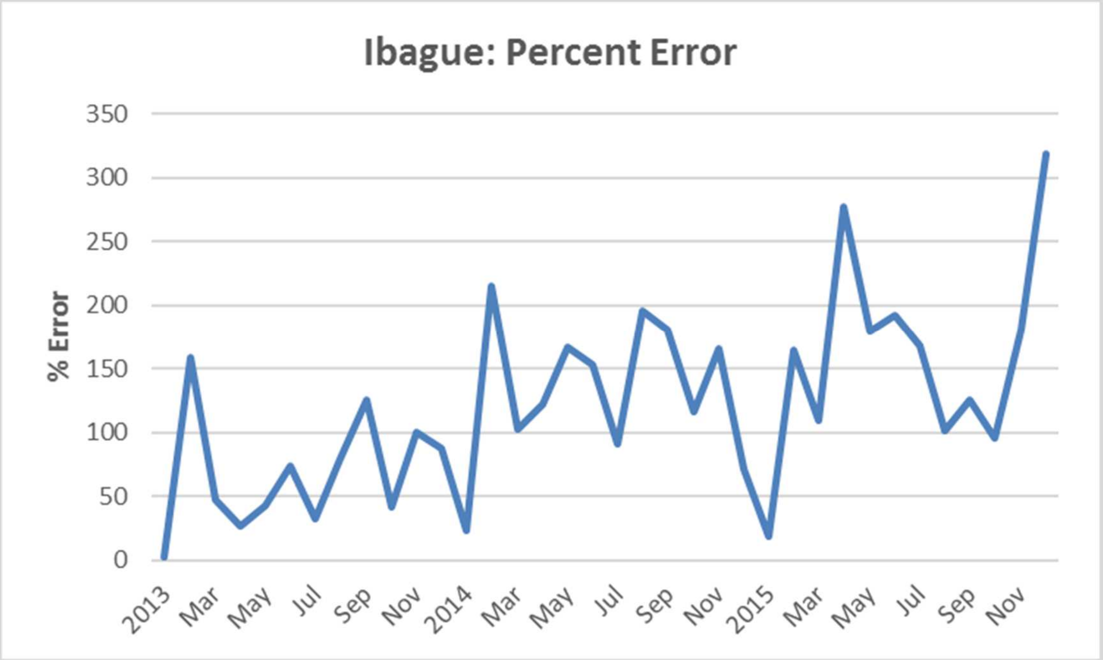


Figure 40. Graph of the percent error of the prediction model for each month of the model prediction period for Ibague.

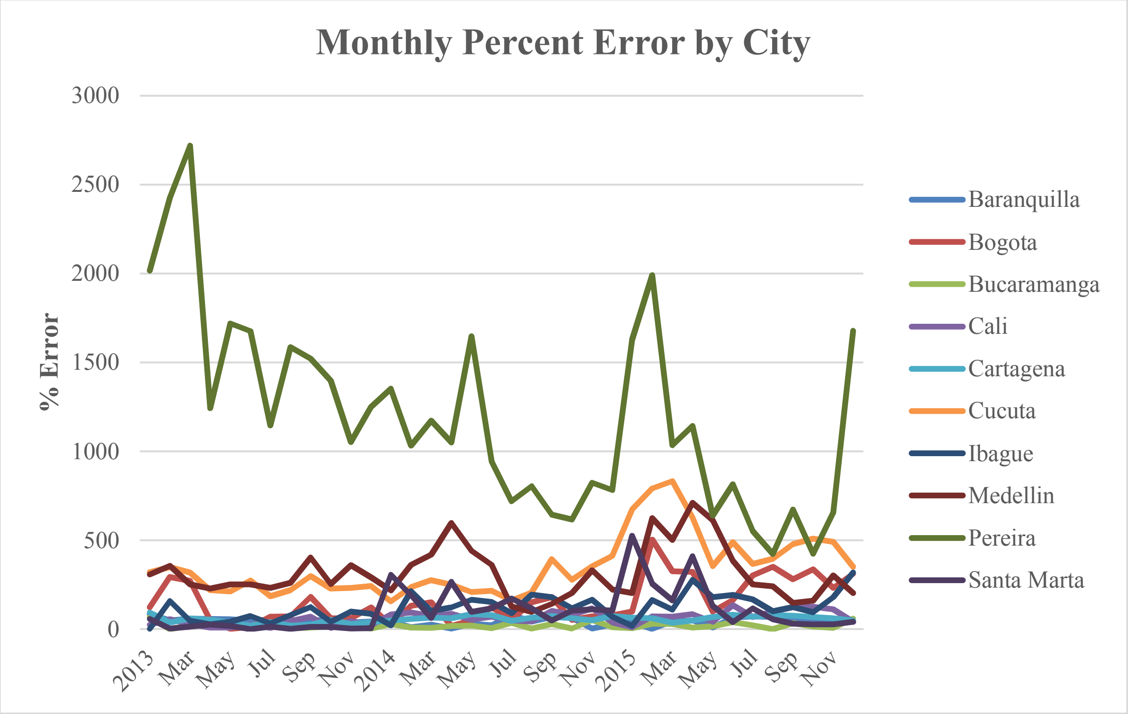


Figure 41. The monthly percent error of the prediction model for each city. Pereira experienced extreme errors.

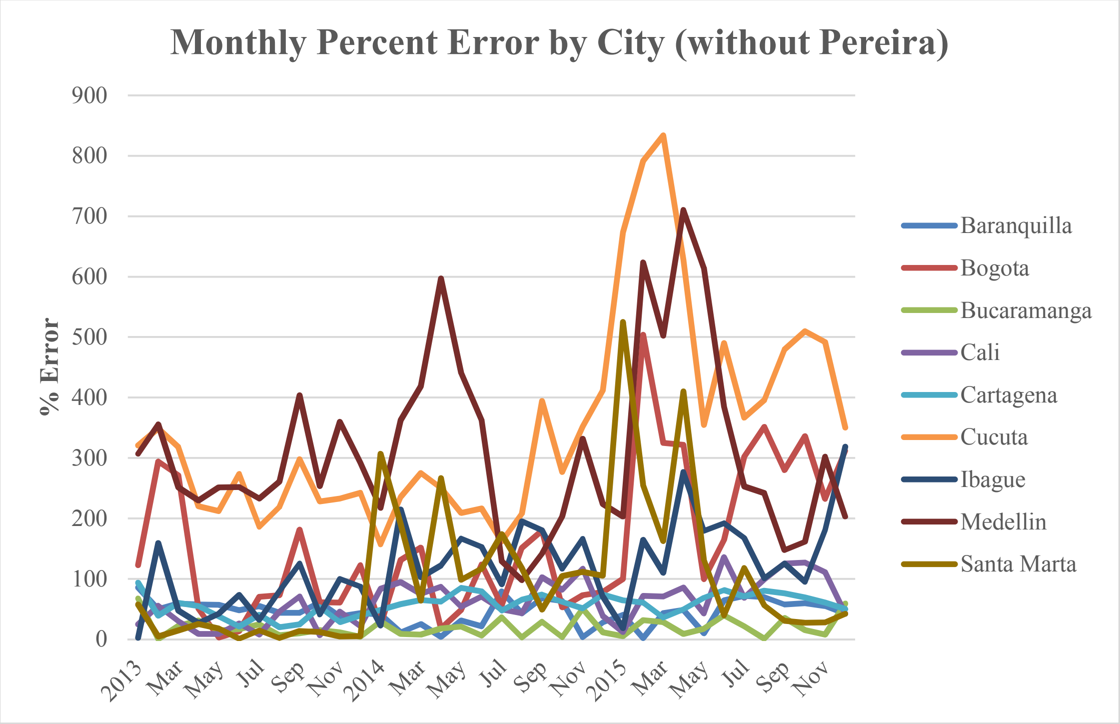


Figure 42. Monthly percent error for each city with Pereira removed so patterns can be examined more closely.

City	Pop2017	Elevation (m)	Model(s) Used	Mean Mo. Dengue Obs.	Mean Mo. Dengue Pred.	Correlation between Pred and Obs	Mean Bias
Bogota	7674366	2644	11, 9 urban center; 1, some 0	0.624780089	1.396934446	0.019889436	0.772154357
Cali	2392877	1014	0,1	7.122879933	4.858381944	0.824875995	2.264497989
Medellin	1999979	1495	11,2	11.63126315	3.247211111	0.639807511	8.640753789
Barranquilla	1380425	18	4	1.261704679	2.498812083	0.903250241	-1.237107404
Cartagena	952024	2	1, 0	0.047420444	0.605479861	0.882884645	-0.29107361
Cucuta	721398	309	1, 7, 4, 11	7.915075026	1.991695194	0.784493375	6.094263861
Bucaramanga	571820	993	0	1.532431663	1.635309444	0.835453151	-0.102877781
Pereira	440118	1411	1, 11, 4	4.711852971	0.444775194	0.12351007	4.391345221
Santa Marta	431781	9	1,3,11	0.642157504	0.535277444	0.84497713	0.099888323
Ibague	421685	1229	0, 2	2.298821192	1.179134444	0.55802805	1.146412614

Table 10. Model results by city summary. Geographic information about each city is provided, along with the mean monthly observed and predicted dengue incidence, the correlation between them and mean bias.

Table 11. The number of actual monthly dengue cases was correlated with the absolute error and percent error for all of Colombia and for the ten most populous cities. R^2 values were almost always negative, indicating that the model is more accurate when dengue incidence is high.

Correlations between Number of Actual Cases and Error		
	R^2: # of cases vs. absolute error	R^2: # of cases vs. % error
Colombia Total	-0.617265204	-0.703333219
Bogota	-0.508141675	-0.780271768
Cali	-0.480253385	-0.806627812
Medellin	-0.40851574	-0.863113719
Baranquilla	0.932228944	0.306743657
Cartagena	0.771098386	-0.489710072
Cucuta	-0.806393894	-0.874345001
Bucaramanga	0.41936884	-0.215166255
Pereira	-0.351267626	-0.855530636
Santa Marta	-0.361616737	-0.678881508
Ibague	-0.529117778	-0.827481531

DISCUSSION

Correlations between each of the weather variables and dengue incidence were found and mapped for each month at time lags of one to six months at each pixel. Correlation patterns often mimicked terrain, which makes sense because the weather data used in the correlation was derived from cokriging with elevation as a secondary variable. Correlations between maximum and mean temperature and dengue incidence at all lag times were positive for most pixels, supporting previous research aligning warmer temperatures with increased risk of dengue transmission (Eastin et al. 2014, Watts et al. 1987). At longer time lags, correlations between minimum temperature and dengue incidence were negative at high elevations, in contrast to previous research. At shorter time lags, correlations between minimum temperature and dengue incidence are mostly positive but weak at high elevations. Minimum temperatures may have less of an influence on dengue at high elevations because low temperatures usually occur at night, and *Ae. aegypti* is a day-biting mosquito (Christophers 1960). Correlations between precipitation and dengue incidence were negative for most of the country at all lag times. *Ae. aegypti* needs enough standing water to breed, but this does not require excessive rainfall. Precipitation can cool temperatures, which may also lead to a decrease in dengue cases.

The cokriging interpolation of weather variables does not consider some important factors. Temperature and precipitation are dependent on more than just elevation. Especially in the mountainous regions, precipitation is highly affected by wind and wind direction. In Colombia, the prevailing direction is easterly for most the year, so there is likely more rainfall on the windward eastern slopes of the mountains then on the leeward western sides of the mountains. The predominant wind direction also changes for some brief periods throughout the year. Wind may be an important factor that is not considered in the cokriging interpolation.

The optimal time lag for each weather variable at each pixel was identified as the time lag with the highest R^2 value at that pixel. The results in Figure 14 show make some patterns apparent. For maximum and minimum temperature and for precipitation, longer lag times dominate at higher elevations, while shorter lag times were more ideal in the valleys and along the coast. Mean temperature had the strongest correlations at shorter lags for much of the country. Temperature fluctuations can be large in the mountainous regions of Colombia, which can result in slower oviposition and virus transmission times, which could be a reason for the longer time lags being optimal at higher elevations. Along the coasts where temperatures are mild and more consistent, oviposition and transmission rates may be faster, resulting in shorter lag times between warm temperatures and high dengue incidence.

Using stepwise regression based on the R^2 values of the optimal time lags, the weather predictors that were important were incorporated into a prediction model at each pixel. The majority of pixels (including the ones that were deemed “no data” pixels because no dengue had occurred there), used Model 0 as the optimal model. Model 0 incorporated only the previous month’s dengue value. This implies that the correlations between weather and dengue were often not strong enough to be deemed influential on the number of dengue cases that occurred at each pixel. Only 10 of the 16 potential models were used in the overall prediction model. Certain combinations of predictors did not adequately increase the correlation between the predictors and dengue incidence.

Three overarching ideas explain the results of this prediction model. One is that the accuracy of each of the ten individual prediction models, and where each model was deemed the optimal model, is crucial to understanding the errors in prediction. Two is that the number of dengue cases is influential on the accuracy of the model. Three is that while weather variables such as temperature and precipitation have consistently been strongly correlated to dengue incidence in the literature, there are many other factors at play besides weather. The availability of mosquito breeding grounds, mosquito eradication programs or lack thereof, population growth and urbanization and human travel are all factors that affect when, where and how many dengue cases will occur (Carbajo et al. 2012).

When considering the results on the scale of the entire country, the overall model consistently over-predicted the number of monthly dengue cases. Whereas actual dengue was concentrated in a few small areas (major cities), high amounts of dengue were predicted across widespread regions. Each month, the model predicted about 100 more dengue cases across the country than actually occurred. This over-prediction is likely due to the incorporation of weather variables to the model. When a weather variable was deemed to significantly improve the correlation between the predictors and dengue incidence, it was added to the model. When these correlations were positive, it means that the outcome of the model was higher than if only incorporating the previous month's dengue. When the correlations of the added variables were negative, this would subtract from the number of cases predicted by the model. Using Model 4 as an example:

$$\hat{y} = \beta_0 + \beta_{Dpop} D_{POP} + \beta_{Ptot} P_{TOT}.$$

Here, β_{Ptot}, the regression coefficient, is negative because the correlation between dengue and precipitation was negative for a pixel. This would decrease the value of $\hat{y}$, the total number of predicted dengue cases. This led to Model 4 having a lower error than many of the other models, because it caused less over-prediction. Model 7 had the highest percent error. The equation for Model 7:

$$\hat{y} = \beta_0 + \beta_{Dpop}D_{POP} + \beta_{Tmax}T_{MAX} + \beta_{Ptot}P_{TOT}$$

and Figures 10, 13 and 15 explain why Model 7 resulted in large errors and over-prediction. From Figure 15, the areas where Model 7 was found to be the optimum model can be identified. When compared to Figure 10, these are areas with strong positive correlations between maximum temperature and dengue incidence, resulting in a large, positive regression coefficient. When compared to Figure 13, these are areas with weak negative correlation, resulting in a small, negative regression coefficient. The end result is an over-prediction of dengue incidence.

Monthly percent error of the overall model for Colombia ranged from 83.77% in May 2013 to 768% in September 2015. The average percent error for the entire country was 232%. Percent error was higher consistently higher at the end of the prediction period than at the beginning. Comparing Figures 18 and 19 shows that percent error starts to increase around October 2014, which coincides exactly with when the number of actual dengue cases begins to consistently decrease or stay low. Correlations between the number of cases and error were strongly negative as shown in Table 11. R^2 between the actual number of cases and absolute error for Colombia as a whole was -0.617 and R^2 between the actual number of cases and percent error for Colombia was -0.703. This implies that the model is more accurate when dengue incidence is high, which is promising because it is important to predict when large outbreaks may occur.

Because the pixels were very small and because it is important to get an idea of model performance in areas of high population, the output of the model was aggregated and analyzed for the ten most populous cities in Colombia. Of these ten cities, Bucaramanga consistently had the lowest percent error between predicted and actual dengue cases, and Pereira had the highest. Each pixel that was aggregated to make up Bucaramanga used Model 0 as its prediction model (which overall resulted in the lowest percent error country-wide), thus giving Bucaramanga the lowest overall percent error. The pixels that made up Pereira mostly used Model 1, which had the third highest error of all of the models. Cucuta, which generally boasted the second largest monthly percent errors, was made up of pixels predicted by models 1 and 7, two of the worst models. The influence that the autoregressive dengue term had on each of the models and in each of the cities is apparent in all of the graphs of predicted vs. actual dengue incidence. The peaks (dips) of the prediction graph always follow the month that the peaks (dips) occur on the actual dengue incidence graph.

Seven out of ten cities had a MAPE lower than the national average of 232%. MAPE for each city is displayed on Table 9. The three cities that were above the national average percent error were Medellin, Cucuta and Pereira. This generally implies that the model works better in areas of high population.

All of the cities generally over-predicted dengue incidence except for Barranquilla, Cartagena and Santa Marta. (Santa Marta had some under-prediction and some over-prediction).These three cities lie on the northwest coast of Colombia. These three cities did not use the same models for their predictions, so it is interesting that they all under-predicted dengue incidence. These three cities also experienced some of the lowest monthly percent errors of the ten cities analyzed. In contrast to the country as a whole and to most of the other cities, absolute error in both Baranquilla and Cartagena were strongly positively correlated to dengue cases. That means that in those cities, the model is more accurate when the number of dengue cases is low. In the rest of the cities, (besides Bucaramanga, where error was weakly negatively correlated to dengue), percent error was strongly negatively correlated to actual dengue incidence. This agrees with the country-wide trend that the overall prediction model is generally more accurate when dengue incidence is high. This also illustrates the importance of examining the results at different spatial scales. Though the country may experience one trend, some areas (in this case, some highly populated areas), do not follow these general trends.

CONCLUSION

Though the prediction model resulted in high errors for some cities and in some months, some important conclusions can be drawn from this study. A dengue prediction model was created for an entire country, whereas previous research has focused on small areas. Determining the optimal lag between each weather variable and dengue incidence at the pixel level over such a wide area is a novel contribution of this study. Such a model can be constructed with publicly available data from weather stations, which is then interpolated, and highly aggregated epidemiological data using population-weighted dengue incidence. A potential benefit of this research is that a similar model could be applicable to other countries with very aggregated epidemiological data. The model could be improved if more weather stations were available to enhance the interpolation. The prediction could also be improved if the number of dengue cases were available at a scale smaller than the departmental level to gain a better understanding of the spatial distribution of reported dengue cases. Interpolation of weather data and disaggregation of dengue incidence data caused uncertainty in the model outcome. Neither the cokriging interpolation nor the population-weighted dengue incidence are perfect representations of reality, which is likely an important reason for large errors in the model.

Another benefit of this model is that it is mostly automated. However, the cokriging process in this study was not automated. Using the geostatistical wizard in ArcMap must be done individually for each weather variable for each month. If more variables or a longer time period was used, this would have been a very tedious process. It is possible to complete cokriging in R, however this proved to be difficult for this study. It would be an improvement to the model if cokriging was completed in R to make the model fully automated, and provide consistency in the interpolation processes. Another drawback is that this study only examined linear models. It would be beneficial to test whether non-linear models increased prediction accuracy.

Overall, the dengue prediction model over-predicted the dengue that actually occurred in Colombia from 2013-2015. The most accurate of the equations used in the prediction model was Model 0, which was based solely on the number of dengue cases that occurred at a pixel the previous month. The autoregressive dengue correlations at every pixel were always strongly positive, and much higher than correlations between dengue incidence and any of the weather variables. Adding weather predictors to the regression equations resulted in over-prediction of dengue. However, general or country-wide over-prediction may not hold true for certain areas, which is why it is important to assess model performance at multiple spatial scales.

Focusing analysis on where the majority of the population lives is important because that is where the majority of dengue cases occur, regardless of weather variables. Dengue is rare in areas of low population, especially in the rainforest and savanna regions of Colombia, so over-predictions in areas of low population could be accounted for in a model because no dengue cases have been reported in those areas. The model developed in this study generally had lower errors in areas of high population than the average error of the entire country.

While it has been well-established that strong correlations exist between weather factors, such as temperature and precipitation, and dengue incidence, there are multiple other factors at work. Population growth, pesticide spraying and the availability of mosquito breeding grounds are just a few examples of other variables that may influence dengue incidence that were not analyzed in this study. To create a more accurate dengue prediction model over a large area, socio-economic factors need to be taken into account in addition to environmental factors.

REFERENCES

Barrera, R., M. Amador, and A. J. MacKay, 2011: Population dynamics of Aedes aegypti and dengue as influenced by weather and human behavior in San Juan, Puerto Rico. *PLoS Neglected Tropical Diseases*, **5**.

Box, G. E. P. and Jenkins, G. M., 1976: *Time Series Analysis: Forecasting and Control, Rev. Edition*. Holden-Day.

Carbajo, A. E., M. V. Cardo and D. Vezzani, 2012: Is temperature the main cause of dengue rise in non-endemic countries? The case of Argentina. *International Journal of Health Geographics*, **11**.

Chai, T. and R.R. Draxler, 2014: Root mean square error (RMSE) or mean absolute error (MAE)?—Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, **7**, 1247-1250.

Clark, I., 1979: *Practical geostatistics*. Elsevier Applied Science.

Christophers, S., 1960: *Aedes aegypti (L.) the yellow fever mosquito: its life history, bionomics and structure*. Rickard.

Durbin, A. P., 2016: A dengue vaccine. *Cell*, **166**, 1.

Descloux, E., M. Mangeas, C.E. Menkes, M. Lengaigne, A. Leroy, T. Tehei, L. Guillaumot, et al., 2012: Climate-based models for understanding and forecasting dengue epidemics. *PLoS Neglected Tropical Diseases*, **6**.

Eastin, M. D., E. Delmelle, I. Casas, J. Wexler, and C. Self, 2014: Intra- and interseasonal autoregressive prediction of dengue outbreaks using local weather and regional climate for a tropical environment in Colombia. *American Journal of Tropical Medicine and Hygiene* **91**, 598–610.

Global Historic Climatology Network (GHCN), 2016. *National Climatic Data Center*. https://www.ncdc.noaa.gov/data-access/land-based-station-data/land-based-datasets/global-historical-climatology-network-ghcn

Goovaerts, P., 2000: Geostatistical approaches for incorporating elevation into the spatial interpolation of rainfall. *Journal of Hydrology*, **228**, 113–129.

Gubler, D. J., 1998: Dengue and dengue hemorrhagic fever. *Clinical Microbiology Reviews*, **11**, 480-496.

Hii, Y. L., H. Zhu, N. Ng, L. C. Ng, and J. Rocklöv, 2012: Forecast of dengue incidence using temperature and rainfall. *PLoS Neglected Tropical Diseases*, **6**.

Hopp, M. J. and J. A. Foley, 2003: Worldwide fluctuations in dengue fever cases related to climate variability. *Climate Research*, **25**, 85–94.

Ishida, T. and S. Kawashima, 1993: Use of co-kriging to estimate surface air temperature from elevation. *Theoretical and Applied Climatology*, **47**, 147–157.

Jetten, T. H. and D. A. Focks, 1997: Potential changes in the distribution of dengue transmission under climate warming. *American Journal of Tropical Medicine and Hygiene*, **57**, 285–97.

Lambrechts, L., K. P. Paaijmans, T. Fansiri, L. B. Carrington, L. D. Kramer, M. B. Thomas, and T. W. Scott, 2011: Impact of daily temperature fluctuations on dengue virus transmission by Aedes aegypti. *Proceedings of the National Academy of Sciences*, **108**, 7460–7465.

Li, J., & Heap, A. D., 2011: A review of comparative studies of spatial interpolation methods in environmental sciences: performance and impact factors. *Ecological Informatics*, **6**, 228–241.

Oliver, M., R. Webster, and J. Gerrard, 1989: Geostatistics in physical geography. Part I: Theory. *Transactions of the Institute of British Geographers, New Series*, **14**, 259–269.

Pfeifer, P. E. and S. J. Deutsch, 1980: A STARIMA model-building procedure with application to description and regional forecasting. *Transactions of the Institute of British Geographers*, **3**, 330-349.

Pitisuttithum, P. and A. Bouckenooghe, 2016: The first licensed dengue vaccine: an important tool for integrated preventive strategies against dengue virus infection. *Expert Review of Vaccines*, **15**, 795-798.

Rueda, L.M., K.J. Patel, R. C. Axtell, R. E. Stinner, 1990: Temperature-dependent development and survival rates of Culex quinquefasciatus and Aedes aegypti (Diptera: Culicidae). *Journal of Medical Entomology*, **27**, 892-898.

SAS Institute Inc., 2014: *SAS/ETS 13.2 User's Guide*. SAS Institute Inc.

Scott, T. W., P.H. Amerasinghe, A. C. Morrison, L. H. Lorenz, G. G. Clark, D. Strickman et al., 2000: Longitudinal Studies of Aedes aegypti (Diptera: Culicidae) in Thailand and Puerto Rico: Blood Feeding Frequency. *Journal of Medical Entomology*, **37**, 89-101.

Suaya, J. A., D. S. Shepard and M. E. Beatty, 2007: Dengue: burden of disease and costs of illness. *TDR. Report of the Scientific Working Group Meeting on Dengue,* **1**, 35-49.

Watts, D. M., D. S. Burke, B. A. Harrison, R. E. Whitmire, A. Nisalak, 1987: Effect of temperature on the vector efficiency of Aedes aegypti for dengue 2 virus. *American Journal of Tropical Medecine and Hygiene* **36**, 143-152.

Wilks, D., 1995: *Statistical methods in the atmospheric sciences*. 1st ed. Academic Press, San Diego, CA.

Wu, P.-C., H.-R. Guo, S.-C. Lung, C.-Y. Lin, and H.-J. Su, 2007: Weather as an effective predictor for occurrence of dengue fever in Taiwan. *Acta Tropica* **103**, 50–57.

Yang, J. S., Wang, Y. Q., & August, P. V., 2004: Estimation of land surface temperature using spatial interpolation and satellite-derived surface emissivity. *Journal of Environmental Informatics*, **4**, 37–44.

Yang, H. M., M. L. G. Marcoris, K. C. Galvani, M. T. M. Andrighetti, and D. M. V. Wanderly, 2009: Assessing the effects of temperature on the population of Aedes aegypti, the vector of dengue. *Epidemiology and Infection*, **137**, 1188-1202.

Zimmerman, D. P., 1999: An experimental comparison of ordinary and universal kriging and Inverse Distance Weighting. *Mathematical Geology*, **31**, 375-390.

APPENDIX: CODES

Population-Weighted Dengue Incidence Code (Python)

```
# pop weighted dengue raster

import arcpy
from arcpy import env
from arcpy.sa import *
import os

arcpy.env.workspace = "C:\\1megan\\1research\\DengueData\\UTM"
# batch project in arcmap

dpoly = arcpy.ListFiles("*.shp")


mo = ["JAN", "FEB", "MAR", "APR", "MAY", "JUN", "JUL", "AUG", "SEPT", "OCT",
"NOV", "DEC"]

count1 = 0
for i in dpoly:
                        count1 += 1
                        count2 = 0
                        for j in mo:
                            f = str(i) + "_" + str(j) + ".lyr"
                            arcpy.PolygonToRaster_conversion(i, j, f)
                            count2 += 1

drast = arcpy.ListFiles("*.tif")
cellpop = Raster("pop05clip")
deptpop = Raster("popdept05fl") #float raster (rather than int)

myList = []
for i in drast:
                        myList.append(Raster(i))

for i in myList:
                        outR = Raster(cellpop) / Raster (deptpop) *
Raster(str(i))
                        orn = "PD_" + str(i) + ".tif"
                        outR.save(orn)


# get rid of NoData cells
```

```
arcpy.env.workspace = "C:\\1megan\\1research\\PopDengue"
outws = "C:\\1megan\\1research\\PopDengue\\NoNull"
inRas = arcpy.ListFiles("*.tif")
myList = []
for i in inRas:
                        myList.append(Raster(i))

for j in myList:
                        string = str(j)
                        string = string[:-4]
                        f = "nona_" + string
                        filled =
arcpy.sa.Con(arcpy.sa.IsNull(j),arcpy.sa.FocalStatistics(j,
arcpy.sa.NbrRectangle(5,5),"MEAN"),j)
                        filled.save(os.path.join(outws,f))


# Clip to outline

arcpy.env.workspace = "C:\\1megan\\1research\\PopDengue\\NoNull"
outws = "C:\\1megan\\1research\\PopDengue\\Final"
inRas = arcpy.ListFiles("*.tif")
myList = []
for i in inRas:
                        myList.append(Raster(i))

for j in myList:
                        string = str(j)
                        string2 = string[15:19]+string[23:-4]
                        f = "PD_" + string2
                        arcpy.Clip_management(j, "", os.path.join(outws,f),
"splitPoly", "", "ClippingGeometry", "MAINTAIN_EXTENT")
```

Inverse Distance Weighting Interpolation Code (R)

```
# install and load packages
install.packages("gstat",repos="http://cran.r-
project.org",destdir="C:/1megan/1research/Kriging")
install.packages("automap",repos="http://cran.r-
project.org",destdir="C:/1megan/1research/Kriging")
install.packages("sp",repos="http://cran.r-
project.org",destdir="C:/1megan/1research/Kriging")
install.packages("maptools",repos="http://cran.r-
project.org",destdir="C:/1megan/1research/Kriging")
install.packages("raster",repos="http://cran.r-
project.org",destdir="C:/1megan/1research/Kriging")
install.packages("rgdal",repos="http://cran.r-
project.org",destdir="C:/1megan/1research/Kriging")

library(gstat)
library(sp)
library(automap)
library(maptools)
library(raster)
library(rgdal)

# read in data
COraster<- raster("C:/1megan/1research/Kriging/CO_raster_UTM")
grid<- SpatialPixels(SpatialPoints(coordinates(COraster)))
# set projections
projection(grid)<- "+proj=utm +zone=18 +south +ellps=WGS84 +towgs84=0,0,0,0,0,0,0
+units=m +no_defs"

# MAXIMUM TEMPERAUTRE

# read in MaxTemp data
setwd("C:/1megan/1research/KrigeUTM/MaxT")
list.maxT<-list.files(pattern=".shp$")
list.data<-list()
for (i in 1:length(list.maxT))
{
                                    list.data[[i]]<-readShapePoints(list.maxT[i])
}

# assign names to each file
n.d<-names(list.data)<-list.maxT

# IDW
setwd("C:/1megan/1research/IDWPlots/UTM/MaxT")
```

```
x=1
for (i in list.data)
{
				# idw interpolation
				projection(i) <- "+proj=utm +zone=18 +south
+ellps=WGS84 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs"
				MaxT_idw<-idw(MAX_T~1, i, grid)

				# Leave-one-out cross validation
				MaxT_cv<-krige.cv(MAX_T~1, i)
				resid<-MaxT_cv$residual
				z_score<-MaxT_cv$zscore
				obs<-MaxT_cv$observed

				# accuracy assessment
				ME<-mean(resid)			# Mean Error, ideal is
0
				MPSE<-mean(resid^2)		# MPSE, ideally small
				MSNE<-mean(z_score^2)			# Mean square
normalized error, ideally close to 1
				COP<-cor(obs, obs-resid)		# cor observed and
predicted, ideally 1
				CPR<-cor(obs-resid, resid)		# cor predicted and
residual, ideally 0
				write.table(data.frame(ME, MPSE, MSNE, COP,
CPR),file="C:/1megan/1research/AccuracyAssess/IDW_MaxT.csv", append=TRUE,
sep=",", row.names=FALSE, col.names=FALSE)


				# save to output raster
				r<-raster(MaxT_idw)
				r.m<-mask(r, COraster)
				f<-paste0('idw_', n.d, '.tif')
				writeRaster(r.m, filename=f[x], overwrite=TRUE)
				x=x+1
}

# MEAN TEMPERATURE

# read in MeanTemp data
setwd("C:/1megan/1research/KrigeUTM/MeanT")
list.meanT<-list.files(pattern=".shp$")
list.data<-list()
for (i in 1:length(list.meanT))
{
				list.data[[i]]<-readShapePoints(list.meanT[i])
```

```
}

# assign names to each file
n.d<-names(list.data)<-list.meanT

# IDW
setwd("C:/1megan/1research/IDWPlots/UTM/MeanT")
x=1
for (i in list.data)
{
                        # idw interp
                        projection(i) <- "+proj=utm +zone=18 +south
+ellps=WGS84 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs"
                        MeanT_idw<- idw(MEAN_T~1, i, grid)

                        #LOOCV
                        # Leave-one-out cross validation
                        MeanT_cv<-krige.cv(MEAN_T~1, i)
                        resid<-MeanT_cv$residual
                        z_score<-MeanT_cv$zscore
                        obs<-MeanT_cv$observed

                        # accuracy assessment
                        ME<-mean(resid)                # Mean Error, ideal is
0
                        MPSE<-mean(resid^2)            # MPSE, ideally small
                        MSNE<-mean(z_score^2)               # Mean square
normalized error, ideally close to 1
                        COP<-cor(obs, obs-resid)       # cor observed and
predicted, ideally 1
                        CPR<-cor(obs-resid, resid)     # cor predicted and
residual, ideally 0
                        write.table(data.frame(ME, MPSE, MSNE, COP,
CPR),file="C:/1megan/1research/AccuracyAssess/IDW_MeanT.csv", append=TRUE,
sep=",", row.names=FALSE, col.names=FALSE)

                        # write output to raster
                        r<-raster(MeanT_idw)
                        r.m<-mask(r, COraster)
                        f<-paste0('idw_', n.d, '.tif')
                        writeRaster(r.m, filename=f[x], overwrite=TRUE)
                        x=x+1
}

# MINIMUM TEMPERATURE
```

```
# read in MinTemp data
setwd("C:/1megan/1research/KrigeUTM/MinT")
list.minT<-list.files(pattern=".shp$")
list.data<-list()
for (i in 1:length(list.minT))
{
                        list.data[[i]]<-readShapePoints(list.minT[i])
}

# assign names to each file
n.d<-names(list.data)<-list.minT

# IDW
setwd("C:/1megan/1research/IDWPlots/UTM/MinT")
x=1
for (i in list.data)
{
                        # idw interp
                        projection(i) <- "+proj=utm +zone=18 +south
+ellps=WGS84 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs"
                        MinT_idw<- idw(MAX_T~1, i, grid)

                        #LOOCV
                        # Leave-one-out cross validation
                        MinT_cv<-krige.cv(MAX_T~1, i)
                        resid<-MinT_cv$residual
                        z_score<-MinT_cv$zscore
                        obs<-MinT_cv$observed

                        # accuracy assessment
                        ME<-mean(resid)               # Mean Error, ideal is
0
                        MPSE<-mean(resid^2)           # MPSE, ideally small
                        MSNE<-mean(z_score^2)               # Mean square
normalized error, ideally close to 1
                        COP<-cor(obs, obs-resid)      # cor observed and
predicted, ideally 1
                        CPR<-cor(obs-resid, resid)    # cor predicted and
residual, ideally 0
                        write.table(data.frame(ME, MPSE, MSNE, COP,
CPR),file="C:/1megan/1research/AccuracyAssess/IDW_MinT.csv", append=TRUE,
sep=",", row.names=FALSE, col.names=FALSE)


                        # write output to raster
                        r<-raster(MinT_idw)
```

```
                    r.m<-mask(r, COraster)
                    f<-paste0('idw_', n.d, '.tif')
                    writeRaster(r.m, filename=f[x], overwrite=TRUE)
                    x=x+1
}

# PRECIPITATION
# read in Precip data
setwd("C:/1megan/1research/KrigeUTM/Precip")
list.Precip<-list.files(pattern=".shp$")
list.data<-list()
for (i in 1:length(list.Precip))
{
                    list.data[[i]]<-readShapePoints(list.Precip[i])
}

# assign names to each file
n.d<-names(list.data)<-list.Precip

# IDW
setwd("C:/1megan/1research/IDWPlots/UTM/Precip")
x=1
for (i in list.data)
{
                    # idw interp
                    projection(i) <- "+proj=utm +zone=18 +south
+ellps=WGS84 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs"
                    Precip_idw<- idw(PRECIP~1, i, grid)

                    # #LOOCV
                    # Leave-one-out cross validation
                    Precip_cv<-krige.cv(PRECIP~1, i)
                    resid<-Precip_cv$residual
                    z_score<-Precip_cv$zscore
                    obs<-Precip_cv$observed

                    # accuracy assessment
                    ME<-mean(resid)              # Mean Error, ideal is
0
                    MPSE<-mean(resid^2)          # MPSE, ideally small
                    MSNE<-mean(z_score^2)             # Mean square
normalized error, ideally close to 1
                    COP<-cor(obs, obs-resid)     # cor observed and
predicted, ideally 1
                    CPR<-cor(obs-resid, resid)   # cor predicted and
residual, ideally 0
```

```
                                        write.table(data.frame(ME, MPSE, MSNE, COP,
CPR),file="C:/1megan/1research/AccuracyAssess/IDW_Precip.csv", append=TRUE,
sep=",", row.names=FALSE, col.names=FALSE)

                                        # write output to raster
                                        r<-raster(Precip_idw)
                                        r.m<-mask(r, COraster)
                                        f<-paste0('idw_', n.d, '.tif')
                                        writeRaster(r.m, filename=f[x], overwrite=TRUE)
                                        x=x+1
}
```

Kriging Interpolation of Weather Variables (R)

```
# open packages
library(gstat)
library(sp)
library(automap)
library(maptools)
library(raster)
library(rgdal)

# read in data
COraster<- raster("C:/1megan/1research/Kriging/CO_raster_UTM")
grid<- SpatialPixels(SpatialPoints(coordinates(COraster)))

# set projections
projection(grid)-> "+proj=utm +zone=18 +south +ellps=WGS84 +towgs84=0,0,0,0,0,0,0
+units=m +no_defs"

# MAXIMUM TEMPERATURE
# read in MaxTemp data
setwd("C:/1megan/1research/KrigeUTM/MaxT")
list.maxT<-list.files(pattern=".shp$")
list.data<-list()
for (i in 1:length(list.maxT))
{
	list.data[[i]]<-readShapePoints(list.maxT[i])
}

# assign names to each file
n.d<-names(list.data)<-list.maxT

# Krige
setwd("C:/1megan/1research/KrigPlots/MaxT")
x=1
for (i in list.data)
{
	# kriging interp
	projection(i) <- "+proj=utm +zone=18 +south +ellps=WGS84
+towgs84=0,0,0,0,0,0,0 +units=m +no_defs"
	MaxT_kr<- autoKrige(MAX_T~1, i, grid, model=c("Sph", "Exp", "Gau", "Ste"))

	# LOOCV
	MaxT_cv<- krige.cv(MAX_T~1, i, model=MaxT_kr$var_model)
	resid<-MaxT_cv$residual
	z_score<-MaxT_cv$zscore
	obs<-MaxT_cv$observed
```

```
	# accuracy assessment
	ME<-mean(resid)			# Mean Error, ideal is 0
	MPSE<-mean(resid^2)			# MPSE, ideally small
	MSNE<-mean(z_score^2)		# Mean square normalized error, ideally
close to 1
	COP<-cor(obs, obs-resid)	# cor observed and predicted, ideally 1
	CPR<-cor(obs-resid, resid)	# cor predicted and residual, ideally 0
	write.table(data.frame(ME, MPSE, MSNE, COP,
CPR),file="C:/1megan/1research/AccuracyAssess/kr_MaxT.csv", append=TRUE,
sep=",", row.names=FALSE, col.names=FALSE)


	# write output to raster
	kr.pred<-MaxT_kr$krige_output
	r<-raster(kr.pred["var1.pred"])
	r.m<-mask(r, COraster)
	f<-paste0('kr_', n.d, '.tif')
	writeRaster(r.m, filename=f[x], overwrite=TRUE)
	x=x+1
}

# MEAN TEMPERAURE
# read in MeanTemp data
setwd("C:/1megan/1research/KrigeUTM/MeanT")
list.meanT<-list.files(pattern=".shp$")
list.data<-list()
for (i in 1:length(list.meanT))
{
	list.data[[i]]<-readShapePoints(list.meanT[i])
}

# assign names to each file
n.d<-names(list.data)<-list.meanT

#Krige
setwd("C:/1megan/1research/KrigPlots/MeanT")
x=1
for (i in list.data)
{
	# kriging interp
	projection(i)<- "+proj=utm +zone=18 +south +ellps=WGS84
+towgs84=0,0,0,0,0,0,0 +units=m +no_defs"
	MeanT_kr<- autoKrige(MEAN_T~1, i, grid, model=c("Sph", "Exp", "Gau",
"Ste"))
```

```
        # LOOCV
        MeanT_cv<- krige.cv(MEAN_T~1, i, model=MeanT_kr$var_model)
        resid<-MeanT_cv$residual
        z_score<-MeanT_cv$zscore
        obs<-MeanT_cv$observed

        # Accuracy Assessment
        ME<-mean(resid)               # Mean Error, ideal is 0
        MPSE<-mean(resid^2)                   # MPSE, ideally small
        MSNE<-mean(z_score^2)                 # Mean square normalized error, ideally
close to 1
        COP<-cor(obs, obs-resid)      # cor observed and predicted, ideally 1
        CPR<-cor(obs-resid, resid)    # cor predicted and residual, ideally 0
        write.table(data.frame(ME, MPSE, MSNE, COP,
CPR),file="C:/1megan/1research/AccuracyAssess/kr_MeanT.csv", append=TRUE,
sep=",", row.names=FALSE, col.names=FALSE)


        # write output to raster
        kr.pred<-MeanT_kr$krige_output
        r<-raster(kr.pred["var1.pred"])
        r.m<-mask(r, COraster)
        f<-paste0('kr_', n.d, '.tif')
        writeRaster(r.m, filename=f[x], overwrite=TRUE)
        x=x+1
}

# MINIMUM TEMPERATURE
# read in MinTemp data
setwd("C:/1megan/1research/KrigeUTM/MinT")
list.minT<-list.files(pattern=".shp$")
list.data<-list()
for (i in 1:length(list.minT))
{
        list.data[[i]]<-readShapePoints(list.minT[i])
}

# assign names to each file
n.d<-names(list.data)<-list.minT

#Krige
setwd("C:/1megan/1research/KrigPlots/MinT")
x=1
for (i in list.data)
{
        # kriging interp
```

```
        projection(i)<- "+proj=utm +zone=18 +south +ellps=WGS84
+towgs84=0,0,0,0,0,0,0 +units=m +no_defs"
        MinT_kr<- autoKrige(MAX_T~1, i, grid, model=c("Sph", "Exp", "Gau", "Ste"))

        #LOOCV
        MinT_cv<- krige.cv(MAX_T~1, i, model=MinT_kr$var_model)
        resid<-MinT_cv$residual
        z_score<-MinT_cv$zscore
        obs<-MinT_cv$observed

        # # Accuracy Assessment
        ME<-mean(resid)                 # Mean Error, ideal is 0
        MPSE<-mean(resid^2)                     # MPSE, ideally small
        MSNE<-mean(z_score^2)                   # Mean square normalized error, ideally
close to 1
        COP<-cor(obs, obs-resid)        # cor observed and predicted, ideally 1
        CPR<-cor(obs-resid, resid)      # cor predicted and residual, ideally 0
        write.table(data.frame(ME, MPSE, MSNE, COP,
CPR),file="C:/1megan/1research/AccuracyAssess/kr_MinT.csv", append=TRUE,
sep=",", row.names=FALSE, col.names=FALSE)

        # write output to raster
        kr.pred<-MinT_kr$krige_output
        r<-raster(kr.pred["var1.pred"])
        r.m<-mask(r, COraster)
        f<-paste0('kr_', n.d, '.tif')
        writeRaster(r.m, filename=f[x], overwrite=TRUE)
        x=x+1
}

# PRECIPITATION
# read in Precip data
setwd("C:/1megan/1research/KrigeUTM/Precip")
list.Precip<-list.files(pattern=".shp$")
list.data<-list()
for (i in 1:length(list.Precip))
{
        list.data[[i]]<-readShapePoints(list.Precip[i])
}

# assign names to each file
n.d<-names(list.data)<-list.Precip

#Krige
setwd("C:/1megan/1research/KrigPlots/Precip")
x=1
```

```
for (i in list.data)
{
        # kriging interp
        projection(i)<- "+proj=utm +zone=18 +south +ellps=WGS84
+towgs84=0,0,0,0,0,0,0 +units=m +no_defs"
        Precip_kr<- autoKrige(PRECIP~1, i, grid, model=c("Sph", "Exp", "Gau", "Ste"))

        #LOOCV
        Precip_cv<- krige.cv(PRECIP~1, i, model=Precip_kr$var_model)
        resid<-Precip_cv$residual
        z_score<-Precip_cv$zscore
        obs<-Precip_cv$observed

        # # Accuracy Assessment
        ME<-mean(resid)                 # Mean Error, ideal is 0
        MPSE<-mean(resid^2)                     # MPSE, ideally small
        MSNE<-mean(z_score^2)                   # Mean square normalized error, ideally
close to 1
        COP<-cor(obs, obs-resid)        # cor observed and predicted, ideally 1
        CPR<-cor(obs-resid, resid)      # cor predicted and residual, ideally 0
        write.table(data.frame(ME, MPSE, MSNE, COP,
CPR),file="C:/1megan/1research/AccuracyAssess/kr_Precip.csv", append=TRUE,
sep=",", row.names=FALSE, col.names=FALSE)

        # write output to raster
        kr.pred<-Precip_kr$krige_output
        r<-raster(kr.pred["var1.pred"])
        r.m<-mask(r, COraster)
        f<-paste0('kr_', n.d, '.tif')
        writeRaster(r.m, filename=f[x], overwrite=TRUE)
        x=x+1
}
```

Correlation Coefficients between Weather Variables and Dengue Incidence (Python)

This code was run for each of the weather variables and for the autroregressive dengue correlations at time lags of one to six months. Here the code for the computation of the correlation between dengue incidence and precipitation at a 6-month lag is presented as an example. It also includes how the optimal time lag was chosen for each predictor (where the file “AllWx.csv” is a combination of the correlations for each variable at each time lag).

```
import os
import glob
import pandas
import numpy as np

path_d = "C:/1megan/1research/XYZTables/Dengue3"
path_maxt = "C:/1megan/1research/XYZTables/MaxT3"
path_meant = "C:/1megan/1research/XYZTables/MeanT3"
path_mint = "C:/1megan/1research/XYZTables/MinT3"
path_p = "C:/1megan/1research/XYZTables/Precip3"

dengue = []
maxT = []
meanT = []
minT = []
precip = []
X=[]
Y=[]

# Read in data
# Dengue
os.chdir(path_d)
i = 0
for f in glob.glob("*.csv"):
    newMonth = []
    file = pandas.read_csv(f,delimiter=',', encoding="utf-8")
    # X and Y values (locations)
    if i == 0:
        for value in file.iloc[:,2]:
            X.append(value)
        for value in file.iloc[:,3]:
            Y.append(value)
```

```
    for value in file.iloc[:,4]:
        newMonth.append(value)
    dengue.append(newMonth)
    i = i + 1

dengue=np.asarray(dengue)
X = np.asarray(X)
Y = np.asarray(Y)

# maxT
os.chdir(path_maxt)
i = 0
for f in glob.glob("*.csv"):
    newMonth1 = []
    file1 = pandas.read_csv(f,delimiter=',', encoding="utf-8")
    for value in file1.iloc[:,4]:
        newMonth1.append(value)
    maxT.append(newMonth1)
    i = i + 1
maxT=np.asarray(maxT)

# meanT
os.chdir(path_meant)
i = 0
for f in glob.glob("*.csv"):
    newMonth2 = []
    file2 = pandas.read_csv(f,delimiter=',', encoding="utf-8")
    for value in file2.iloc[:,4]:
        newMonth2.append(value)
    meanT.append(newMonth2)
    i = i + 1
meanT=np.asarray(meanT)

# minT
os.chdir(path_mint)
i = 0
for f in glob.glob("*.csv"):
    newMonth3 = []
    file3 = pandas.read_csv(f,delimiter=',', encoding="utf-8")
    for value in file3.iloc[:,4]:
        newMonth3.append(value)
    minT.append(newMonth3)
    i = i + 1
minT=np.asarray(minT)

# precip
```

```
os.chdir(path_p)
i = 0
for f in glob.glob("*.csv"):
   newMonth4 = []
   file4 = pandas.read_csv(f,delimiter=',', encoding="utf-8")
   for value in file4.iloc[:,4]:
      newMonth4.append(value)
   precip.append(newMonth4)
   i = i + 1
precip=np.asarray(precip)


# Compute Correlation Coefficients
# Precipitation at a 6 month lag

dlist=[]
tlist=[]

for pixel in range(26360):
   newpixelD = []
   newpixelT = []
   for i, j in zip(dengue[6:95,pixel], precip[0:89,pixel]):
      newpixelD.append(i)
      newpixelT.append(j)
   dlist.append(newpixelD)
   tlist.append(newpixelT)

dlist=np.asarray(dlist)
tlist=np.asarray(tlist)
cclist=[]
for i in range(26360):
   cc = np.corrcoef(dlist[i,:],tlist[i,:])
   cclist.append(cc)

Precip_6=open("C:/1megan/1research/Correlations/Tables/Precip_6.csv", "w")

for i in cclist:
   cor=i[0,1]
   print(cor, file=Precip_1)

Precip_6.close()

# Pick optimal time lags

f = "C:/1megan/1research/Correlations/Tables/AllWx.csv"
file = pandas.read_csv(f,delimiter=',', encoding="utf-8")
```

```
allwx = np.asarray(file)

try1 = open("C:/1megan/1research/Correlations/Tables/optLagDengue.csv", "w")

new_Pixel=[]
new_index = []
for i in allwx:
    maxT = i[2:8]
    meanT = i[8:14]
    minT = i[14:20]
    Precip = i[20:26]
    Dengue = i[26:32]

    max_maxT = max(maxT)
    max_meanT = max(meanT)
    max_minT = max(minT)
    max_Precip = max(Precip)
    max_Dengue = max(Dengue)

    min_maxT = min(maxT)
    min_meanT = min(meanT)
    min_minT = min(minT)
    min_Precip = min(Precip)
    min_Dengue = min(Dengue)

    a_maxT = abs(min_maxT)
    a_meanT = abs(min_meanT)
    a_minT = abs(min_minT)
    a_Precip = abs(min_Precip)
    a_Dengue = abs(Dengue)

    new_Pixel.append(max_Dengue)

    ii = np.where(Dengue == max_Dengue)
    string = str(ii)
    num = string[8]
    new_index.append(num)

for x,y in zip(new_Pixel, new_index):
    print(x, ",", y, file = try1)

try1.close()
```

Code for Choosing the Optimal Predictors for Each Pixel (Python)

```
import pandas
#from scipy import stats
import statsmodels.formula.api as sm
#import os
#import glob
import numpy as np

# read in data and create arrays
lag_data = "C:/1megan/1research/Correlations/Tables/LAG.csv"
lagData = pandas.read_csv(lag_data, delimiter=',', encoding="utf-8" )
lag = np.asarray(lagData)
maxT_optLag = lag[:,0]
meanT_optLag = lag[:,1]
minT_optLag = lag[:,2]
precip_optLag = lag[:,3]

dengue_data = "C:/1megan/1research/XYZTables/dengueArray.csv"
dengueData = pandas.read_csv(dengue_data, delimiter=',', encoding="utf-8" )
dengue = np.asarray(dengueData)

maxT_data = "C:/1megan/1research/XYZTables/maxTArray.csv"
maxTData = pandas.read_csv(maxT_data, delimiter=',', encoding="utf-8" )
maxT = np.asarray(maxTData)

meanT_data = "C:/1megan/1research/XYZTables/meanTArray.csv"
meanTData = pandas.read_csv(meanT_data, delimiter=',', encoding="utf-8" )
meanT = np.asarray(meanTData)

minT_data = "C:/1megan/1research/XYZTables/minTArray.csv"
minTData = pandas.read_csv(minT_data, delimiter=',', encoding="utf-8" )
minT = np.asarray(minTData)

Precip_data = "C:/1megan/1research/XYZTables/PrecipArray.csv"
PrecipData = pandas.read_csv(Precip_data, delimiter=',', encoding="utf-8" )
precip = np.asarray(PrecipData)

new_file = open("C:/1megan/1research/FinalModels/PythonOutput/nah.csv", "w")

print("Model,", "Rsquared,","AIC,", "MSE,", "Bint,", "Bdengue,", "BmaxT,",
    "BmeanT,", "BminT,", "Bprecip,", "Pint,", "Pdengue,", "PmaxT,", "PmeanT,"
    , "PminT,", "Pprecip", file = new_file)

for pixel in range(5):
    num_max = maxT_optLag[pixel]
```

```
num_mean = meanT_optLag[pixel]
num_min = minT_optLag[pixel]
num_precip = precip_optLag[pixel]
denguestr = dengue[pixel, 6:95]

if num_max == 1:
    maxTstr = maxT[pixel, 5:94]
if num_max == 2:
    maxTstr = maxT[pixel, 4:93]
if num_max == 3:
    maxTstr = maxT[pixel, 3:92]
if num_max == 4:
    maxTstr = maxT[pixel, 2:91]
if num_max == 5:
    maxTstr = maxT[pixel, 1:90]
if num_max == 6:
    maxTstr = maxT[pixel, 0:89]

if num_mean == 1:
    meanTstr = meanT[pixel, 5:94]
if num_mean == 2:
    meanTstr = meanT[pixel, 4:93]
if num_mean == 3:
    meanTstr = meanT[pixel, 3:92]
if num_mean == 4:
    meanTstr = meanT[pixel, 2:91]
if num_mean == 5:
    meanTstr = meanT[pixel, 1:90]
if num_mean == 6:
    meanTstr = meanT[pixel, 0:89]

if num_min == 1:
    minTstr = minT[pixel, 5:94]
if num_min == 2:
    minTstr = minT[pixel, 4:93]
if num_min == 3:
    minTstr = minT[pixel, 3:92]
if num_min == 4:
    minTstr = minT[pixel, 2:91]
if num_min == 5:
    minTstr = minT[pixel, 1:90]
if num_min == 6:
    minTstr = minT[pixel, 0:89]

if num_precip == 1:
    precipstr = precip[pixel, 5:94]
```

```
if num_precip == 2:
    precipstr = precip[pixel, 4:93]
if num_precip == 3:
    precipstr = precip[pixel, 3:92]
if num_precip == 4:
    precipstr = precip[pixel, 2:91]
if num_precip == 5:
    precipstr = precip[pixel, 1:90]
if num_precip == 6:
    precipstr = precip[pixel, 0:89]

#dengue optimal lag always one month
dengue_lag = dengue[pixel, 5:94]

df = pandas.DataFrame({"Dengue": denguestr, "dengueLag": dengue_lag,
                       "maxTdf": maxTstr, "meanTdf": meanTstr, "minTdf"
                       : minTstr, "precipdf": precipstr})
aa = "Dengue ~"
aaa = "dengueLag"
ab, ac, ad, ae, af = "dengueLag+","maxTdf","meanTdf","minTdf","precipdf"

D0 = sm.ols(formula = aa+aaa, data = df).fit()
D1 = sm.ols(formula = aa+ab+ac, data = df).fit()
D2 = sm.ols(formula = aa+ab+ad, data = df).fit()
D3 = sm.ols(formula = aa+ab+ae, data = df).fit()
D4 = sm.ols(formula = aa+ab+af, data = df).fit()
D5 = sm.ols(formula = aa+ab+ac+"+"+ad, data = df).fit()
D6 = sm.ols(formula = aa+ab+ac+"+"+ae, data = df).fit()
D7 = sm.ols(formula = aa+ab+ac+"+"+af, data = df).fit()
D8 = sm.ols(formula = aa+ab+ad+"+"+ae, data = df).fit()
D9 = sm.ols(formula = aa+ab+ad+"+"+af, data = df).fit()
D10 = sm.ols(formula = aa+ab+ae+"+"+af, data = df).fit()
D11 = sm.ols(formula = aa+ab+ac+"+"+ad+"+"+ae, data = df).fit()
D12 = sm.ols(formula = aa+ab+ac+"+"+ad+"+"+af, data = df).fit()
D13 = sm.ols(formula = aa+ab+ac+"+"+ae+"+"+af, data = df).fit()
D14 = sm.ols(formula = aa+ab+ad+"+"+ae+"+"+af, data = df).fit()
D15 = sm.ols(formula = aa+ab+ac+"+"+ad+"+"+ae+"+"+af, data = df).fit()


model_list = [D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13,
              D14, D15]
r_list = [D0.rsquared, D1.rsquared, D2.rsquared, D3.rsquared, D4.rsquared,
          D5.rsquared, D6.rsquared, D7.rsquared, D8.rsquared, D9.rsquared,
          D10.rsquared, D11.rsquared, D12.rsquared, D13.rsquared,
          D14.rsquared, D15.rsquared]
```

```
m1 = max(r_list[1:5])
ind1 = r_list.index(m1)
m2 = max(r_list[5:12])
ind2 = r_list.index(m2)
m3 = max(r_list[12:16])
ind3 = r_list.index(m3)

r_diff = m1-r_list[0]
r_perc = (r_diff / r_list[0]) * 100
model1 = model_list[ind1]
if r_perc > 1 and model1.pvalues[2] < 0.1:
   model = model1
   r_diff2 = m2-m1
   r_perc2 = (r_diff2 / m1) * 100
   model2 = model_list[ind2]

   if r_perc2 > 1 and model2.pvalues[2] < 0.1 and model2.pvalues[3] < 0.1:
      model = model2
      r_diff3 = m3-m2
      r_perc3 = (r_diff3 / m2) * 100
      model3 = model_list[ind3]

      if (r_perc3 > 1 and model3.pvalues[2] < 0.1 and model3.pvalues[3]
      < 0.1 and model3.pvalues[4] < 0.1):
         model = model3
         r_diff4 = D15.rsquared - m3
         r_perc4 = (r_diff4 / m3) * 100

         if (r_perc4 > 1 and D15.pvalues[2] < 0.1 and D15.pvalues[3] <
         0.1 and D15.pvalues[4] < 0.1 and D15.pvalues[5] < 0.1):
            model = D15
            #print("model4,", model.pvalues[2], ",", model.pvalues[3],
             #     ",", model.pvalues[4], ",", model.pvalues[5], file =
              #    new_file)
         else:
            model = model3
            #print("model3,", model.pvalues[2], ",", model.pvalues[3],
             #     ",", model.pvalues[4], file = new_file)
      else:
         model = model2
         #print("model2,", model.pvalues[2], ",", model.pvalues[3],
          #     file = new_file)
   else:
      model = model1
      #print("model1,", model.pvalues[2], file = new_file)
else:
```

```
        model = model_list[0]
        #print("model0", file = new_file)

    ind = model_list.index(model)

    if ind == 0:
        print(ind, ",", model.rsquared, ",", model.aic, ",", model.mse_model,
              ",", model.params[0], ",", model.params[1], ",", "-9999.9999,",
              "-9999.9999,", "-9999.9999,", "-9999.9999,", model.pvalues[0],
              ",", model.pvalues[1], ",-9999.9999,", "-9999.9999,",
              "-9999.9999,", "-9999.9999", file = new_file)
    if ind == 1:
        print(ind, ",", model.rsquared, ",", model.aic, ",", model.mse_model,
              ",", model.params[0], ",", model.params[1], ",", model.params[2],
              ",-9999.9999,", "-9999.9999,", "-9999.9999,", model.pvalues[0],
              ",", model.pvalues[1], ",", model.pvalues[2], ",-9999.9999,",
              "-9999.9999,", "-9999.9999", file = new_file)
    if ind == 2:
        print(ind, ",", model.rsquared, ",", model.aic, ",", model.mse_model,
              ",", model.params[0], ",", model.params[1], ",", "-9999.9999,",model.params[2],
              ",-9999.9999,", "-9999.9999,", model.pvalues[0], ",", model.pvalues[1],
               ",-9999.9999,", model.pvalues[2], ",-9999.9999,", "-9999.9999",
              file = new_file)
    if ind == 3:
        print(ind, ",", model.rsquared, ",", model.aic, ",", model.mse_model,
              ",", model.params[0], ",", model.params[1], ",", "-9999.9999,", "-9999.9999,",
              model.params[2], ",-9999.9999,", model.pvalues[0], ",", model.pvalues[1],
              ",", "-9999.9999,", "-9999.9999,", model.pvalues[2], ",-9999.9999",
              file = new_file)
    if ind == 4:
        print(ind, ",", model.rsquared, ",", model.aic, ",", model.mse_model,
              ",", model.params[0], ",", model.params[1], ",", "-9999.9999,", "-9999.9999,",
              "-9999.9999,", model.params[2], ",", model.pvalues[0], ",", model.pvalues[1],
              ",", "-9999.9999,", "-9999.9999,", "-9999.9999,", model.pvalues[2],
              file = new_file)
    if ind == 5:
        print(ind, ",", model.rsquared, ",", model.aic, ",", model.mse_model,
              ",", model.params[0], ",", model.params [1], ",", model.params[2], ",",
model.params[3],
              ",-9999.9999,", "-9999.9999,", model.pvalues[0], ",", model.pvalues[1],
              ",", model.pvalues[2], ",", model.pvalues[3], ",-9999.9999,", "-9999.9999",
              file = new_file)
    if ind == 6:
        print(ind, ",", model.rsquared, ",", model.aic, ",", model.mse_model,
              ",", model.params[0], ",", model.params [1], ",", model.params[2], ",-
9999.9999,",
```

```
              model.params[3], ",-9999.9999,", model.pvalues[0], ",", model.pvalues[1],
              ",", model.pvalues[2], ",-9999.9999,", model.pvalues[3], ",-9999.9999",
              file = new_file)
    if ind == 7:
        print(ind, ",", model.rsquared, ",", model.aic, ",", model.mse_model,
              ",", model.params[0], ",", model.params [1], ",", model.params[2], ",-
9999.9999,",
              "-9999.9999,", model.params[3], ",", model.pvalues[0], ",", model.pvalues[1],
              ",", model.pvalues[2], ",-9999.9999,", "-9999.9999,", model.pvalues[3],
              file = new_file)
    if ind == 8:
        print(ind, ",", model.rsquared, ",", model.aic, ",", model.mse_model,
              ",", model.params[0], ",", model.params [1], ",-9999.9999,", model.params[2],
              ",", model.params[3], ",-9999.9999,", model.pvalues[0], ",", model.pvalues[1],
              ",-9999.9999,", model.pvalues[2], ",", model.pvalues[3],",-9999.9999",
              file = new_file)
    if ind == 9:
        print(ind, ",", model.rsquared, ",", model.aic, ",", model.mse_model,
              ",", model.params[0], ",", model.params [1], ",-9999.9999,", model.params[2],
              ",-9999.9999,", model.params[3], ",", model.pvalues[0], ",", model.pvalues[1],
              ",-9999.9999,", model.pvalues[2], ",-9999.9999,", model.pvalues[3],
              file = new_file)
    if ind == 10:
        print(ind, ",", model.rsquared, ",", model.aic, ",", model.mse_model,
              ",", model.params[0], ",", model.params [1], ",-9999.9999,", "-9999.9999,",
              model.params[2], ",", model.params[3], ",", model.pvalues[0], ",",
model.pvalues[1],
              ",-9999.9999,", "-9999.9999,", model.pvalues[2], ",", model.pvalues[3],
              file = new_file)
    if ind == 11:
        print(ind, ",", model.rsquared, ",", model.aic, ",", model.mse_model,
              ",", model.params[0], ",", model.params [1], ",", model.params[2], ",",
model.params[3],
              ",",model.params[4], ",-9999.9999,", model.pvalues[0], ",", model.pvalues[1],
              ",", model.pvalues[2], ",", model.pvalues[3], ",", model.pvalues[4],
              ",-9999.9999", file = new_file)
    if ind == 12:
        print(ind, ",", model.rsquared, ",", model.aic, ",", model.mse_model,
              ",", model.params[0], ",", model.params [1], ",", model.params[2], ",",
model.params[3],
              ",-9999.9999,", model.params[4], ",", model.pvalues[0], ",", model.pvalues[1],
              ",", model.pvalues[2], ",", model.pvalues[3], ",-9999.9999,", model.pvalues[4],
              file = new_file)
    if ind == 13:
        print(ind, ",", model.rsquared, ",", model.aic, ",", model.mse_model,
```

```
             ",", model.params[0], ",", model.params [1], ",", model.params[2], ",-
9999.9999,",
             model.params[3], ",", model.params[4], ",", model.pvalues[0], ",",
model.pvalues[1],
             ",", model.pvalues[2], ",-9999.9999,", model.pvalues[3], ",", model.pvalues[4],
             file = new_file)
    if ind == 14:
        print(ind, ",", model.rsquared, ",", model.aic, ",", model.mse_model,
             ",", model.params[0], ",", model.params [1], ",-9999.9999,", model.params[2],
             ",", model.params[3], ",", model.params[4], ",", model.pvalues[0], ",",
model.pvalues[1],
             ",-9999.9999,", model.pvalues[2], ",", model.pvalues[3],",", model.pvalues[4],
             file = new_file)
    if ind == 15:
        print(ind, ",", model.rsquared, ",", model.aic, ",", model.mse_model,
             ",", model.params[0], ",", model.params [1], ",", model.params[2], ",",
model.params[3],
             ",", model.params[4], ",", model.params[5], ",", model.pvalues[0], ",",
model.pvalues[1],
             ",", model.pvalues[2], ",", model.pvalues[3], ",", model.pvalues[4],
             ",", model.pvalues[5], file = new_file)


new_file.close()
```

Prediction Model Code (Python)

```
import pandas
import numpy as np

# read in data and create arrays
lag_data = "C:/1megan/1research/Correlations/Tables/LAG.csv"
lagData = pandas.read_csv(lag_data, delimiter=',', encoding="utf-8" )
lag = np.asarray(lagData)
maxT_optLag = lag[:,0]
meanT_optLag = lag[:,1]
minT_optLag = lag[:,2]
precip_optLag = lag[:,3]

dengue_data = "C:/1megan/1research/XYZTables/dengueArray.csv"
dengueData = pandas.read_csv(dengue_data, delimiter=',', encoding="utf-8" )
dengue = np.asarray(dengueData)

maxT_data = "C:/1megan/1research/XYZTables/maxTArray.csv"
maxTData = pandas.read_csv(maxT_data, delimiter=',', encoding="utf-8" )
maxT = np.asarray(maxTData)

meanT_data = "C:/1megan/1research/XYZTables/meanTArray.csv"
meanTData = pandas.read_csv(meanT_data, delimiter=',', encoding="utf-8" )
meanT = np.asarray(meanTData)

minT_data = "C:/1megan/1research/XYZTables/minTArray.csv"
minTData = pandas.read_csv(minT_data, delimiter=',', encoding="utf-8" )
minT = np.asarray(minTData)

Precip_data = "C:/1megan/1research/XYZTables/PrecipArray.csv"
PrecipData = pandas.read_csv(Precip_data, delimiter=',', encoding="utf-8")
precip = np.asarray(PrecipData)

wholeThing = "C:/1megan/1research/Correlations/Final/cats.csv"
wT = pandas.read_csv(wholeThing, delimiter=',', encoding="utf-8")
param = np.asarray(wT)

file2 = open("C:/1megan/1research/FinalModels/PythonOutput/ModelOutput.csv", "wb")

d_pred_list = range(96,132)

d_arrays = np.zeros(36)

# for each pixel for each month during the prediction period, set which column
# of the array represents each paramater
```

```
for pixel in range(26359):

    d_pred_mos = dengue[pixel,96:132]
    d_lag_mos = dengue[pixel,95:131]
    maxT_pred = maxT[pixel,:]
    meanT_pred = meanT[pixel,:]
    minT_pred = minT[pixel,:]
    precip_pred = precip[pixel,:]

    model_num = param[pixel,0]
    Bint = param[pixel,4]
    Bdengue = param[pixel,5]
    BmaxT = param[pixel,6]
    BmeanT = param[pixel,7]
    BminT = param[pixel,8]
    Bprecip = param[pixel,9]
    Pint = param[pixel,10]
    Pint = float(Pint)

    num_max = maxT_optLag[pixel]
    num_mean = meanT_optLag[pixel]
    num_min = minT_optLag[pixel]
    num_precip = precip_optLag[pixel]

# Set each model with the predictors associated with that model

    if model_num == 0:
        d_list = []
        for mo, lagmo in zip(d_pred_mos.T, d_lag_mos.T):
            if Pint < 0.01:
                d_pred = Bint + (Bdengue * lagmo)
            else:
                d_pred = Bdengue * lagmo
            d_list.append(d_pred)
        d_array = np.array(d_list)

    if model_num == 1:
        d_list = []
        for a, mo, lagmo in zip(d_pred_list, d_pred_mos.T, d_lag_mos.T):
            maxTlag = a - num_max
            if Pint < 0.01:
                d_pred = Bint + (Bdengue * lagmo) + (BmaxT * maxT_pred[maxTlag])
            else:
                d_pred = (Bdengue * lagmo) + (BmaxT * maxT_pred[maxTlag])
            d_list.append(d_pred)
```

```
    d_array = np.array(d_list)

if model_num == 2:
    d_list = []
    for a, mo, lagmo in zip(d_pred_list, d_pred_mos.T, d_lag_mos.T):
        meanTlag = a - num_mean
        if Pint < 0.01:
            d_pred = Bint + (Bdengue * lagmo) + (BmeanT * meanT_pred[meanTlag])
        else:
            d_pred = (Bdengue * lagmo) + (BmeanT * meanT_pred[meanTlag])
        d_list.append(d_pred)
    d_array = np.array(d_list)

if model_num == 3:
    d_list = []
    for a, mo, lagmo in zip(d_pred_list, d_pred_mos.T, d_lag_mos.T):
        minTlag = a - num_min
        if Pint < 0.01:
            d_pred = Bint + (Bdengue * lagmo) + (BminT * minT_pred[minTlag])
        else:
            d_pred = (Bdengue * lagmo) + (BminT * minT_pred[minTlag])
        d_list.append(d_pred)
    d_array = np.array(d_list)

if model_num == 4:
    d_list = []
    for a, mo, lagmo in zip(d_pred_list, d_pred_mos.T, d_lag_mos.T):
        preciplag = a - num_precip
        if Pint < 0.01:
            d_pred = Bint + (Bdengue * lagmo) + (BminT * precip_pred[preciplag])
        else:
            d_pred = (Bdengue * lagmo) + (Bprecip * precip_pred[preciplag])
        d_list.append(d_pred)
    d_array = np.array(d_list)

if model_num == 7:
    d_list = []
    for a, mo, lagmo in zip(d_pred_list, d_pred_mos.T, d_lag_mos.T):
        maxTlag = a - num_max
        preciplag = a - num_precip
        if Pint < 0.01:
            d_pred = Bint + (Bdengue * lagmo) + (BmaxT * maxT_pred[maxTlag])
            + (BminT * precip_pred[preciplag])
        else:
            d_pred = (Bdengue * lagmo) + (BmaxT * maxT_pred[maxTlag])
            + (Bprecip * precip_pred[preciplag])
```

```
        d_list.append(d_pred)
    d_array = np.array(d_list)

if model_num == 9:
    d_list = []
    for a, mo, lagmo in zip(d_pred_list, d_pred_mos.T, d_lag_mos.T):
        meanTlag = a - num_mean
        preciplag = a - num_precip
        if Pint < 0.01:
            d_pred = Bint + (Bdengue * lagmo) + (BmeanT * meanT_pred[meanTlag])
            + (BminT * precip_pred[preciplag])
        else:
            d_pred = (Bdengue * lagmo) + (BmeanT * meanT_pred[meanTlag])
            + (Bprecip * precip_pred[preciplag])
        d_list.append(d_pred)
    d_array = np.array(d_list)

if model_num == 10:
    d_list = []
    for a, mo, lagmo in zip(d_pred_list, d_pred_mos.T, d_lag_mos.T):
        minTlag = a - num_min
        preciplag = a - num_precip
        if Pint < 0.01:
            d_pred = Bint + (Bdengue * lagmo) + (BminT * minT_pred[minTlag])
            + (BminT * precip_pred[preciplag])
        else:
            d_pred = (Bdengue * lagmo) + (BminT * minT_pred[minTlag])
            + (Bprecip * precip_pred[preciplag])
        d_list.append(d_pred)
    d_array = np.array(d_list)

if model_num == 11:
    d_list = []
    for a, mo, lagmo in zip(d_pred_list, d_pred_mos.T, d_lag_mos.T):
        maxTlag = a - num_max
        meanTlag = a - num_mean
        minTlag = a - num_min
        if Pint < 0.01:
            d_pred = Bint + (Bdengue * lagmo) + (BmaxT * maxT_pred[maxTlag])
            + (BmeanT * meanT_pred[meanTlag]) + (BminT * minT_pred[minTlag])
        else:
            d_pred = (Bdengue * lagmo) + (BmaxT * maxT_pred[maxTlag])
            + (BmeanT * meanT_pred[meanTlag]) + (BminT * minT_pred[minTlag])
        d_list.append(d_pred)
    d_array = np.array(d_list)
```

```
    if model_num == 15:
        d_list = []
        for a, mo, lagmo in zip(d_pred_list, d_pred_mos.T, d_lag_mos.T):
            maxTlag = a - num_max
            meanTlag = a - num_mean
            minTlag = a - num_min
            preciplag = a - num_precip
            if Pint < 0.01:
                d_pred = Bint + (Bdengue * lagmo) + (BmaxT * maxT_pred[maxTlag])
                + (BmeanT * meanT_pred[meanTlag]) + (BminT * minT_pred[minTlag])
                + (Bprecip * precip_pred[preciplag])
            else:
                d_pred = (Bdengue * lagmo) + (BmaxT * maxT_pred[maxTlag])
                + (BmeanT * meanT_pred[meanTlag]) + (BminT * minT_pred[minTlag])
                + (Bprecip * precip_pred[preciplag])
            d_list.append(d_pred)
        d_array = np.array(d_list)

    d_arrays = np.vstack((d_arrays, d_array))

# output results to a file that has the predicted dengue count at each pixel
# for every month of the prediction period

np.savetxt(file2, d_arrays, delimiter = ',')

file2.close()
```