Claim Assessment using Predictive Models for the Citizen’s Observatory in Bogota (Veeduria Distrital de Bogota - Colombia)

By

Armando Saavedra

A Capstone Project Submitted to

Dr. Hong Cui

Associated Professor

School of Information

University of Arizona

May 2020

In Partial Fulfillment of the Requirement for the Degree of

Master of Science in

Information

Acknowledge

I just want to say thank you to my supervisor Dr. Hong for her patience and sharing her knowledge during the last two years. I also would like to thanks to Dr. Bethard for the neural network lectures and advice that he has given me. This project would not have been possible without the love and support of my family. Thank you to my sister for her assistance in getting me connected with the Citizen’s Observatory in Bogota and a special thank you to my wife and daughter who sacrificed family time and helped keep me motivated to make it to the end!

Introduction

One of the main problems that public and private institutions face in improving goodwill or public opinion is how quickly they respond to the petitions issued by customers or users. This is one of the main issues of the Citizen’s Observatory; its office receives thousands of petitions from different sources and responding in a timely manner has a big impact on the perceived efficiency of the office and its budget. For this reason, creating a predictive model measuring the level of response to claims, may assist in determining patterns that allow the Citizen’s Observatory to focus its attention on more relevant issues and give better service.

Before continuing, it is helpful to explain that the Citizen’s Observatory (Veeduria Distrital de Bogota) is a public institution which receives claims related to the supervision of the district administration in order to exert preventive control, promote social control, strength integrity and fight corruption to bolster district public management. The Citizen’s Observatory, like all public and private institutions and organizations, has a time limit for answering claims from customers or users of their services. If an organization passes a legal deadline in answering a claim, that organization could be subject to legal sanctions. These deadlines are also used as a benchmark in determining the efficiency of an organization and ultimately determining budget allocations.

Having that in mind, the goal of his capstone project is to create two comparative machine learning models to predict when the Citizen’s Observatory solved or did not solve a claim given.

Dataset

The information used for this analysis was provided by the Citizen’s Observatory through a personal request made by a data analyst from the University of Arizona, Armando Saavedra. The database weighs almost 400 Mb; it is a csv format file with 1,110,037 observations and 18 variables. Given the origin of the database, the information is labeled in Spanish.

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Data Type** | **Description** |
| ID: | Integer | It is a variable that denotes the identification of a specific claim. |
| Peticion | String | Type of petition. This could be public management, mobility and so on. |
| Sector | String | Type of organization or institution managed by the government. |
| Entidad | String | Specific corporation or public institution. |
| Dependencia | String | Dependency or division of a corporation or public institution. |
| Tema | String | Topic or issue that the petition is about. |
| Subtema | String | Specific topic. |
| Localidad | String | Sub-region defining how a city is geographically divided. |
| Canal | String | Means by which people can have access to the information that the petition is about. |
| TipoPeticion | String | Type of petition. Is it informing us of something, it is requiring information about something, and so on? |
| EstadoInicial | String | First stage of the petition. |
| EstadoFinal | String | Final stage of the petition. |
| FechaIngreso | Datetime | Date when the petition was made. |
| FechaVencimiento | Datetime | Expiration date of the petition. |
| FechaFinalizacion | Datetime | Date when the petition was forwarded to another entity. |
| FechaCierre | Datetime | Date when the petition was solved |
| DiasGestion | Datetime | How long it took to find a solution to the petition. |
| DiasVencimiento | Datetime | Range of time between expiration date and the date when the problem is solved. |

Methods

The implementation of this project began by importing the database using R. Next, the quality of the data was verified looking for inconsistencies that could generate problems in the model implementation. Finally, two machine learning algorithms were created to predict the office’s efficiency in solving claims.

* 1. Data Cleaning, Data Transformation

There were multiple problems importing the data related to language. In order to fix these, the data was imported using encoded latin1. Once those problems were resolved, the traditional method of importing cvs files with R was used, leaving the headers as they came in their original data sets.

> data<-read.csv('dataset.csv',header = T)

Next, implementing the data cleaning process required the following actions:

The column “Peticion*”* was deleted since it counted every claim differently by assigning it a unique label. To do this the next piece of code shows how the “Peticion” variable is deleted in R.

> data$Peticion<-NULL

Most variables had similar problems: digits were interspersed among words, upper-case letters and special symbols were included throughout, etc. Another common problem was that there were multiple labels for the same class in every variable.

To demonstrate the cleaning process, the variable “Sector” was chosen because it had many of the issues mentioned above and provides a good snapshot of the coding that was applied, thus avoiding the need to repeat the same implementation for each of the variables.

A special library called “stringr” provided functions which organized the categorical data: replacing, trimming, and changing the wrong data format when necessary.

The function “str\_replace\_all()” offered a good approach in solving multiple problems. It removed all punctuation using the [:punct:] argument and digits using the [:digit:] argument. Next, all labels were transformed to all caps using the function “str\_to\_title”. Empty spaces are common in this sort of variables, so the function “str\_trim()” help to removed blank spaces from the right or left side.

> library(stringr)

> data$Sector<-str\_replace\_all(data$Sector,'[:punct:]|[:digit:]','')

> data$Sector<-str\_to\_title(data$Sector)

> data$Sector<-str\_trim(data$Sector,side ='both')

As mentioned before, mislabeled data was another characteristic of this dataset. Two different strategies were implemented to address this problem. First, common words from multiple labels that belong to the same type of data were indexed. Later, they were rewritten using the index that was created.

When it was difficult to find common labels or there were too few, the function “str\_replace\_all()” was applied in which the first argument is the variable to modify, the second is the mislabeled variable and the third is the fixed label.

> SectorEd<-str\_detect(data$Sector,'Gestion Juridica')

> data$Sector[SectorEd]<-'Gestion Juridica'

> data$Sector<-str\_replace\_all(data$Sector,

'Seguridad Convivencia Y Justicia',

'Seguridad Convivencia Y Justicia')

In order to see a synopsis of the changes made to the dataset, the function “summarize()” offers a good overview. Here, the data is shown in a better format and the variables are easier to describe.

> summary(data)

Sector Entidad

Length:520110 Secretaria De Gobierno : 68768

Class :character Secretaria General : 63160

Mode :character Universidad Distrital : 52124

Secretaria De Integracion Social: 34315

Secretaria De Salud : 31868

Secretaria Movilidad : 30489

(Other) :239386

Dependencia

Oficina De Atencion A La Ciudadania : 65575

Oficina De Atencion Al Ciudadano : 32602

Servicio Al Ciudadano : 26057

Usuarios Asesores Linea : 25418

Direccion De Servicio Al Ciudadano : 21451

Central De Peticiones Distrito Capital: 12635

(Other) :336372

Tema

Sin Dato : 69587

Gobierno Local : 63934

Salud : 58146

Educacion : 50625

Movilidad Transporte Malla Vial: 43514

Funcion Publica Administracion : 35537

(Other) :198767

Subtema Localidad

Sin Dato : 94223 Sin Dato :436010

Admision De Proyectos De Pregrado Y Posgrado: 27840 Suba : 8707

Traslado Por No Competencia : 27405 Teusaquillo : 7020

Atencion Y Servicio A La Ciudadania : 24372 Kennedy : 6721

Traslado A Entidades Distritales : 18151 Ciudad Boliva: 6438

Sistema De Corresponencia Y Radicacion : 15012 Engativa : 6282

(Other) :313107 (Other) : 48932

Canal TipoPeticion

Buzon : 38621 Derecho De Peticion De Interes Particular:169407

E-Mail : 65975 Derecho De Peticion De Interes General :115768

Escrito :192657 Reclamo : 63211

Presencial: 71460 Solicitud De Acceso A La Informacion : 46322

Telefono : 44168 Queja : 44648

Web :107229 Solicitud De Informacion : 36884

(Other) : 43870

EstadoInicial

Registro Con Preclasificacion:390675

Registro Para Asignacion : 92449

Registro Sin Preclasificacion: 20942

En Tramite Por Asignacion : 11562

En Tramite Por Traslado : 1797

Registro Para Traslado : 1632

(Other) : 1053

EstadoFinal FechaIngreso

Solucionado Por Asignacion :237117 Min. :2016-01-01

Solucionado Por Respuesta Definitiva :115241 1st Qu.:2017-03-14

Solucionado Registro Con Preclasificacion: 92449 Median :2018-02-02

Solucionado Por Traslado : 64491 Mean :2018-02-08

En Tramite Respuesta Preparada : 2545 3rd Qu.:2019-04-01

Por Ampliar Por Solicitud Ampliacion : 1953 Max. :2019-11-29

(Other) : 6314

FechaVencimiento FechaFinalizacion FechaCierre

Min. :2016-01-04 Min. :2016-01-01 Min. :1900-01-01

1st Qu:2017-03-15 1st Qu:2017-03-15 1st Qu:2016-11-02

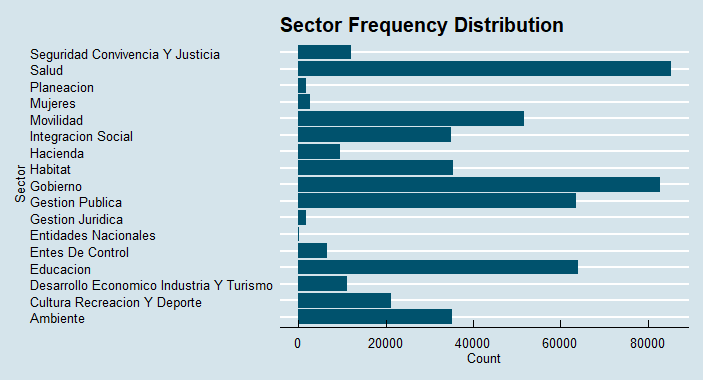
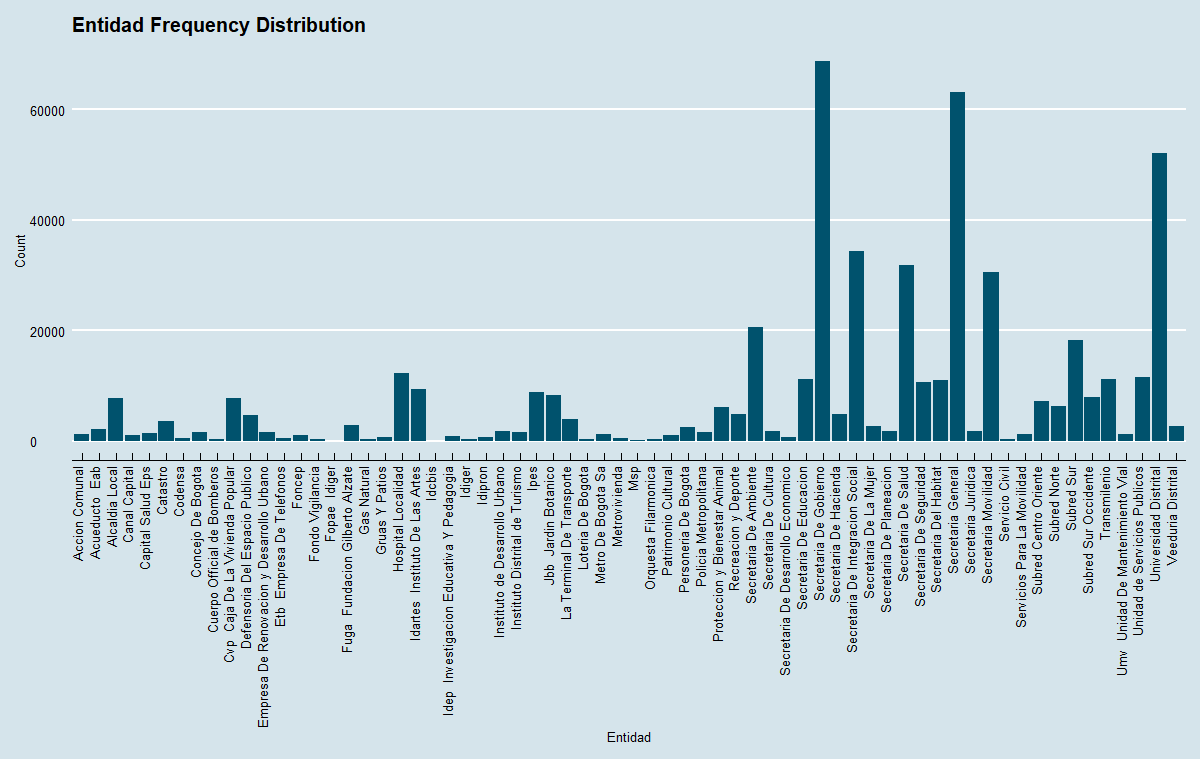
Median:2018-02-05 Median:2018-02-02 Median :2017-09-21

Mean :2018-02-14 Mean :2018-02-09 Mean :2006-10-31

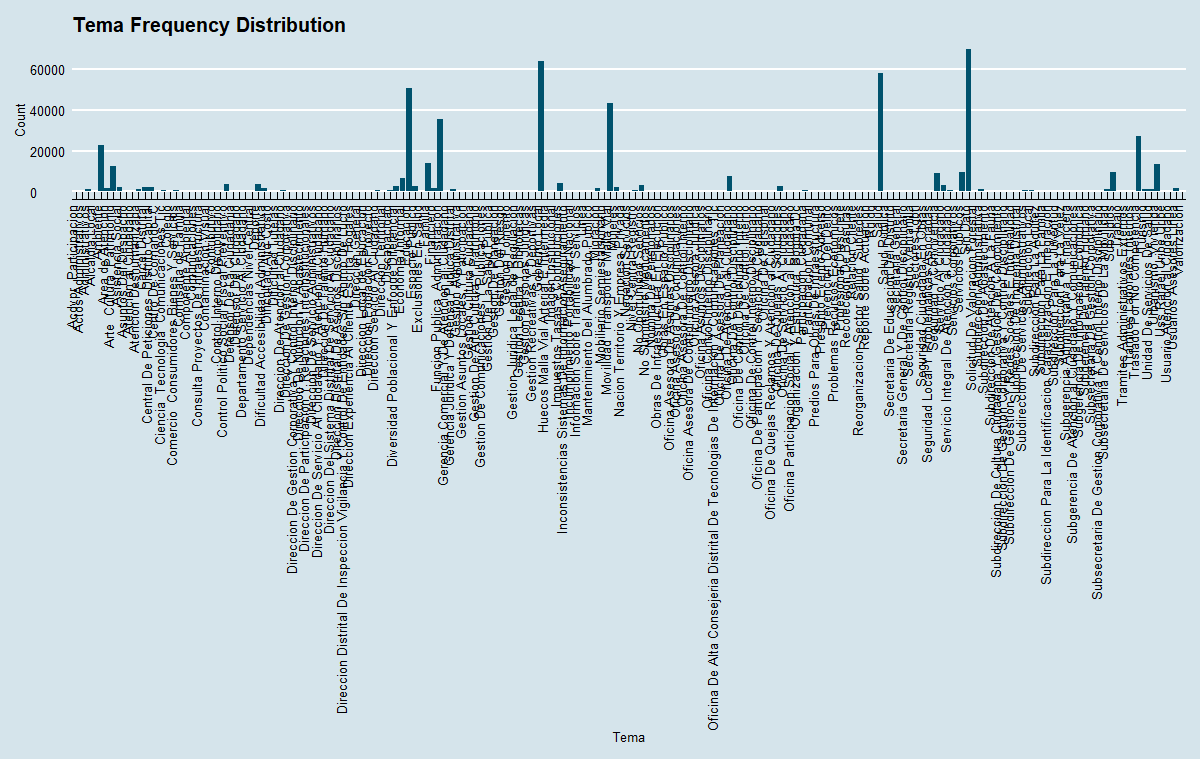
3rd Qu.:2019-04-02 3rd Qu.:2019-04-02 3rd Qu.:2018-10-24

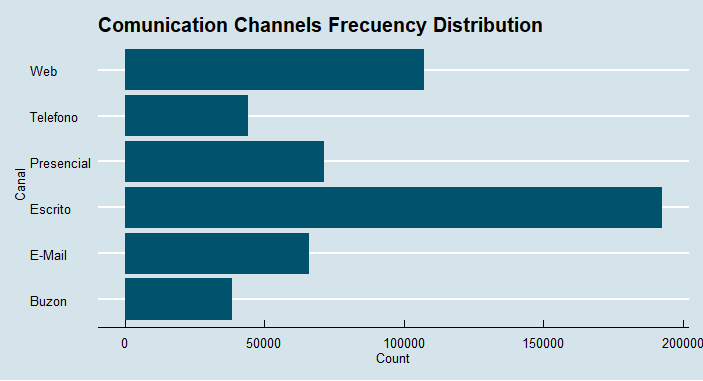
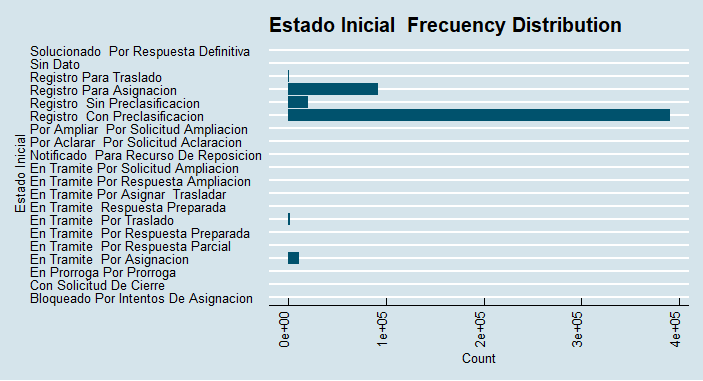
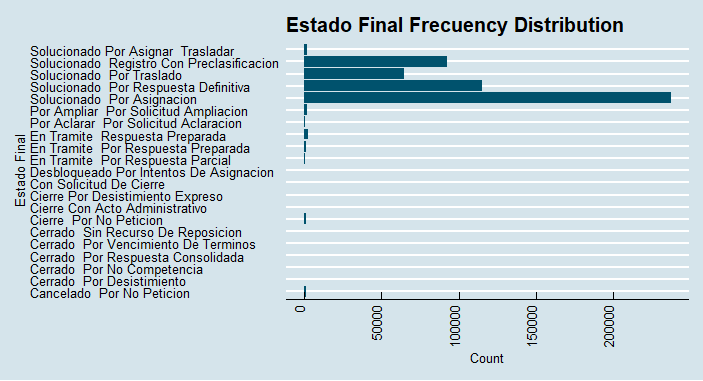
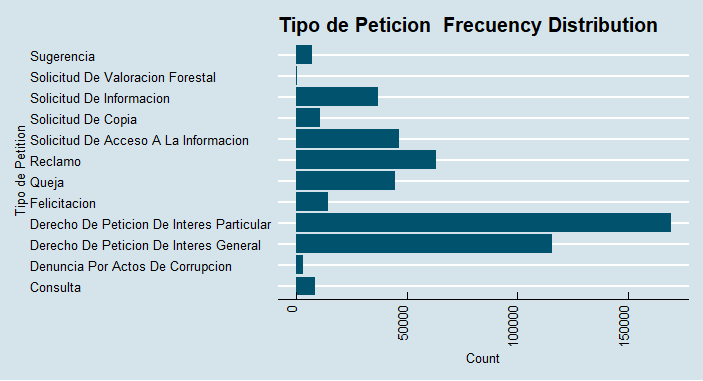
Max. :2020-01-09 Max. :2019-11-29 Max. :2019-11-30

1.2 Data Visualization

Data visualization is also a useful way to observe the structure of the data. For instance, the “Sector” variable had different labels for the “Gestion Publica” label: “GESTION PUBLICA” using uppercase letters, or “Gestion Publica (Nueva)”.

The variable “Channels de Comunication” and “Tema” had a lot of mislabeled classes. However, after reformatting them, it was easier to visualize their frequency distribution, as evidenced by the graphs below.





1.3 Data Transformation

All work up to this point has been related to cleaning the features of the dataset. However, as of yet, there was not a specific target variable. The available data was just date columns that describe when a claim was processed, when it was solved, and its respective legal deadline. Nevertheless, two variables of interest were emerging: “Fecha de Vencimiento” and “Fecha de Finalizacion”. The first described the date when legally a claim needed to be solved and the second was when the personnel of the Citizen’s Observatory solved the claim. The goal at this point was to defined when the claim was solved before the legal requirements and when it was not.

To accomplish this goal, it was necessary to transform the date variables using a couple of R packages like “Lubridate” and “Zoo”. The next step is to change the variables’ data type from a character to a date.

> data$FechaVencimiento<-as.Date(data$FechaVencimiento)

> data$FechaFinalizacion<-as.Date(data$FechaFinalizacion)

1.3.1 Fecha de Vencimiento

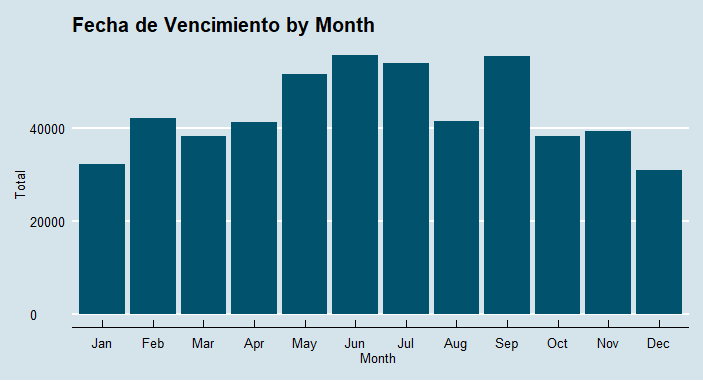
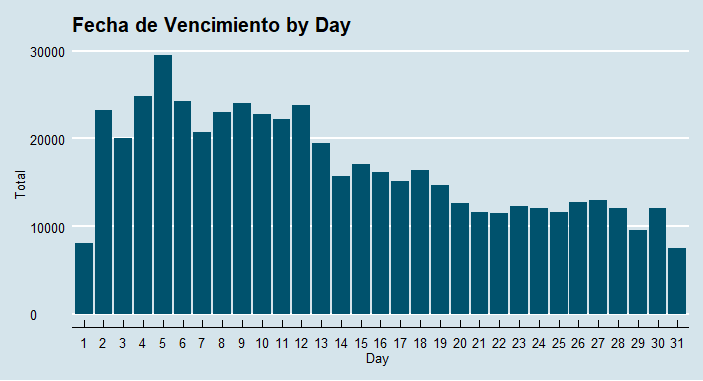
All components of the variable “Fecha de Vencimiento” were separated in different temporal variables. This meant new variables would be created for the day, month, year and day of the week for every observation.

> data$dayFv<- factor(day(data$FechaVencimiento))

> data$monthFv <- factor(month(data$FechaVencimiento, label = TRUE))

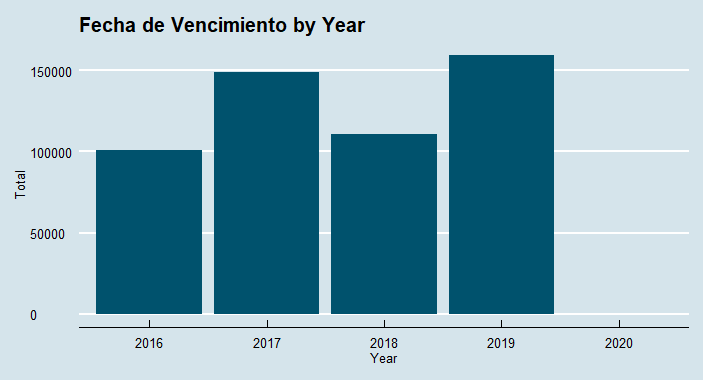
> data$yearFv <- factor(year(data$FechaVencimiento))

> data$dayofweekFv <- factor(wday(data$FechaVencimiento, label = TRUE))

The information extracted by day shows that in the first 15 days more claims were solved than in the next 15 days on average.

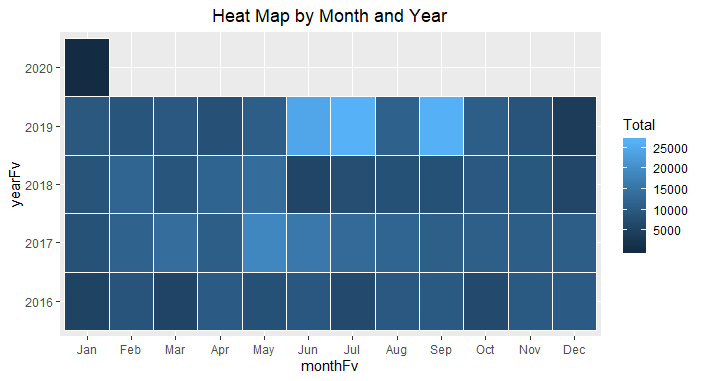
The “Fecha de Vencimiento” data by month shows that more claims had deadlines during the middle months (Summer) of the year than during the beginning and end of the year.

Furthermore, “Fecha de Vencimiento” by year shows that more claims were given deadlines in 2019 than in previous years. The new variables were giving a clearer picture of the trends emerging regarding the Citizen’s Observatory.



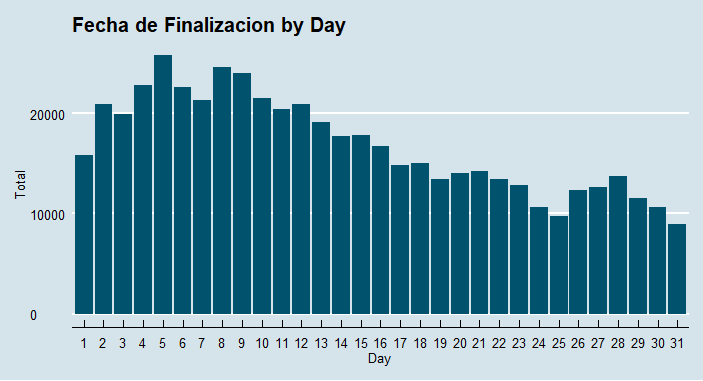
An alternative way to visualize the information is to look at the year and the month at the same time using what is known as a Heat Map. This visualization is more detailed and appears to be a better approach, providing a better understanding of the data.

According to the Heat Map by month and year, more claims were assigned deadlines in June, July and September of 2019 than in previous years. An argument could be made that this increase was due to the fact that it was an election year and all claims had to be resolved by the first of August with the start of a new Administration. This is a typical practice of the Colombian government where all government projects are frozen during the transition after an election to avoid corruption.

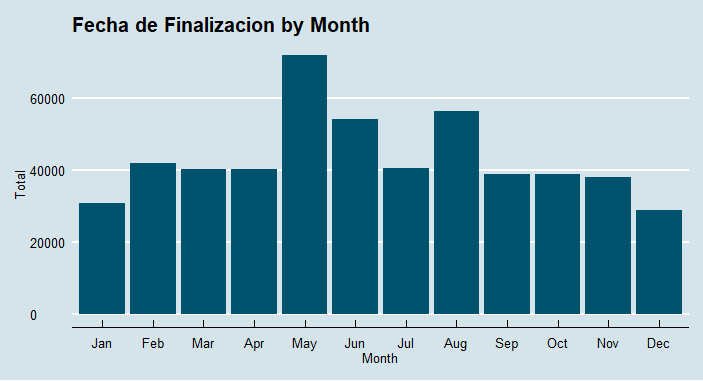
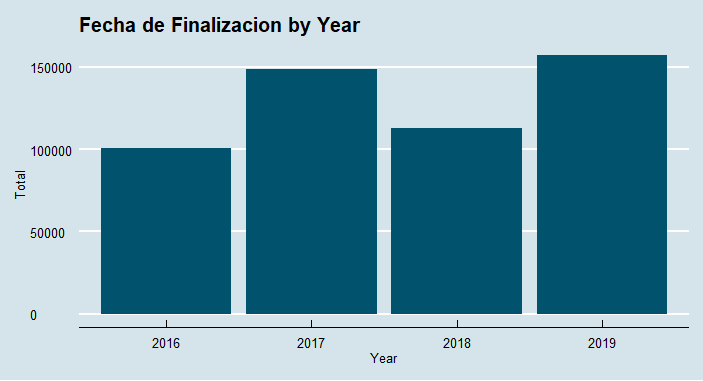


1.3.2 Fecha de Finalizacion

Like “Fecha de Vencimiento”, “Fecha de Finalizacion” has the same code execution. For “Fecha de Finalizacion” by day, it is notable that it follows the same right skewed distribution similar to “Fecha de Finalizacion by Day”. This could mean that there is a tendency to solved claims the first 15 days of the month.



Monthly and Yearly data from “Fecha de Finalizacion” also show the same patterns that “Fecha de Vencimiento” for the same type of temporal data.



1.3.3 Target Variable

Having the “Fecha de Vencimiento” and “Fecha de Finalizacion” in the right date format, it meets the requirements to use the function “difftime()”.

> data$dayFFFV<-difftime(data$FechaVencimiento,

data$FechaFinalizacion, units = "day")

This function basically created a new variable called “dayFFFV” as a result of the difference between “Fecha de Vencimiento” and “Fecha de Finalizacion”. This output identified two sets of numbers, positive and negative. Positive numbers mean that the claims were solved before the legal deadline and negative means that the claim expired, and the Citizen’s Observatory Office missed their deadline. This data is useful in that it demonstrates how frequently the Citizen’s Observatory does or does not miss their deadlines making them vulnerable to legal action against them.

Since there are different datetime variables, the next variable summary was executed, and it showed the basic statistical characteristics of the date variables that originally came with the dataset, as well as those that were a produced.

DiasVencimiento FiDay FiMonth FiDayWeek FiYear

Min. : 0.0000 5 : 27401 May : 71852 Sun: 9054 2016:100996

1st Qu.: 0.0000 4 : 27179 Aug : 56349 Mon: 91306 2017:148865

Median : 0.0000 8 : 27028 Jun : 54148 Tue:106790 2018:112985

Mean : 0.4394 2 : 25902 Feb : 41726 Wed:107130 2019:157264

3rd Qu.: 0.0000 1 : 24830 Jul : 40362 Thu:100342

Max. :72.0000 6 : 24492 Apr : 40256 Fri: 94649

(Other):363278 (Other):215417 Sat: 10839

dayFv monthFv yearFv dayofweekFv dayFF

5 : 29525 Jun : 55657 2016:100931 Mon: 90301 5 : 25802

4 : 24833 Sep : 55377 2017:148812 Tue:114803 8 : 24623

6 : 24324 Jul : 54017 2018:110886 Wed:107472 9 : 24029

9 : 24016 May : 51622 2019:159448 Thu:107159 4 : 22776

12 : 23777 Feb : 42080 2020: 33 Fri:100375 6 : 22613

2 : 23292 Aug : 41457 10 : 21508

(Other):370343 (Other):219900 (Other):378759

monthFF yearFF dayofweekFF dayFFFV

May : 71853 2016:100996 Sun: 3973 Length:520110

Aug : 56349 2017:148865 Mon: 91599 Class :difftime

Jun : 54148 2018:112984 Tue:108948 Mode :numeric

Feb : 41726 2019:157265 Wed:108072

1.4 Geographical Data Visualization

Another interesting variable in the dataset is the variable “Localidad” which represents the different districts in which Bogota is divided. In total there are twenty “Localidades”. Although the dataset did not contain all of the claims by "Localidad, it had most of them.

Now that the variable with the different number of days that the Citizen’s Observatory solved claims was available, the next step was to show the average of solved claims by localidad. The variable “Localidad” was grouped and the mean of days was calculated for each.

> dayFVFFlocaMean<-data %>%

+ group\_by(Localidad) %>%

+ summarise(no\_rows = mean(dayFFFV))%>%

+ arrange(desc(no\_rows))

When looking at the data frame dayFVFFlocaMean, it is obvious that there are two rows that were not required; “Interlocalidad” and “Sin Dato” labels. Basically, those two instances do not fulfill the requirement of the 20 localidades required for mapping. The next piece of code shows how row 8 and 20 were deleted.

> dayFVFFlocaMean<- dayFVFFlocaMean[-c(8),]

> dayFVFFlocaMean<- dayFVFFlocaMean[-c(20),]

The next code section shows how the titles of the dayFVFFlocaMean were changed, putting the title of “localidad” and “averDaySolve” for the average of days required to solve a claim. Additionally, a new variable needed to be created. This variable is called “CodLoc” and it represents the number for every “Localidad”.

> names(dayFVFFlocaMean)<-c('Localidad','AverDaySolve')

> dayFVFFlocaMean$CodLoc<-c('6','8','5','10','18',

'4','9','14','11','19',

'20','2','3','1','16',

'12','7','15','17','13')

While analyzing the structure of the data, it is seen that “the variable “CodLoc” is a string, and the “AverDaySolve” are not integers. To solve that problem, both variables were changed so that their data type was an integer as seen in the next two lines of code.

> dayFVFFlocaMean$CodLoc<-as.integer(dayFVFFlocaMean$CodLoc)

> dayFVFFlocaMean$AverDaySolve<-as.integer(dayFVFFlocaMean$AverDaySolve)

Once the data subject of analysis is ready, it is key to call the packages that will help in the creation of the map for the case, “tmap” and “sf”. Using the function st\_read() the shapefile of “Bogota” is called and this data is stored in a data frame called “Bogotalocal”.

Additionally, the variable CODIGO\_LOC is transformed into an integer given that this variable needs to be compared with the data frame dayFVFFlocaMean, specifically with the variable CodLoc in order to merge the two datasets as shown in the same line of code below.

> Bogolocal<- st\_read("localidades/localidades.shp")

Bogolocal$CODIGO\_LOC<-as.integer(Bogolocal$CODIGO\_LOC)

> common\_local <- union(Bogolocal$CODIGO\_LOC, dayFVFFlocaMean$CodLoc)

> length(common\_local) == length(Bogolocal$CODIGO\_LOC)

[1] TRUE

> BogdayFFFV<- merge(x=Bogolocal, y=dayFVFFlocaMean, by.x=c("CODIGO\_LOC"),

+ by.y=c("CodLoc"), all.y=TRUE)

Once the shape file of “Bogota” and the object “dayFVFFlocaMean” were in the same data frame, the average days to solve claims by localidades were ready to plot.

The next piece of code shows how the map implementation was executed. It used the function tm\_poligons() which was in charge of drawing the lines of the map and fitting the selected variables according to the geographical location. For this task, the variable averDaySolve is selected and the different shades of blue are chosen to characterize the map.

> ttm()

> tmap\_last()

> tm\_shape(BogdayFFFV) +

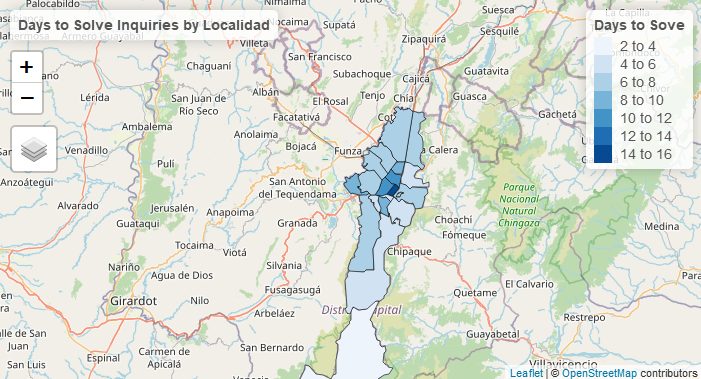
+ tm\_polygons(col = "AverDaySolve", id = "NOMBRE",palette='Blues',

+ border.alpha = 0.6,border.col = 'black', title='Days to Sove')+

+ tm\_layout(title = 'Days to Solve Inquiries by Localidad')

The following figure describes the average days needed to solve claims by Localidad. Unfortunately, it was not possible to label “Localidades” although R has the option to export HTML files which allows us to see the label for every localidad.

A basic analysis to the geographical data visualization says that except for the localidad “Los Martires” the number of days required to solve claims are uniform.



1.4 Exporting Data from R

After observing the available data, it was concluded that from 2016 to 2019, there was not a significant difference among the data during this timeframe. Because of this, the year 2018 was filtered and a copy of the same time was made, renaming it “data2” without selecting the variables required for the model implementation.

> data2<-data%>%

+ filter(FiYear=='2018')

> data2<-data2[,c(1,2,3,4,5,6,7,8,9,10,22)]

> data2$dayFFFV<-as.numeric(data2$dayFFFV)

After several tries, the target variable behaved better when it was turned into binary terms. For that reason, the values of the variable “dayFFFV” were converted to ones for positive values and zeros for negative ones.

> data2$dayFFFV[data2$dayFFFV >=0]<- 1

> data2$dayFFFV[data2$dayFFFV<0] <-0

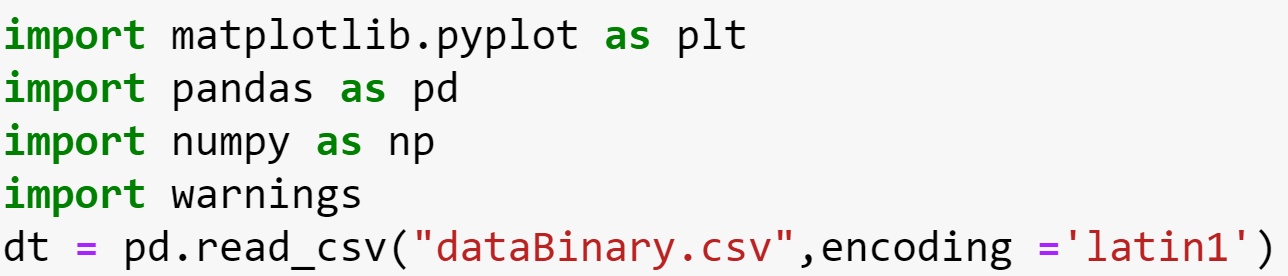
> data2$dayFFFV<-factor(data2$dayFFFV)

The use of R ended here because the model implementation is executed using Python, more specific Jupyter Notebooks. However, it is necessary to export the data frame “data2” to carry out the next task.

write.csv(data2,"dataBinary.csv", row.names = FALSE)

1.5 Import Data with Python

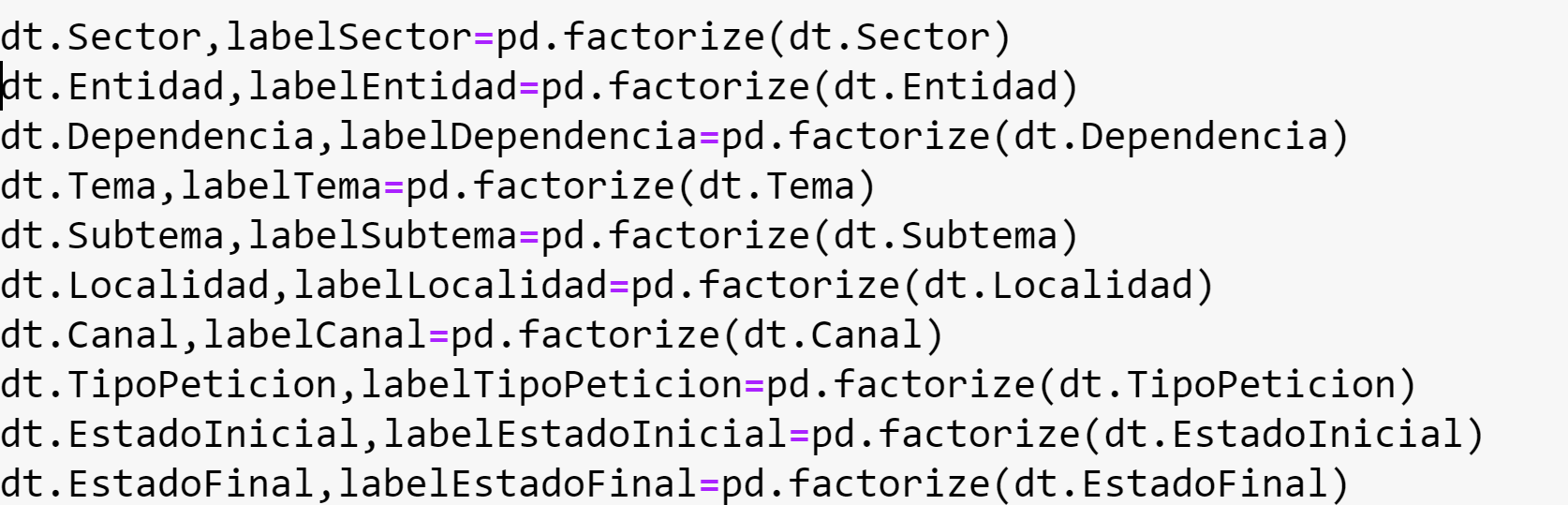
Continuing in the construction of the two machine learning models, with Python the following packages were imported: matplotlib, pandas, and numpy. The following piece of code shows each line of code used to import the clean dataset created in R called dataBinary which came in a csv format.



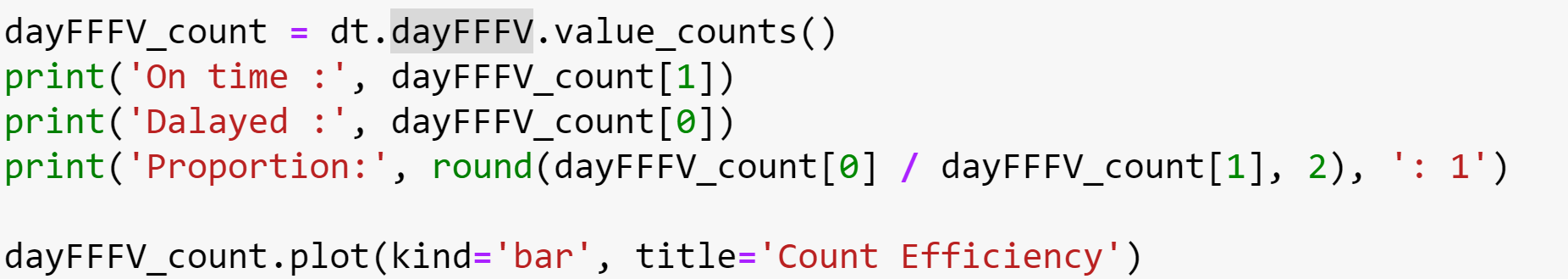
1.6 Data Encoding

In order to optimize the performance of Python, all variables were encoded to numerical values and a label for every variable was created with the purpose of relating it to its origin.

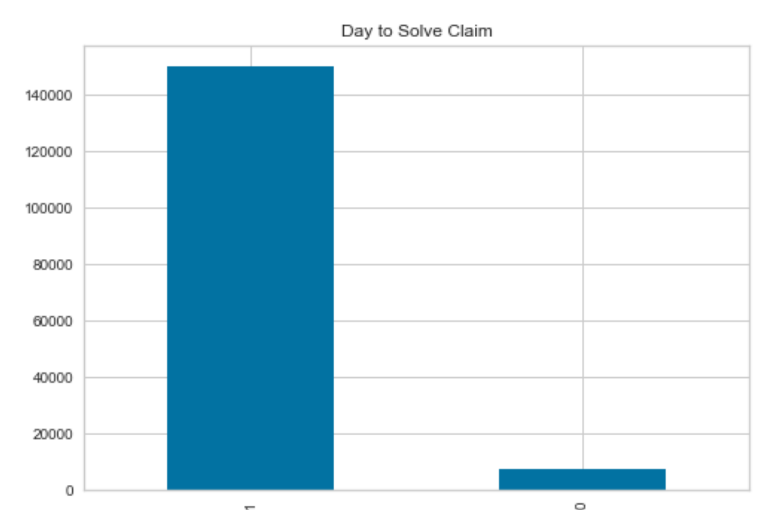
The next piece of code solves that need by using the function factorize from pandas.



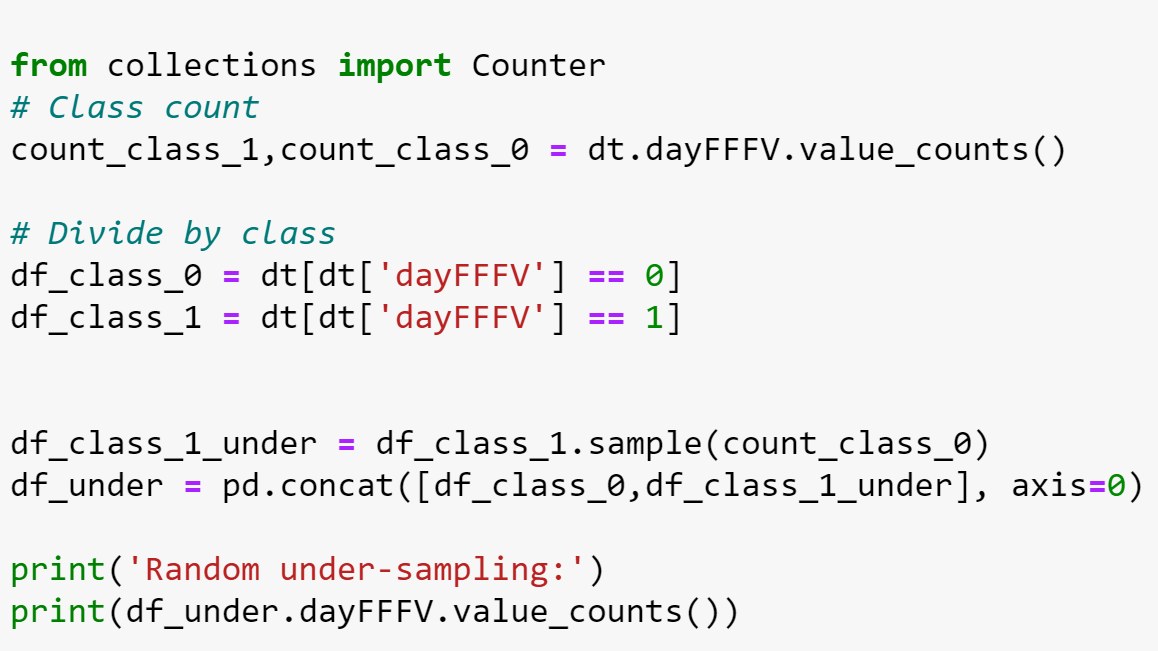
After encoding the dataset, it was necessary to observe how the target variable was distributed. The following lines of code counted the classes of the variable “dayFFFV” and described whether the variable was balanced or not.



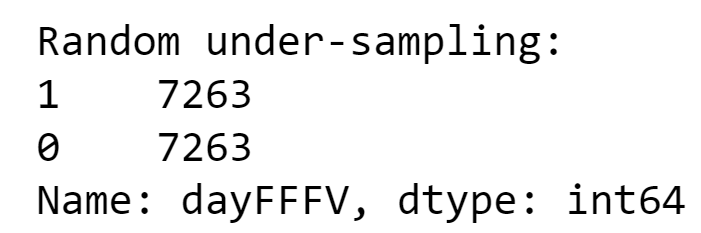
The result of the last code execution is demonstrated in the figure below showed how unbalanced the dataset was. The class “On time” displayed more observations than the “Delayed” ones.



1.7 Random Over-Sampling for Solving Imbalance Dataset  
One way to fix an unbalanced dataset is to randomly under-sample the current dataset based on the class with less observations. That is, randomly reduce the number of observations of the class “On time” using as a reference the label with less observations “Delayed”.



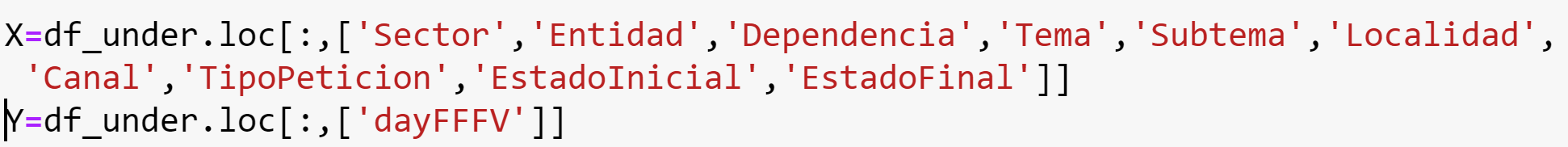
The output of the above lines of code is the following figure which shows how the binary variable has the same number of classes.



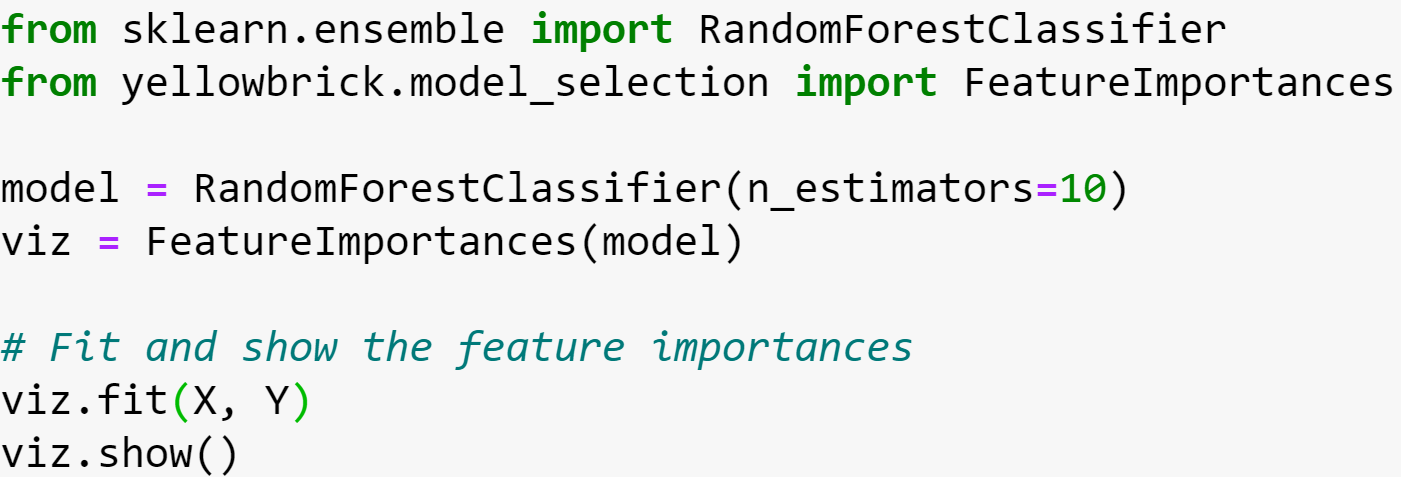
1.8 Feature Selection

The next step in the model creation is to measure the level of impact that every feature has over the target variable in order to define variables that are not required to be included in the final model.

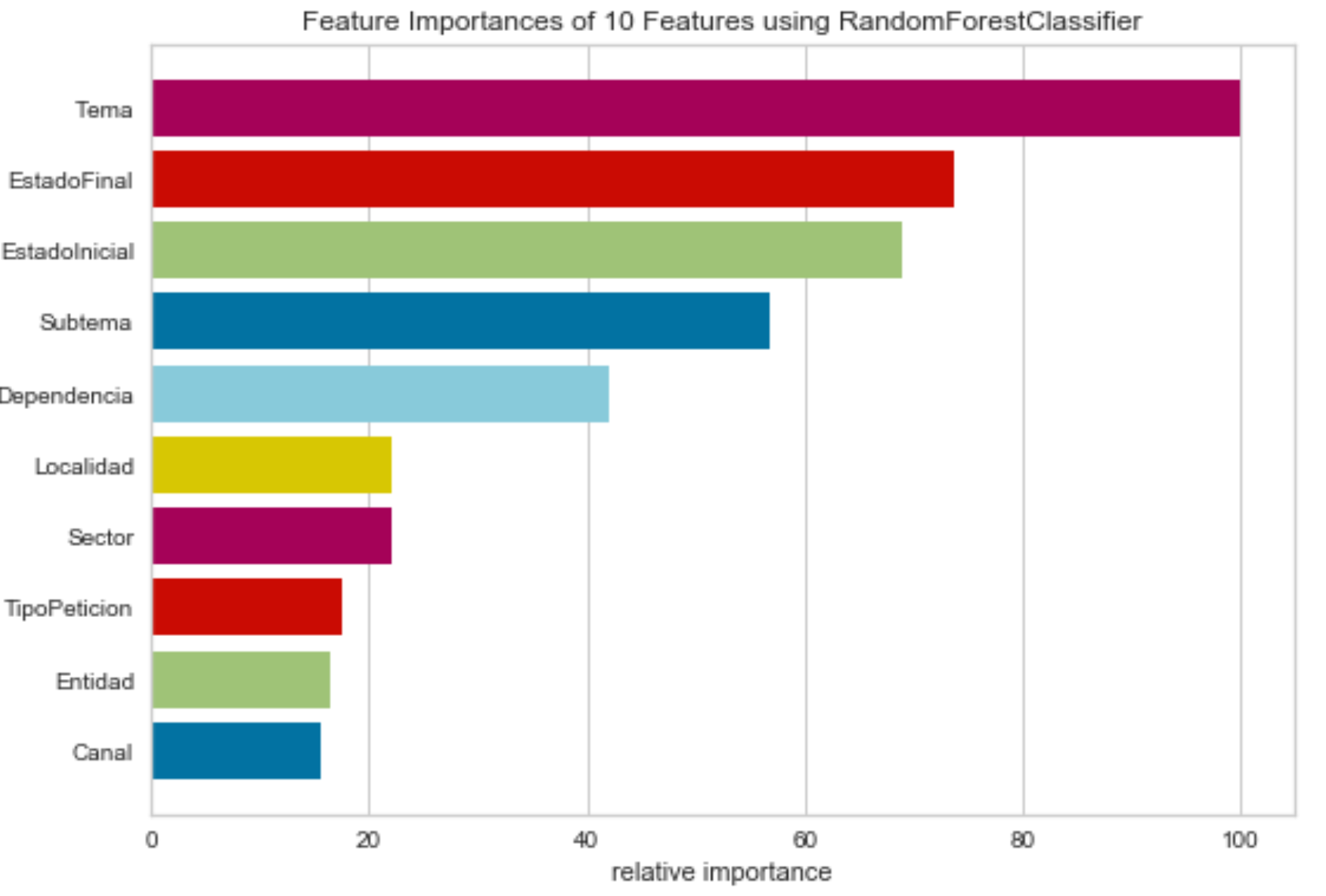
So far, there was one dataset that included both feature and target variables. However, in order to continue, it was important to split the main dataset by features using the label “X” and target variable using the label “Y” based on the random under-sampled dataset.



Since both “X” and “Y” are discrete data types in nature, a built-in random forest model from the “sklearn” package was going to be used to assess future impact. To this purpose, the “Yellowbricks” package was also used because it fits the model and offers a visualization to determine the rank of importance for all features.



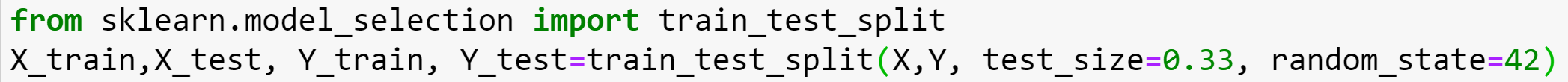
The following figure shows the rank of importance for every feature. The most important was the variable “Tema” and the least important was the variable “Canal”. Nevertheless, all featured variables were regarded as statistically significant.



1.9 Training Data and Test Data

With a clean dataset, a balanced target variable and the appropriate features, it was time to start deploying the two machine learning models. However, before this could happen, the methodology states that it is necessary to split both dataset “X” and “Y” into two subsets called “training data” and “test data”. Basically, the purpose of this step is to let the machine learning model learn using the training data and assess the model’s performance using the test data.

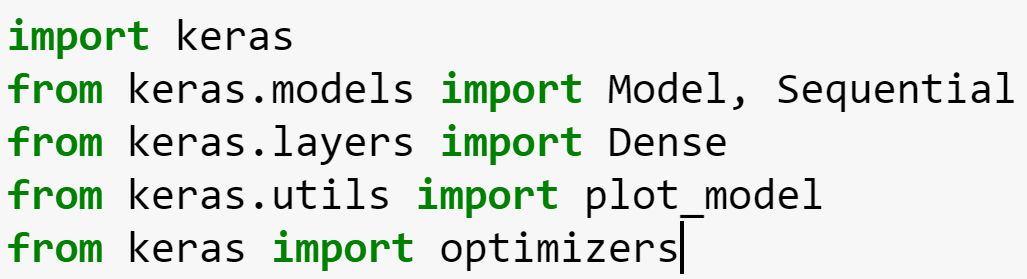
The Sklearn package offers the function train\_test\_split() function to divide “X” and “Y” using a random selection and limiting the subset of data using the argument test\_size which allocated 77% of the original dataset to the training data and 33% to the test data.



2. Machine Learning Application

2.1 Neural Network Model

The first machine learning model is a neural network which was built with the help of a Keras package. The following line of code imported the basic libraries required to implement the model.



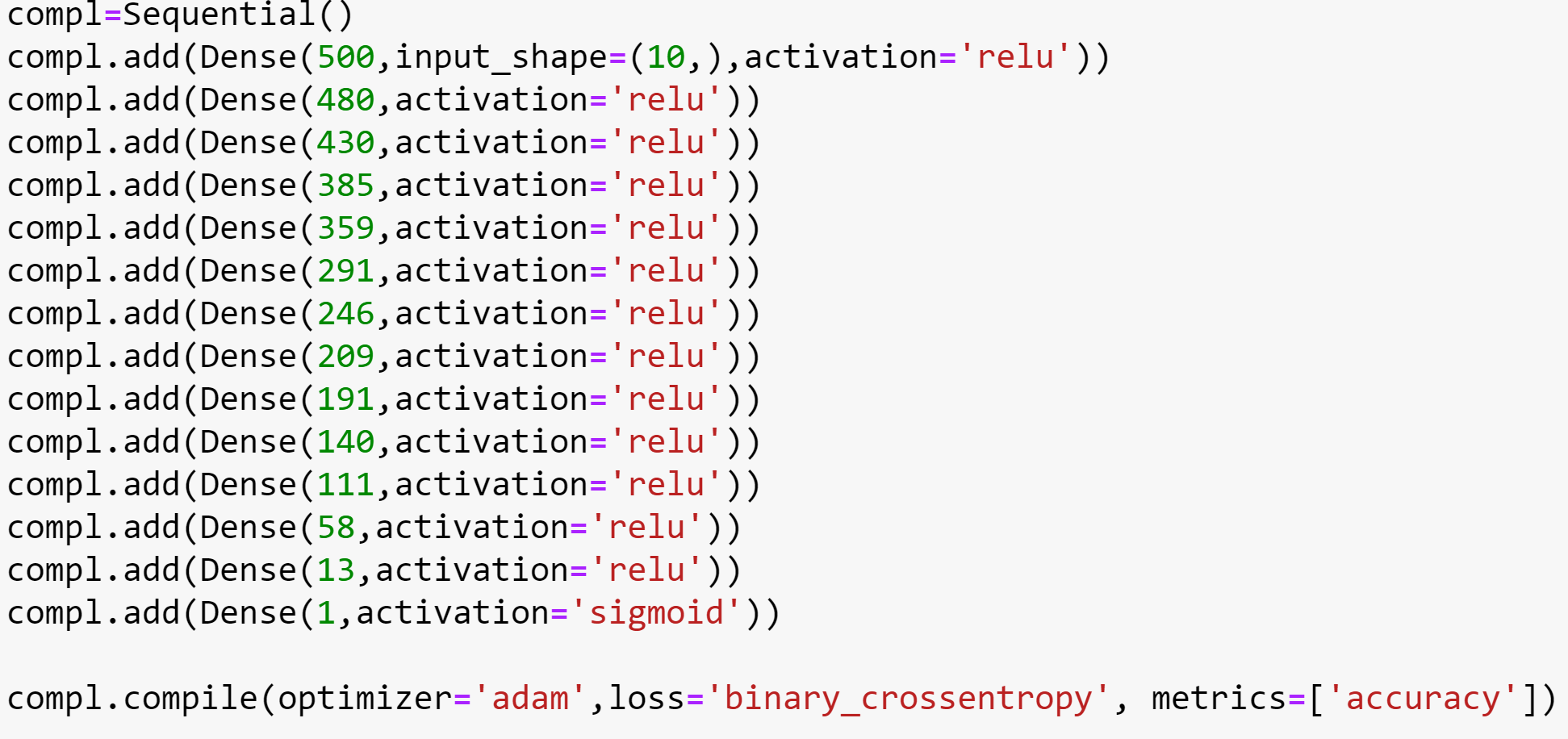
The function sequential() basically created the model “compl”. This function allows us to create models layer by layer. For instance, the first layer recognized 10 input variables using a relu activation function and created 500 nodes.

There is a descending number of nodes through the whole neural network and the same relu activation function is applied. However, the last layer included the target variable and it required an activation function according to the problem. In this case, a sigmoid activation function was chosen since the problem was to classify a binary target variable.

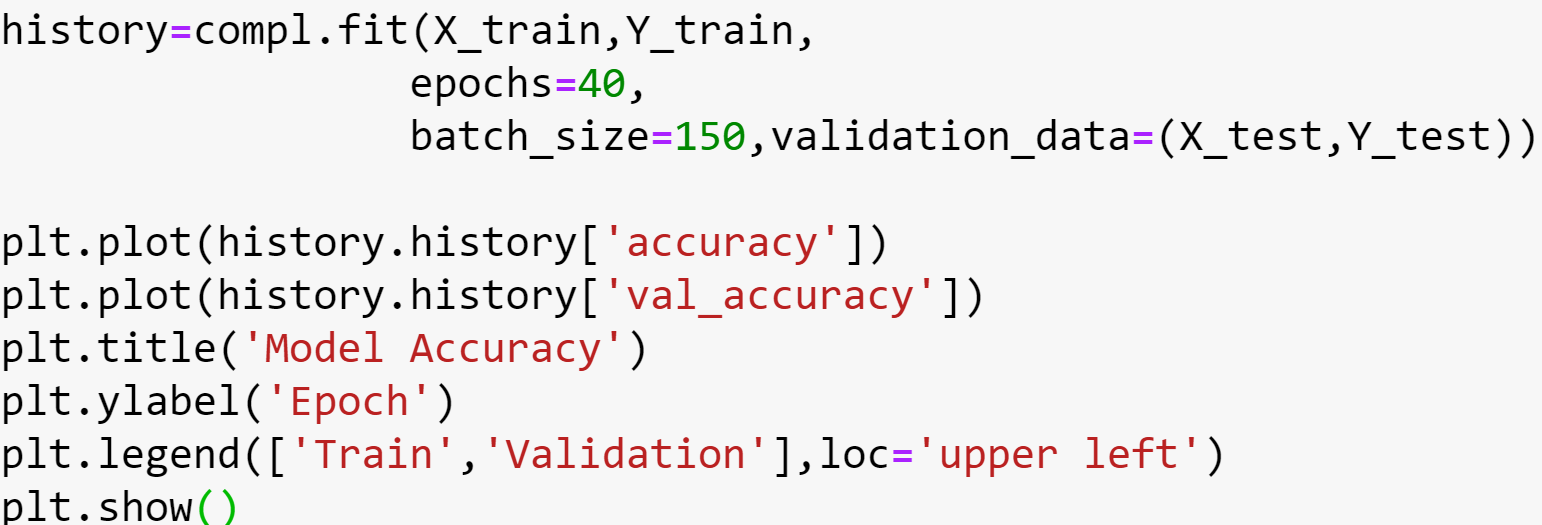
The same block of code has the compile() function that basically contains the type of optimizer, loss function and sort of metric as the main argument in order to assess the model’s performance.

For binary classification, it is normally required to have a “binary\_crossentropy” loss function. The loss function allows for an assessment of how well the model behaves when comparing real values with respect to those generated by the model. The goal is to have a minimum value which demonstrates that real and predicted values are alike. On the other hand, the optimizer was selected using the methodology of trial and error. After trying a couple of times, the optimizer “adam” was selected. It is important to remember that the optimizer updates the network weights in order to give more precision to the predictions of the training data.

Finally, there is the accuracy metric that shows how every model iteration performed.

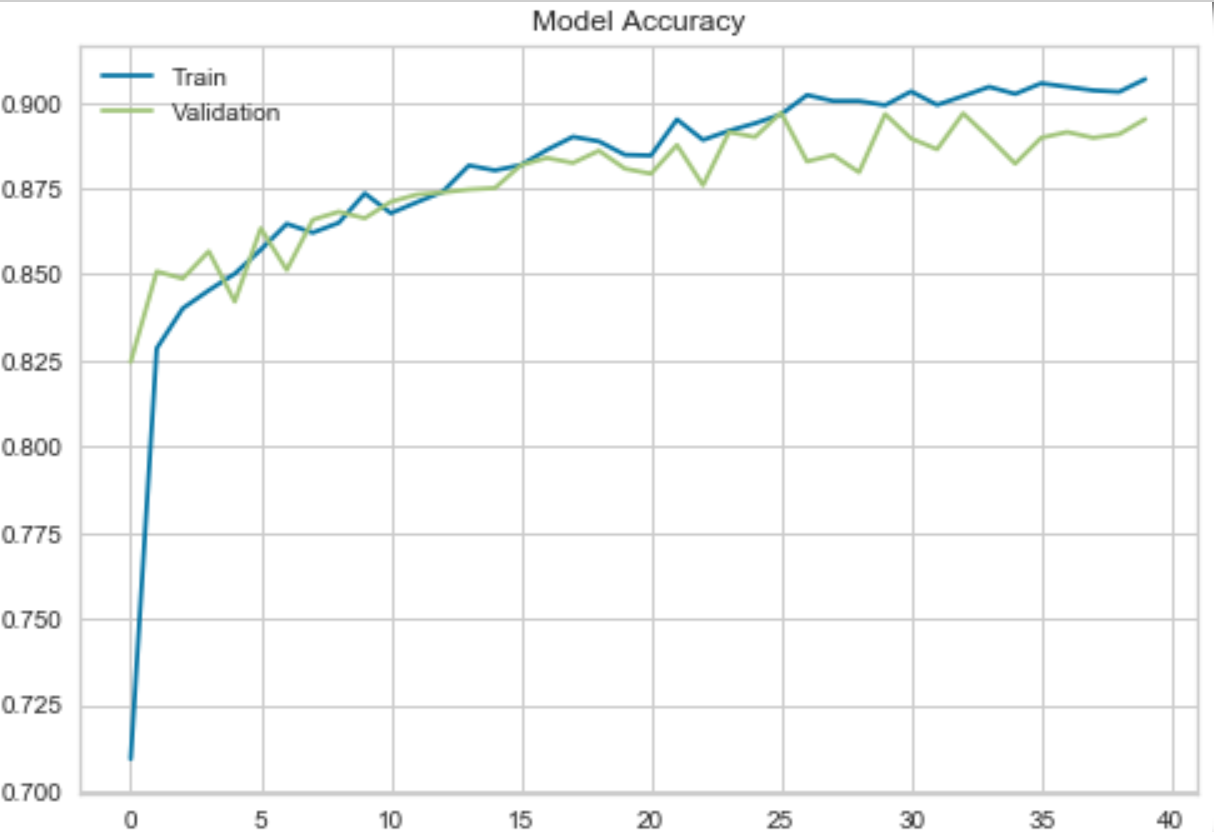


Once the model was created, the next step was to fit the data using the function fit() of keras. The output of the next section of code also displayed how well the training and test data behaved with respect to specifying the number of epochs (The number of times the algorithm is going to iterate through the whole dataset). A batch size of 150 is chosen, meaning that for every epoch there are 150 training samples.

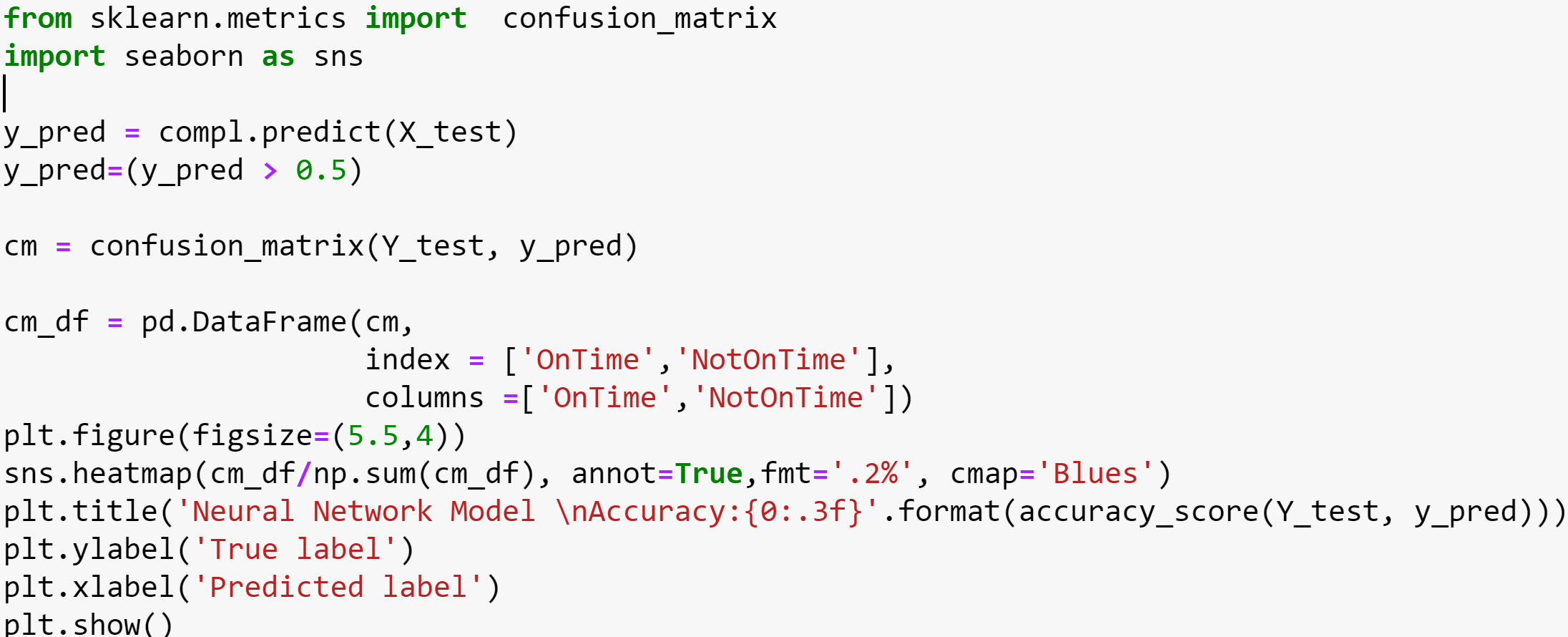
`

Model Assessment

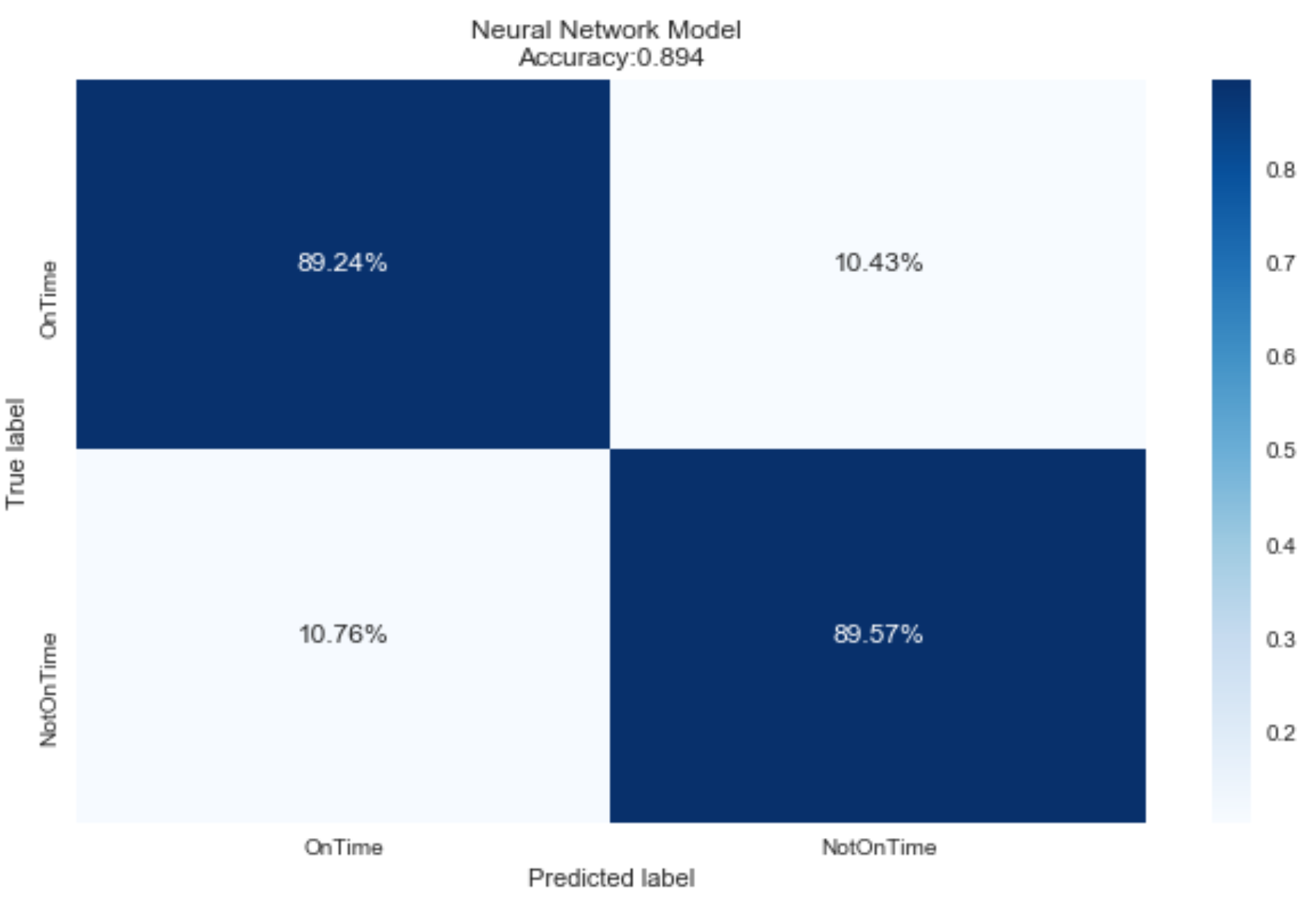
The next figure shows the performance of both training and test data as they were measured by using the accuracy metric. What they show is that the neural network model learned and both datasets achieved a high level of accuracy.



Accuracy is not regarded as the best way to analyze a classification model. For that reason, a confusion matrix was used that displayed the number of correct classifications for both classes and misclassifications as well.

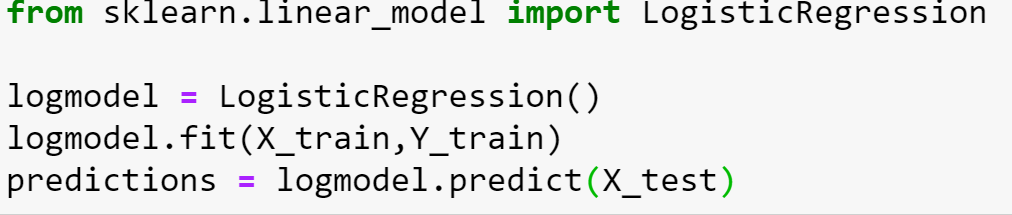


The result of the above lines of code is the next confusion matrix. In it, we can see how well both classes are predicted. According to the matrix, the “On Time label” and “Not on Time” were predicted correctly 89% of the time. However, 10% was not correctly classified and it is necessary to keep this in mind when drawing conclusions about the results.



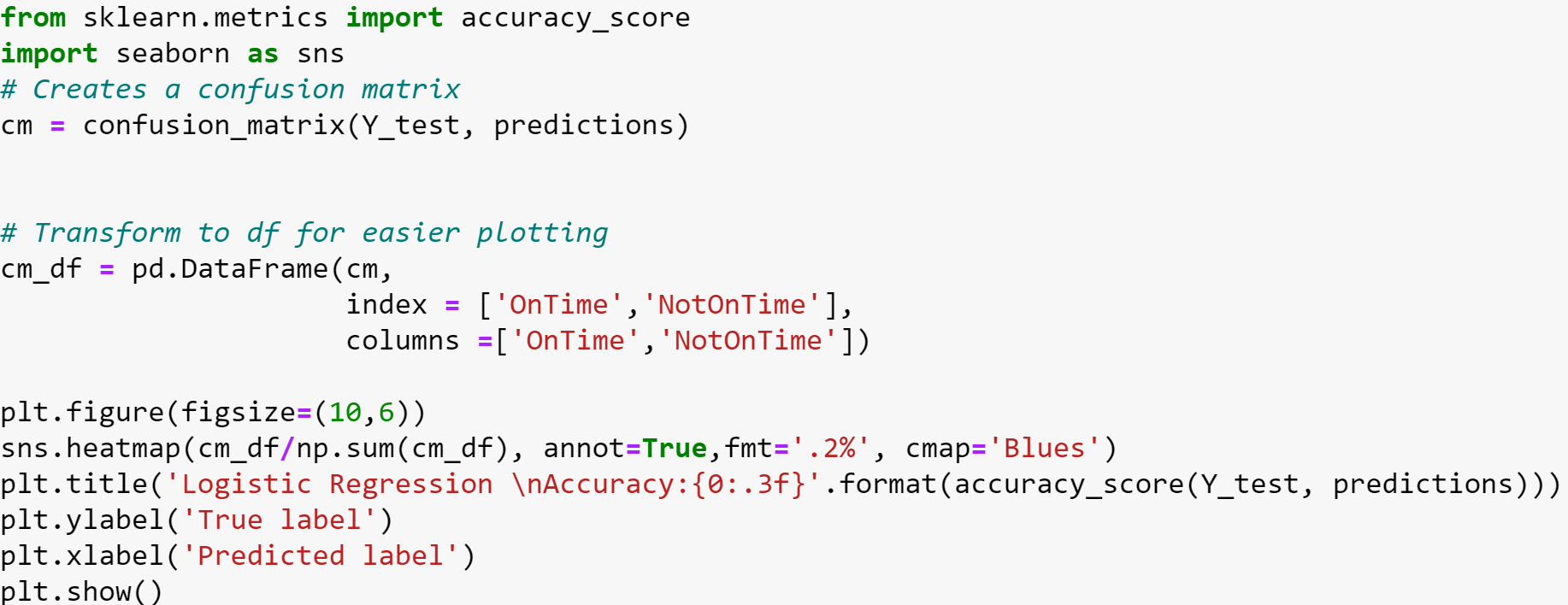
2.2 Logistic Regression Model

A logistic regression model is basically a generalization of a linear regression model which does not have a continuous target variable, instead it has a binary target variable. The logistic regression model was chosen as another sort of classification model given its performance. The implementation of this model is based on a sklearn package which has a function call “LogisticRegression”

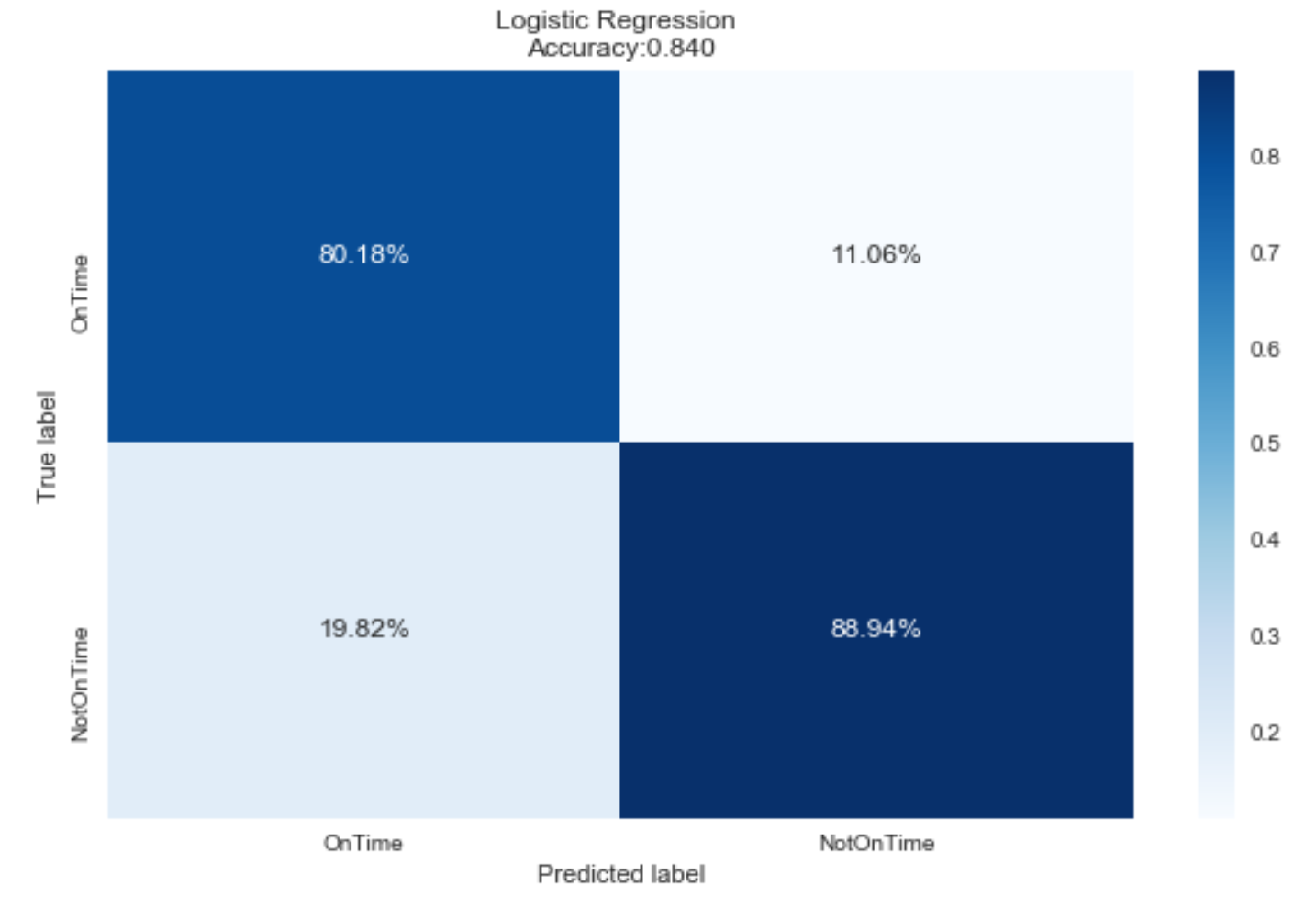


Model Assessment

In order to evaluate the model’s performance, a confusion matrix is used once again and the accuracy is scored.



Since a logistic model does not have the complexity level that a Neural Network has, its performance level is obviously lower. Its accuracy was 84% and the correct classification was around 80% for the class “On Time” and 88% for the class “Not On Time”. However, the confusion matrix had a 20% false negative misclassification of classes and 11% false positive classes.



Discussion

This capstone project offers several different points of analysis:

* The major goal of this capstone project was to create two models that would predict the reaction of the Citizen’s Observatory to solve claims. Two machine learning implementations demonstrated that a neural network performed better than a logistic regression model predicting the target variable with a level of accuracy of 89%.
* The most common type of claims,“Tipo de Peticion”, are the “Derechos de Peticion” which are basically claims that have legal implications if they are not answered within the first 15 days of summiting the claim.
* Although, 5 claims were not answered, compared to the 100 claims that were, by the Citizen’s Observatory; and the variable “Tipo de Peticion” had only a mild level of impact on the target variable, it should be assumed that it is necessary for the Citizen’s Observatory to take actions in order to avoid legal implications in the future.
* The information provided by the geographical visualization demonstrate that all “Localidades” or districs in Bogota have uniform levels of reaction from the Citizen’s Observatory, with the exception being the district “Los Martires”. This can be explained since “Los Martires” is considered a “high crime area” with documented instances of prostitution and illegal activities.

Limitations

There are several different aspects of this capstone project that required attention, given the nature of the raw material that was the basis of this analysis:

* The dataset had a high frequency of mislabeling in almost all variables. This meant, that when every instance was created, there was not a consistent format to introduce the data from the functionaries of the Citizen’s Observatory. The consequence of different sorts of names for a unique label required massive data transformation based on the criteria of this project designer.
* Imbalanced dataset problems were found since there were more active responses to claims that delayed reactions. The proportion was 5 to 100. The first executions of the machine learning models gave a low level of accuracy, as well as, more than 10% were false positive and false negative in the confusion matrix. Finally, there a randomly under-sampling technique was implemented in order to balance the dataset, and the results had a positive impact in the metrics mentioned before.
* Even though the dataset did not have missing values, there were labels like “Sin Datos” that showed a problem with assigned values for every attribute or column.

Conclusion

Given the level of assertiveness demonstrated by metrics like accuracy and the confusion matrix, there is strong evidence showing that all attributes had a significant impact on when the Citizen’s Observatory solved every claim. However, the three main variables “Tema”, “EstadoFinal”, and “Subtemas” played a more important role in the behavior of the target variable.

An implementation of the neural network classifier would benefit the performance and image of the Citizen’s Observatory. Since the variable ‘Tema’ was the most influential, a good alternative would be to pay more attention to those topics included in this variable.

Competencies

The present capstone project addresses the following Ischool student competencies:

* C1 Computational and analytic thinking and doing decomposition, pattern recognition, abstraction, and algorithms in solving information and data challenges, in addition to analysis.
* C1.A: Given the size of the database and its results, the present project was divided in three part in order to tackle it in a better way. One stage for data cleaning and exploratory data analysis. A second stage involve visualization. The third stage will be model implementation using a deep leaning and logistic regression model.
* C1B: In order to carry out the project goal, it was fundamental to implement a pattern recognition process where the similarities among instances and correlations between attributes are established.
* C1.C: The first stage of the project, data cleaning, allowed a better specification of the information required to the present analysis given there would be data not applicable to the final purpose.
* C1.D: The second stage of the project will implement two machine leaning algorithms to predict the reaction of the citizen’s observatory to solve claims where there were analyzed different metric to assess the model’s performance.
* C1.E: This project is going to use Python for data analysis and prediction, R to create charts and maps.
* C2 Data manipulation, analysis, and interpretation: Students will obtain the skills of collecting, manipulating, and analyzing different types of data at different scales, and interpreting the results properly.
* C2.A: The dataset is made basically of categorical and time data. There is not continuous data in the dataset.
* C2.B: R and Python will be implemented as computational methods to clean, format, transfer and store data.
* C2.C: In the second stage, two machine learning algorithms will be implanted: logistic regression model and a Neural Network model to predict Citizen’s efficiency answering claims, as well visualization tools using maps and charts.
* C2.E: The information entered in the database is anonymized. However, all ethical parameters were followed to ensure transparency in the project execution.
* C3 Communication and teamwork: Students will acquire skills to work with others within and across disciplines and be effective communicators.
* C3.A: Working with members of the School of Information and external professionals (Citizen’s Observatory office) ensuring communication channels to gain experience about the topic under analysis, methodological tool to carry out the project and correct approach to the cycle of this project.
* C3.B: The project findings will be articulated using different tools such as charts, maps, diagrams and communication performance.
* C4 Creative contributions: The project is designed to provide an innovative way to tackle efficiency in the Citizen’s Observatory Office given tools to shows points where the administration can focus its attention more effectively. The methodology used to implement this is based on exploratory data analysis using statistical computational methods and predictive models with respective model validation that will measure it predictively capacity.
* C5 Ethics and Values: Students will demonstrate an understanding of information/data ethics, and the values of information fields to serve diverse user groups
* The project product was based on ethical standards and good values given professional qualities of the person in charge of the same as well as the inner characteristics of the study that is public and ensures anonymity of the information sources.