



Trabajo Práctico N° 4 Random Forest

Presentado en la fecha: 19/10/2019

Hecho por: Andrada Alexander

Encina Guadalupe

Huarca Brian

Contents

Introducción	2
Objetivo	3
Modelo Predictivo Random Forest	4
0.1 Transformacion y Limpieza de Datos	4
0.2 Cargar Datos	7
0.2.1 Dataset de Entrenamiento	7
0.2.2 Crear dataset de Test	8
0.3 Aplicacion del modelo RandomForest	9
0.4 Analisis de Matriz de Confusion	9
0.5 Analisis de Arbol de Decision	10
Conclusión	11
Anexo	12

Introducción

En el presente trabajo desarrollaremos el proceso llevado a cabo para la realización y obtención de un **Modelo Predictivo de Clasificación** dado un dataset de la **Fundación Bunge y Born** obtenido del sitio web *fundacionbyb.org*. A lo largo del desarrollo, se detallará los pasos que se deben seguir para realizar la predicción mediante el uso de la Interfaz RStudio del lenguaje R y como técnica predictiva Random Forest.

El conjunto de datos muestra la afinidad de Chagas de una provincia(**Índice de Prevalencia Potencial de Chagas** de cada radio censal). Agregando por localidad para las variables de Afinidad para con la zona endémica y Vulnerabilidad Sanitaria.

Objetivo

Se tiene por objetivo mostrar el proceso llevado a cabo para la obtencion de un modelo predictivo tal que me permita predecir una variable categorica. A partir del conjunto elegido y transformado, se busca representar datos estadisticos y de grafico para una mejor comprension del modelo obtenido

Modelo Random Forest

0.1 Tranformacion y Limpieza de Datos

En primera instancia se tuvo que extraer y transformar los datos desde un .PDF Para la transformacion del mismo se utilizo como Pagina de referencia: <https://www.ilovepdf.com>

Nom_Prov	Nom_Dpto	Nom_Local	Media_Af	Media_VulSanit	Max_VulSanit	Max_Af	Rad_Extremo
C.A.B.A.	Comuna 01	COMUNA 1	0.0859	0.2593	0.8793	0.2804	FALSE
C.A.B.A.	Comuna 02	COMUNA 2	0.0747	0.0643	0.8090	0.1635	FALSE
C.A.B.A.	Comuna 03	COMUNA 3	0.0845	0.1579	0.6838	0.3929	TRUE
C.A.B.A.	Comuna 04	COMUNA 4	0.0639	0.3139	0.9705	0.1071	FALSE
C.A.B.A.	Comuna 05	COMUNA 5	0.0627	0.1233	0.4338	0.0902	FALSE
C.A.B.A.	Comuna 06	COMUNA 6	0.0604	0.1476	0.2768	0.1038	FALSE
C.A.B.A.	Comuna 07	COMUNA 7	0.0469	0.3037	0.7972	0.0595	FALSE
C.A.B.A.	Comuna 08	COMUNA 8	0.0493	0.5124	0.8280	0.0824	FALSE
C.A.B.A.	Comuna 09	COMUNA 9	0.0441	0.3367	0.6932	0.0676	FALSE
C.A.B.A.	Comuna 10	COMUNA 10	0.0454	0.2508	0.4414	0.0595	FALSE
C.A.B.A.	Comuna 11	COMUNA 11	0.0485	0.1962	0.3423	0.0600	FALSE
C.A.B.A.	Comuna 12	COMUNA 12	0.0504	0.2289	0.5195	0.0647	FALSE
C.A.B.A.	Comuna 13	COMUNA 13	0.0627	0.1967	0.4594	0.1763	FALSE
C.A.B.A.	Comuna 14	COMUNA 14	0.0723	0.1023	0.3662	0.1150	FALSE
C.A.B.A.	Comuna 15	COMUNA 15	0.0581	0.2094	0.5804	0.1053	FALSE
Buenos Aires	Adolfo Alsina	MAZA	0.0228	0.5882	0.6035	0.0228	FALSE
Buenos Aires	Adolfo Alsina	LA PALA	0.0228	0.6748	0.6748	0.0228	FALSE
Buenos Aires	Adolfo Alsina	ZONA RURAL	0.0174	0.6593	0.9071	0.0344	FALSE
Buenos Aires	Adolfo Alsina	YUTUYACO	0.0105	0.8726	0.8726	0.0105	FALSE
Buenos Aires	Adolfo Alsina	RIVERA	0.0105	0.2924	0.4512	0.0105	FALSE
Buenos Aires	Adolfo Alsina	DELFIN HUERGO	0.0105	0.6460	0.6460	0.0105	FALSE
Buenos Aires	Adolfo Alsina	ESTEBAN AGUSTIN GASCON	0.0116	0.6921	0.6921	0.0116	FALSE
Buenos Aires	Adolfo Alsina	CNIA. SAN MIGUEL ARCANGEL	0.0116	0.7449	0.7511	0.0116	FALSE
Buenos Aires	Adolfo Alsina	CARHUE	0.0181	0.2218	0.5792	0.0181	FALSE
Buenos Aires	Adolfo Alsina	ESPARTILLAR	0.0223	0.4486	0.4486	0.0223	FALSE
Buenos Aires	Adolfo González Chaves	ZONA RURAL	0.0162	0.6849	0.8745	0.0631	FALSE
Buenos Aires	Adolfo González Chaves	JUAN E. BARRA	0.0121	0.7360	0.7360	0.0121	FALSE
Buenos Aires	Adolfo González Chaves	DE LA GARMA	0.0121	0.1634	0.1983	0.0121	FALSE
Buenos Aires	Adolfo González Chaves	ADOLFO GONZALES CHAVES	0.0068	0.2727	0.6062	0.0068	FALSE
Buenos Aires	Adolfo González Chaves	VASQUEZ	0.0413	0.8532	0.8532	0.0413	FALSE
Buenos Aires	Alberti	ZONA RURAL	0.0103	0.6057	0.8704	0.0214	FALSE
Buenos Aires	Alberti	VILLA ORTIZ	0.0085	0.5526	0.5562	0.0085	FALSE
Buenos Aires	Alberti	CORONEL SEGUI	0.0085	0.5674	0.5674	0.0085	FALSE

Figure 1: Datos Iniciales

La Figura 1 corresponde a un mapa calorico que refleja la AFINIDAD de las provincias a contraer CHAGAS. Los datos originales que obtuvimos de la página: Fundacion BUNGE Y BORN

A continuación se muestra la estructura de la tabla sobre la cual se trabajara.

Como se puede observar en la Figura 3 (Taba de contenidos) se perciben 8 variables, cuyas descripciones se desglosan a continuación:

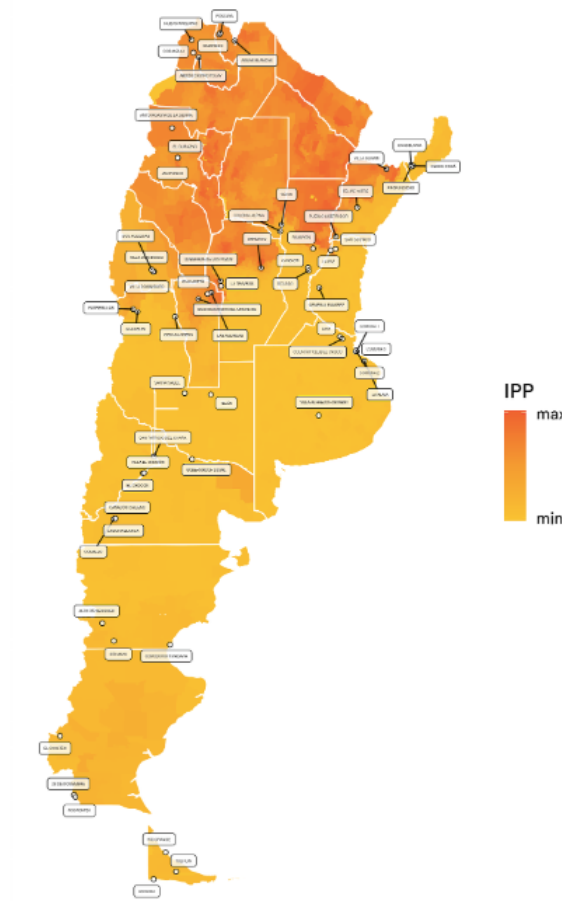


Figure 2: Mapa Calorico

- *Nom_Prov* : Nombre de la Provincia.
- *Nom_Dpto* : Nombre del Departamento.
- *Nom_Local* : Nombre de la Localidad.
- *Media_Af* : Media de afinidad_chagas.
- *Media_VulSanit* : Score de vulnerabilidad sanitaria.
- *Max_VulSanit* : Máximo valor observado de vulnerabilidad sanitaria en radios.
- *Max_Af* : Máximo valor observado de afinidad_chagas en radios.
- *Rad_Extremo* : si el valor es TRUE es que existe radio con afinidad_chagas con valores estadísticamente similares a los de la zona endémica.

A continuacion se muestra el dataset completo limpiado y transformado:

	Nom_Prov	Nom_Dpto	Nom_Local	Media_Af	Media_VulSanit	Max_VulSanit	Max_Af	Rad_Extremo
1	C.A.B.A.	Comuna 01	COMUNA 1	0,0859	0,2593	0,8793	0,2804	FALSE
2	C.A.B.A.	Comuna 02	COMUNA 2	0,0747	0,0643	0,8090	0,1635	FALSE
3	C.A.B.A.	Comuna 03	COMUNA 3	0,0845	0,1579	0,6838	0,3929	TRUE
4	C.A.B.A.	Comuna 04	COMUNA 4	0,0639	0,3139	0,9705	0,1071	FALSE
5	C.A.B.A.	Comuna 05	COMUNA 5	0,0627	0,1233	0,4338	0,0902	FALSE
6	C.A.B.A.	Comuna 06	COMUNA 6	0,0604	0,1476	0,2768	0,1038	FALSE
7	C.A.B.A.	Comuna 07	COMUNA 7	0,0469	0,3037	0,7972	0,0595	FALSE
8	C.A.B.A.	Comuna 08	COMUNA 8	0,0493	0,5124	0,8280	0,0824	FALSE
9	C.A.B.A.	Comuna 09	COMUNA 9	0,0441	0,3367	0,6932	0,0676	FALSE
10	C.A.B.A.	Comuna 10	COMUNA 10	0,0454	0,2508	0,4414	0,0595	FALSE
11	C.A.B.A.	Comuna 11	COMUNA 11	0,0485	0,1962	0,3423	0,0600	FALSE
12	C.A.B.A.	Comuna 12	COMUNA 12	0,0504	0,2289	0,5195	0,0647	FALSE
13	C.A.B.A.	Comuna 13	COMUNA 13	0,0627	0,1967	0,4594	0,1763	FALSE
14	C.A.B.A.	Comuna 14	COMUNA 14	0,0723	0,1023	0,3662	0,1150	FALSE
15	C.A.B.A.	Comuna 15	COMUNA 15	0,0581	0,2094	0,5804	0,1053	FALSE
16	Buenos Aires	Adolfo Alsina	MAZA	0,0228	0,5882	0,6035	0,0228	FALSE
17	Buenos Aires	Adolfo Alsina	LA PALA	0,0228	0,6748	0,6748	0,0228	FALSE
18	Buenos Aires	Adolfo Alsina	ZONA RURAL	0,0174	0,6593	0,9071	0,0344	FALSE
19	Buenos Aires	Adolfo Alsina	YUTUYACO	0,0105	0,8726	0,8726	0,0105	FALSE

Figure 3: Tabla de contenidos

	Nom_Prov	Nom_Dpto	Nom_Local	Media_Af	Media_VulSanit	Max_VulSanit	Max_Af	Rad_Extremo
1	C.A.B.A.	Comuna 01	COMUNA 1	0,0859	0,2593	0,8793	0,2804	0
2	C.A.B.A.	Comuna 02	COMUNA 2	0,0747	0,0643	0,8090	0,1635	0
3	C.A.B.A.	Comuna 03	COMUNA 3	0,0845	0,1579	0,6838	0,3929	1
4	C.A.B.A.	Comuna 04	COMUNA 4	0,0639	0,3139	0,9705	0,1071	0
5	C.A.B.A.	Comuna 05	COMUNA 5	0,0627	0,1233	0,4338	0,0902	0
6	C.A.B.A.	Comuna 06	COMUNA 6	0,0604	0,1476	0,2768	0,1038	0
7	C.A.B.A.	Comuna 07	COMUNA 7	0,0469	0,3037	0,7972	0,0595	0
8	C.A.B.A.	Comuna 08	COMUNA 8	0,0493	0,5124	0,8280	0,0824	0
9	C.A.B.A.	Comuna 09	COMUNA 9	0,0441	0,3367	0,6932	0,0676	0
10	C.A.B.A.	Comuna 10	COMUNA 10	0,0454	0,2508	0,4414	0,0595	0
11	C.A.B.A.	Comuna 11	COMUNA 11	0,0485	0,1962	0,3423	0,0600	0
12	C.A.B.A.	Comuna 12	COMUNA 12	0,0504	0,2289	0,5195	0,0647	0
13	C.A.B.A.	Comuna 13	COMUNA 13	0,0627	0,1967	0,4594	0,1763	0
14	C.A.B.A.	Comuna 14	COMUNA 14	0,0723	0,1023	0,3662	0,1150	0
15	C.A.B.A.	Comuna 15	COMUNA 15	0,0581	0,2094	0,5804	0,1053	0
16	Buenos Aires	Adolfo Alsina	MAZA	0,0228	0,5882	0,6035	0,0228	0
17	Buenos Aires	Adolfo Alsina	LA PALA	0,0228	0,6748	0,6748	0,0228	0

Figure 4: Tabla General

0.2 Cargar Datos

La realizacion del mismo se hizo mediante dos tipos de Datasets: ENTRENAMIENTO Y TEST. El primero se realiza para entrenar el modelo y el segundo para medir la precision del mismo

0.2.1 Dataset de Entrenamiento

El primer modelo de tabla se hizo teniendo en cuenta el 72porciento de los datos del dataset original, está hecho de manera intuitiva.

	Nom_Prov	Media_Af	Media_VulSanit	Max_VulSanit	Max_Af	Rad_Extremo
1	C.A.B.A.	0.0859	0.2593	0.8793	0.2804	0
2	C.A.B.A.	0.0747	0.0643	0.8090	0.1635	0
3	C.A.B.A.	0.0845	0.1579	0.6838	0.3929	1
4	C.A.B.A.	0.0639	0.3139	0.9705	0.1071	0
5	C.A.B.A.	0.0627	0.1233	0.4338	0.0902	0
6	C.A.B.A.	0.0604	0.1476	0.2768	0.1038	0
7	C.A.B.A.	0.0469	0.3037	0.7972	0.0595	0
8	C.A.B.A.	0.0493	0.5124	0.8280	0.0824	0
9	C.A.B.A.	0.0441	0.3367	0.6932	0.0676	0
10	C.A.B.A.	0.0454	0.2508	0.4414	0.0595	0
11	C.A.B.A.	0.0485	0.1962	0.3423	0.0600	0
12	Buenos Aires	0.0073	0.2927	0.6458	0.0073	0
13	Buenos Aires	0.0242	0.7519	0.8808	0.1049	0
14	Buenos Aires	0.1049	0.8808	0.8808	0.1049	0
15	Buenos Aires	0.0104	0.8081	0.8081	0.0104	0
16	Buenos Aires	0.0293	0.7989	0.7989	0.0293	0
17	Buenos Aires	0.0093	0.2607	0.6846	0.0119	0
18	Buenos Aires	0.0186	0.7613	0.8845	0.0570	0

Figure 5: Dataset Entrenamiento

0.2.2 Crear dataset de Test

El segundo dataset lo obtenemos a partir de la observación general del dataset principal, con la diferencia de que este tendrá una transformación de variables como las de tipo Factor para el caso de la Variable NombreProvincia.

	Nom_Prov	Nom_Dpto	Nom_Local	Media_Af	Media_VulSanit	Max_VulSanit	Max_Af	Rad_Extremo
1	C.A.B.A.	Comuna 01	COMUNA 1	0.0859	0.2593	0.8793	0.2804	0
2	C.A.B.A.	Comuna 02	COMUNA 2	0.0747	0.0643	0.8090	0.1635	0
3	C.A.B.A.	Comuna 03	COMUNA 3	0.0845	0.1579	0.6838	0.3929	1
4	C.A.B.A.	Comuna 04	COMUNA 4	0.0639	0.3139	0.9705	0.1071	0
5	C.A.B.A.	Comuna 05	COMUNA 5	0.0627	0.1233	0.4338	0.0902	0
6	C.A.B.A.	Comuna 06	COMUNA 6	0.0604	0.1476	0.2768	0.1038	0
7	C.A.B.A.	Comuna 07	COMUNA 7	0.0469	0.3037	0.7972	0.0595	0
8	C.A.B.A.	Comuna 08	COMUNA 8	0.0493	0.5124	0.8280	0.0824	0
9	C.A.B.A.	Comuna 09	COMUNA 9	0.0441	0.3367	0.6932	0.0676	0
10	C.A.B.A.	Comuna 10	COMUNA 10	0.0454	0.2508	0.4414	0.0595	0
11	C.A.B.A.	Comuna 11	COMUNA 11	0.0485	0.1962	0.3423	0.0600	0
12	C.A.B.A.	Comuna 12	COMUNA 12	0.0504	0.2289	0.5195	0.0647	0
13	C.A.B.A.	Comuna 13	COMUNA 13	0.0627	0.1967	0.4594	0.1763	0
14	C.A.B.A.	Comuna 14	COMUNA 14	0.0723	0.1023	0.3662	0.1150	0
15	C.A.B.A.	Comuna 15	COMUNA 15	0.0581	0.2094	0.5804	0.1053	0
16	Buenos Aires	Adolfo Alsina	MAZA	0.0228	0.5882	0.6035	0.0228	0
17	Buenos Aires	Adolfo Alsina	LA PALA	0.0228	0.6748	0.6748	0.0228	0

Showing 1 to 21 of 2,668 entries, 8 total columns

```

~/
$ ./
$ 'data.frame': 1932 obs. of 6 variables:
$ Nom_Prov : Factor w/ 20 levels "Buenos Aires",...: 2 2 2 2 2 2 2 2 2 2 ...
$ Media_Af : num 0.0859 0.0747 0.0845 0.0639 0.0627 0.0604 0.0469 0.0493 0.0441 0.0454 ...
$ Media_VulSanit: num 0.2593 0.0643 0.1579 0.3139 0.1233 ...
$ Max_VulSanit : num 0.879 0.809 0.684 0.971 0.434 ...
$ Max_Af : num 0.2804 0.1635 0.3929 0.1071 0.0902 ...
$ Rad_Extremo : Factor w/ 2 levels "0","1": 1 1 2 1 1 1 1 1 1 1 ...

```

Figure 6: Dataset Test

0.3 Aplicacion del modelo RandomForest

Con el fin de obtener el modelo, se aplico la funcion RandomForest que especifica en mayor medida y de manera aleatoria la mayor precision que me acerca al modelo de interes.

A continuacion se muestran las estadisticas del modelo indicando, ademas, el indice OOB:

```
> print(mod) # 25,23 err 3--0.71>

call:
  randomForest(x = dataf[training.ids, 4:7], y = dataf[training.ids,      8], ntree = 500, keep.forest = TRUE)
              Type of random forest: classification
              Number of trees: 500
No. of variables tried at each split: 2

              OOB estimate of  error rate: 0.05%
Confusion matrix:
      0   1 class.error
0 1752   1 0.0005704507
1    0 116 0.0000000000
>
```

Figure 7: Matriz Confusion

0.4 Analisis de Matriz de Confusion

Para obtener una mejor visualizacion de los aciertos predictivos de nuestro modelo, se emplea la matriz de confusión como herramienta para la visualización del desempeño de dicho modelo resultante. Cada columna de la matriz representa el número de predicciones de cada clase, y cada fila representa a los casos de clase reales.

```
> print(mod) # 25,23 err 3--0.71>

call:
  randomForest(x = dataf[training.ids, 4:7], y = dataf[training.ids,      8], ntree = 500, keep.forest = TRUE)
              Type of random forest: classification
              Number of trees: 500
No. of variables tried at each split: 2

              OOB estimate of  error rate: 0.05%
Confusion matrix:
      0   1 class.error
0 1752   1 0.0005704507
1    0 116 0.0000000000
>
```

Figure 8: Resumen del Modelo

Tasa de error general del modelo:

```

> # Crea prediccion y Matriz de confusion
> prediccion <- predict(modeloFore, training[], type='class'); # Matriz de confusion
> prediccion2 <- predict(modeloFore, test[], type='class'); # Matriz de confusion
> matriz_c
      Rad_Extremo
prediccion  0    1
0 1794    0
1      0 138
> matriz_c2
      Rad_Extremo
prediccion2 0    1
0 2502    0
1      1 165

```

Figure 9: MatrizConfusion Entrenamiento y Test

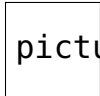
Test.png Test.png  pictures/Indice_error Test.png

Figure 10: Indice Error

0.5 Analisis de Arbol de Decision

Para predecir las provincias propensas a contraer Chagas, podría usarse el algoritmo de árbol de decision como clasificador. Con el modelo creado y el árbol, se aplican a los datos de TEST para realizar predicción y medir precisión del árbol.

A continuación se describe el Analisis:

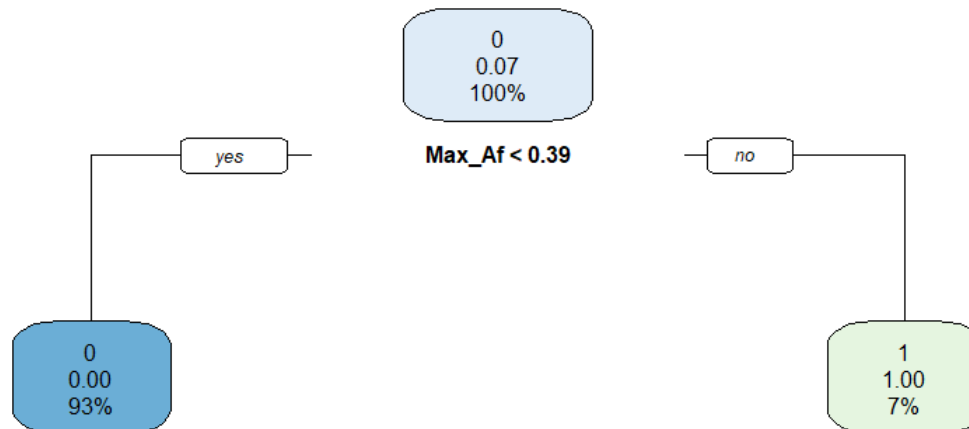


Figure 11: Árbol Decision

Conclusión

Luego del desarrollo del análisis (mas alla de no obetener los resulados esperados de forma particular), y enfocalizando en las distintas funciones que nos permiten obtener resultados concretos y proximos, concluimos que el Análisis de Random Forest nos permite realizar un estudio predictivo muy poderoso con respecto al algoritmo de Arboles de Decision, debido a que este no solo busca un camino para obtener los resultados o nodos, sino que buscan de forma aleatoria distintos caminos para el desarrollo predictivo.

Sin duda que para el ambito de la salud es una gran herramienta aplicada desde simples diagnosticos hasta un nivel de observacion mas alla como este caso en principio emplea registros de llamadas telefonicas para establecer patrones de conducta de las personas..

Anexo

```
1 library(readr)
2 library(MASS)
3 library(tree)
4 library(caret)
5 library(rpart)
6 library(rpart.plot)
7 library(ROCR)
8 library(randomForest)
9 library(C50);
10 library(party)
11 library(RCurl)
12
13 ##### LIMPIEZA Y TRANSFORMACION DE DATOS #####
14 #####
15 #Dataframe Original sin Filtrar
16 chagas = read_excel("C:/Users/brian/Desktop/Arboles_Decision/Chagas/Chagas.xlsx")
17 #Debug del dataset Original
18 str(chagas)
19 View(chagas)
20 #False = 2503
21 #True = 165
22
23
24 #Transformacion de la Variable Categorica a Numerica
25 chagas$Rad_Extremo[chagas$Rad_Extremo == "FALSE"] <- 0
26 chagas$Rad_Extremo[chagas$Rad_Extremo == "TRUE"] <- 1
27 str(chagas)
28 chagas$Rad_Extremo = as.numeric(chagas$Rad_Extremo)
29 str(chagas)
30 length(chagas$Rad_Extremo=="TRUE")
31 View(chagas)
32
33 #0 = 2503
34 #1 = 165
35
36 #Exportamos para Dividir el dataset en ENTRENAMIENTO Y TEST
37 write.xlsx(chagas, 'chagas_train.xlsx')
38
39 chagas$Rad_Extremo = factor(chagas$Rad_Extremo)
40 str(chagas)
41 ##### METODO 1 DE RESOLUCION #####
42 ##### METODO 1 DE RESOLUCION #####
```

```

43 ##### METODO 1 DE RESOLUCION #####
44
45 ##### Dataset Entrenamiento #####
46 ##### Dataset Entrenamiento #####
47 ##### Dataset Entrenamiento #####
48
49 DF2 = read_excel("C:/Users/brian/Desktop/Arboles_Decision/Chagas/chagas_train.xlsx")
50 str(DF2)
51 View(DF2)
52 DF2$Rad_Extremo = factor(DF2$Rad_Extremo)
53
54 #Creacion del data frame
55 dataf2 = data.frame(DF2)
56 str(dataf2)
57 View(dataf2)
58
59 dataf2$Nom_Prov = as.factor(dataf2$Nom_Prov)
60
61 ##### Dataset TEST #####
62 ##### Dataset TEST #####
63 ##### Dataset TEST #####
64 DF3 = read_excel("C:/Users/brian/Desktop/Arboles_Decision/Chagas/chagas_test.xlsx")
65 str(DF3)
66 View(DF3)
67 DF3$Rad_Extremo = factor(DF3$Rad_Extremo)
68
69 #Creacion del data frame
70 dataf3 = data.frame(DF3)
71 str(dataf3)
72 View(dataf3)
73
74 dataf3$Nom_Prov = as.factor(dataf3$Nom_Prov)
75 ##### Dataset TEST #####
76 ##### Dataset TEST #####
77 ##### Dataset TEST #####
78
79 ## SELECCION DE VARIABLES
80 var <- c('Nom_Prov'
81          , 'Media_Af'
82          , 'Media_VulSanit'
83          , 'Max_VulSanit'
84          , 'Max_Af'
85          , 'Rad_Extremo')
86 View(var)
87 #----- preparo test con ENTRENAMIENTO
88 #-----
89 #-----Obtengo todos los registros con las variables de interes para
    predecir
90
91 training= dataf2[,var]
92 test= dataf3[,var]
93 str(training)
94 View(training)
95 str(test)
96 View(test)

```

```

97
98 #####
99 # comienzo el modelo randomforest
100 #####
101 set.seed(47)
102 ##### RF Video #####
103 modelo_rf <- randomForest(training$Rad_Extremo ~ ., data = training, mtry = 5)
104 modelo_rf
105 ##### RF Video #####
106
107 ##### RF PROFE #####
108 modelofore <- randomForest(Rad_Extremo ~ .
109                             , data=training
110                             , ntree=500 # cantidad de arboles
111                             , mtry=2    # cantidad de variables
112                             , replace=T  # muestras con reemplazo
113                             , importance=T)
114 #                                     , class = NULL)
115 ##### RF PROFE #####
116 modelofore
117 print(modelofore)
118 # Crea prediccion y Matriz de confusion
119 prediccion <- predict(modelofore, training[,], type='class'); # Matriz de Confusion
120 prediccion2 <- predict(modelofore, test[,], type='class'); # Matriz de Confusion
121
122 # si en la anterior linea de error cambiar tipo de dato a numerico
123 matriz_c <- with(training, table(prediccion, Rad_Extremo))
124 matriz_c
125
126 matriz_c2 <- with(test, table(prediccion2, Rad_Extremo))
127 matriz_c2
128
129 # calculo los aciertos totales:
130 err2= sum(matriz_c2[1,1], matriz_c2[2,2])/sum(matriz_c2)
131 err2
132
133
134 # -----NO FUNCA-----
135 # Grafico del error OOB en cada iteracion
136 X11()
137 tuneRF(x = training[,],      # data set de entrenamiento
138        y = training$Rad_Extremo, # variable a predecir
139        mtryStart = 1,    # cantidad de variables inicial
140        stepFactor = 2,   # incremento de variables
141        ntreeTry   =1000, # cantidad arboles a ejecutar en cada iteracion
142        improve    = 1    # mejora minima del OOB para seguir iteraciones
143 )
144
145 #####
146 #####
147 #####
148 # PASO 2: Crea Arbol de Decision
149 # -----
150 str(training)
151 View(training)

```

```

152 ModeloArbol1<-rpart(training$Rad_Extremo ~ .,data=training[], parms=list(split="
      information"))
153 print(ModeloArbol1)
154
155 X11()
156 rpart.plot(ModeloArbol1) ## aqui tengo el resultado graficamente en entrenamiento
157
158 X11() # otra forma de graficarlo
159 rpart.plot(ModeloArbol1, type=1, extra=100,cex = .7,
160           box.col=c("gray99", "gray88")[ModeloArbol1$frame$yval])
161 # -----
162 #####
163 #####
164 #####
165 #####
166 #####
167 #####
168
169 #####
170 #####
171 #####
172
173 #####
174 #####
175 #####
176
177 ##### METODO 2 DE RESOLUCION #####
178 ##### METODO 2 DE RESOLUCION #####
179 ##### METODO 2 DE RESOLUCION #####
180
181 ##### Dataset Entrenamiento #####
182 ##### Dataset Entrenamiento #####
183 ##### Dataset Entrenamiento #####
184
185 DF1 = read_excel("C:/Users/brian/Desktop/Arboles_Decision/Chagas/chagas_train.xlsx")
186 str(DF1)
187 View(DF1)
188 DF1$Rad_Extremo = factor(DF1$Rad_Extremo)
189
190 #Creacion del data frame
191 dataf1 = data.frame(DF1)
192 str(dataf1)
193 View(dataf1)
194
195 #Data entrenamiento
196 training.ids =createDataPartition(dataf1$Rad_Extremo, p = 0.7, list = F)
197 #Aplicacion de RandomForest
198 set.seed(2018)
199 mod=randomForest(x = dataf1[training.ids, 4:7],
200                 y = dataf1[training.ids,8],
201                 ntree = 500, #Numero de Arboles
202                 keep.forest = TRUE) #?????????
203
204 #Data de PRediccion o de test
205 pred=predict(mod, dataf1[-training.ids,])

```



```

206 #Matriz de Confusion
207 table(dataf[-training.ids,"Rad_Extremo"], pred, dnn=c("actual", "predicho"))
208
209 #Probabilidad del modelo Test o prediccion
210 probs = predict(mod, dataf[-training.ids,], type="prob")
211 head(probs)
212 pred <- prediction(probs[,2], dataf[-training.ids, "Rad_Extremo"])
213 #Grafico de performance del modelo test
214 perf <- performance(pred, "tpr", "fpr")
215 plot(perf)
216
217 #####
218 print(mod) # 25,23 err 3--0.71>;
219 gc()
220
221 #####
222 # Importance of each predictor.
223 round(importance(mod), 2)##### NO FUNCA
224 # Crea prediccion y Matriz de confusion
225 prediccion <- predict(mod, test[,-1], type='Rad_Extremo'); # Matriz de Confusion
226 # si en la anterior linea de error cambiar tipo de dato a numerico
227 mc <- with(test,table(prediccion, tipo_isib))
228 mc
229 # calculo los aciertos totales:
230 err= sum(mc[1,1], mc[2,2], mc[3,3])/sum(mc)
231 err # 0.746 con 7 var T y F o T y T ; .748 para 8 y t y T
232 #faltaria completar la mc con %
233
234 # -----
235 # Grafico del error OOB en cada iteracion
236 X11()
237 tuneRF(x = entrenamiento[, -1], # data set de entrenamiento
238        y = entrenamiento$tipo_isib, # variable a predecir
239        mtryStart = 1, # cantidad de variables inicial
240        stepFactor = 2, # incremento de variables
241        ntreeTry = 1000, # cantidad arboles a ejecutar en cada iteracion
242        improve = 1 # mejora minima del OOB para seguir iteraciones
243 )
244
245 # fin forestRandom
246 #####
247 #rpart
248 #PASO 2: Crea Arbol de Decision
249 # -----
250 str(training.ids)
251 View(training.ids)
252 ModArbol<-rpart(dataf$Nom_Prov ~ ., data=dataf[4:7], parms=list(split="information"))
253 print(ModeloArbol)
254 X11()
255 rpart.plot(ModArbol) ## aqui tengo el resultado graficamente en entrenamiento
256 X11() # otra forma de graficarlo
257 rpart.plot(ModArbol, type=1, extra=100, cex = .7,
258            box.col=c("gray99", "gray88")[ModArbol$frame$yval])

```