APRENDIZAJE AUTOMÁTICO EN R STUDIO CLOUD

CLASIFICACIÓN SOCIOECONÓMICA – ESPACIAL DE HOGARES DEL SISBEN POR UPZ EN BOGOTÁ

SUBSECRETARÍA DE INFORMACIÓN
Y ESTUDIOS ESTRATÉGICOS
Dirección de Estudios Macro

ALCALDÍA MAYOR DE BOGOTÁ D.C.
SECRETARÍA DE PLANEACIÓN
BOGOTÁ

Alcaldesa Mayor de Bogotá
Claudia Nayibe López Hernández

Secretaria Distrital de Planeación
Adriana Córdoba Alvarado

Subsecretario de Información y Estudios Estratégicos
Antonio José Avendaño Arosemena

Directora de Estudios Macro
Daniela Pérez Otavo

Investigador
Edwin Alberto Cuevas Chaves

Equipo de la Dirección de Estudios Macro
Adriana Roa Pineda
Camilo Gaitán Victoria
Diana Cuellar Orjuela
Henry Rincón Melo
Nelson Chaparro Escobar
Vanessa Cediel Sanchez
Diego Buelvas Ramírez
Antonio Villalobos Rubiano
Zoraida Galindo Quintero

Diseño portada
Liliana Andrade Fernández
Oficina Asesora de Prensa y Comunicaciones
Foto: Archivo Secretaria Distrital de Planeación
Recursos gráficos: freepik.com

Noviembre de 2020

Contenido

Lista de Tablas

Lista de Gráficas

1. INRODUCCIÓN

Este ejercicio de análisis exploratorio de Data Science hace parte del trabajo de implementación de técnicas de Big Data y Machine Learning para la solución de problemas que requieren ayudas tecnológicas de avanzada enfocadas en inteligencia artificial.

Particularmente en este ejercicio, mediante el entrenamiento de varios modelos predictivos ML, se explora la capacidad de un algoritmo robusto de las librerías de R Studio para clasificar espacialmente hogares del Sisben Bogotá por Unidad de Planeamiento Zonal – UPZ, a través de la selección de variables socioeconómicas determinantes de la calidad de vida de esta población.

2. Objetivo

Clasificar a la población Sisben de la localidad Usaquén de Bogotá en la Unidad de Planeamiento Zonal – UPZ donde reside el hogar, a partir de un conjunto de características socioeconómicas determinantes de la calidad de vida de los hogares.

3. Requerimiento técnico

La clasificación que se desea hacer es por UPZ para los hogares de la localidad de Usaquén en Bogotá, para lo cual se trabajará sobre el software R Studio Cloud, ya que este soporta perfectamente el volumen de la data a utilizar y suministra los modelos de *machine learning* – ML requeridos para el entrenamiento de aprendizaje supervisado, y las pruebas necesarias para validar la presión y así resolver el problema.

Se utilizará un porcentaje del conjunto de datos para entrenar la máquina y se reservará otro porcentaje menor de la data para los procesos de validación para 5 modelos diferentes ML, en busca del modelo óptimo.

Se presentará el paso a paso del trabajo y seguidamente los resultados obtenidos y las debidas interpretaciones. La programación final con código R hará parte de los anexos técnicos de este documento.

4. Descripción del sistema de información

El sistema de información que se utiliza para el desarrollo de este ejercicio corresponde a los registros administrativos de potenciales beneficiarios de programas sociales - Sisben, (población del Sisben), agregado por hogares que viven en las UPZs de la localidad de Usaquén en Bogotá.

La recolección de la información Sisben se hace en dos momentos:

1) Consolidación de la base de datos inicial por barrido censal a hogares ubicados en las manzanas de estratos 1, 2 y parte del 3.

2) Encuestas a la demanda, esto es, la Secretaría Distrital de Planeación – SDP realiza una encuesta socioeconómica estandarizada al hogar que la solicita.

Las bases de datos del Sisben (vivienda, hogares y personas) se encuentran en archivo plano tipo **.csv, .sav, .dat** para SPSS, SAS, STATA, R, Python, entre otros, junto con los documentos metodológicos asociados a la encuesta y son custodiados por la SDP dada la sensibilidad que se maneja con información de personas; sin embargo se disponen al público cifras

agregadas y estadísticas anonimizadas en la página de la Secretaría Distrital de Planeación - SDP: www.sdp.gov.co.

5. Detección y Corrección de la data

Selección de Variables

Las variables de la data seleccionadas para este ejercicio de clasificación son las que se listan a continuación:

código_hogar
vivienda: tipo de vivienda
estrato: estrato de la vivienda
pared: material de las paredes
piso: material de los pisos
servicios públicos:
- energia
- alcanta
- gas
- basura
- acueduc

tcuartosvi: número de cuartos de la vivienda
thogar: total hogares en la vivienda
teneviv: forma de tenencia de la vivienda
tcuartos: número de cuartos de la vivienda que usa el hogar
tdormir: número de cuartos de la vivienda que usa el hogar para dormir
tsanitar: número de baños
preparan: existencia de cocina
lavadora: posesión de máquina lavadora
tpersona: número de personas del hogar
sexo: sexo del jefe
nivel: nivel escolar del jefe
edad: edad del jefe
ingresos: ingresos mensuales del hogar
puntaje_sisben_3
barrio
upz

Es de aclarar que se hacen los análisis del ejercicio de esta práctica para los 3.397 registros de hogares urbanos de Usaquén de la base de datos original del Sisben Bogotá seleccionada.

Corrección de datos

El ejercicio de obtención y limpieza se hace como parte del alistamiento de la data en el ambiente de trabajo de R Studio_Cloud.

Como parte de preparación de la data se eliminan variables del data-frame original relacionadas con la ubicación e identificación de los sujetos.

Se verificó la correspondencia de la upz a los registros de localidad, se eliminan los que están errados y cuyo registro UPZ está fuera de la localidad.

No se hacen imputaciones a la variable ingreso, por considerar que cuando se tienen ceros significa que el hogar no reporta ingresos económicos, situación que es válida en la metodología de la encesta Sisben.

6. Ejecución

Paso 1: Preprocesamiento

6.1 Carga del data set (conjunto de datos)

Los datos de la base de datos certificada con corte seleccionado fueron los que se utilizaron para este análisis. Este conjunto de datos contiene 3.397 observaciones de hogares registrados en Sisbén de la localidad Usaquén de Bogotá. La data contiene 26 columnas de medidas de las variables más relevantes para la clasificación socioeconómica de las personas. La última columna es la UPZ donde vive el hogar observado. Todos los hogares observados pertenecen a una de las siguientes UPZ que conforman la localidad:

01 - Paseo de los Libertadores
09 - Verbenal
10 - La Uribe
11 - San Cristóbal Norte
12 - Toberín
13 - Los Cedros
14 - Usaquén
15 - Country Club
16 - Santa Bárbara

Para iniciar, se debe adjuntar la data al entorno de trabajo de R-Studio Cloud.

6.2 Paquete Caret

El paquete de intercalación en R Studio se utilizó para construir los modelos. Este paquete proporciona una interfaz consistente en cientos de algoritmos de aprendizaje automático y proporciona métodos prácticos útiles para la visualización de datos, remuestreo de datos, ajuste de modelos y comparación de modelos, entre otras características. Es una herramienta imprescindible para proyectos de aprendizaje automático en R. Se debe descargar e instalar la librería asociada a este paquete.

Paso 2: Creación del conjunto de datos para entrenamiento y validación

Para esto, de dividió el conjunto de datos del dataset cargado en dos partes así:

- 80% utilizado para entrenar a los modelos ML
- 20% retenido como un conjunto de datos para la validación

El reparto se soporta en la importancia de hacer una validación ya que es fundamental para determinar si los modelos construidos son buenos.

Después de usar métodos estadísticos para estimar la precisión de los modelos creados con datos no vistos, se requirió una estimación de precisión más concreta del mejor modelo con datos no vistos, evaluándolo con datos no vistos reales en el conjunto de validación. Es decir, se reservó un subconjunto de datos que los algoritmos no vieron (el conjunto de validación) y se usaron esos datos para obtener una segunda idea independiente de cuán preciso sería realmente el mejor modelo.

Paso 3: Resumen del conjunto de datos

En este apartado se verificaron algunas características estadísticas fundamentales del dataset, como las siguientes:

1. Dimensiones del conjunto de datos

2. Tipos de atributos

3. Vista a los datos

4. Niveles del atributo de clase

5. Desglose de las instancias de cada clase

6. Resumen estadístico de todos los atributos

Ahora se detallarán los resultados obtenidos del trabajo de cada procesamiento.

6.3 Dimensiones del conjunto de datos

Se verifica que el tamaño de la data usada para entrenamiento está efectivamente constituido por 2.721 registros (filas) y 26 variables (columnas).

6.4 Tipos de atributos

A continuación, se identificaron los atributos particulares de los datos, es decir cómo están dadas las variables (tipos). Conocer los tipos es importante, ya que esto da una idea de cómo resumir mejor los datos y qué transformaciones se podrían necesitar aplicar para preparar los datos antes de modelarlos. Tabla 1.

Tabla 1. Tipología de las variables

cod_hog	vivienda	pared	piso
"numeric"	"numeric"	"numeric"	"numeric"
energia	alcanta	gas	basura
"numeric"	"numeric"	"numeric"	"numeric"
acueduc	estrato	tcuartosvi	thogar
"numeric"	"numeric"	"numeric"	"numeric"
teneviv	tcuartos	tdormir	tsanitar
"numeric"	"numeric"	"numeric"	"numeric"
preparan	lavadora	tpersona	sexo
"numeric"	"numeric"	"numeric"	"numeric"
nivel	ingresos	edad	puntaje_sisben_3
"numeric"	"numeric"	"numeric"	"numeric"
barrio	upz		
"numeric"	"factor"		

6.5 Vista de los datos

Se dio una mirada a las primeras líneas de la data, para tener una mejor idea de cómo se ven los datos.

Tabla 2. Cabeza de la data

	cod_hog	vivienda	pared	piso	energia	alcanta	gas	basura	acueduc	estra
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<d
1	1	1	4	3	1	1	2	1	1	
2	2	1	1	3	1	1	2	1	1	
3	3	1	1	1	1	2	2	2	2	
4	4	1	1	2	1	2	2	1	1	
5	6	1	1	3	1	1	2	1	1	
6	7	1	1	3	1	1	2	1	2	

6.6 Niveles de las variables factoriales

La UPZ es la única variable de tipo característica, pero se necesitó convertirla en factor, así que ahora es la única factorial. Con esto, fue necesario profundizar más en el contenido de R para hacer la transformación de manera adecuada e identificar sus niveles.

La variable UPZ cuenta con 9 niveles, por lo que este es un problema de clasificación de clases múltiples o multinomiales. Pero, si por ejemplo, solo existieran dos niveles, habría sido un problema de clasificación binaria.

Tabla 3. Niveles de la variable UPZ

```
[1] "barbara"     "cedros"      "country"     "cristobal"    "libertad
[6] "toberin"     "uribe"       "usaquen"     "verbenal"
```

6.7 Distribución de clases

Luego se determinó el número de registros que pertenecen a cada clase de UPZ como un recuento absoluto y como un porcentaje. Se debe tener en cuenta que las clases no tenían el mismo número de registros, es decir, no hay un reparto en proporciones iguales para cada UPZ.

Tabla 4. Distribución de clases

	freq	percentage
barbara	20	0.7350239
cedros	303	11.1356119
country	15	0.5512679
cristobal	696	25.5788313
libertadores	6	0.2205072
toberin	374	13.7449467
uribe	393	14.4432194
usaquen	383	14.0757075
verbenal	531	19.5148842

6.8 Resumen estadístico

Ver tabla 5.

Tabla 5. Resumen estadístico

cod_hog	vivienda	pared	piso	energia
Min. : 1	Min. :1.000	Min. :1.00	Min. :1.00	Min. :1.000
1st Qu.: 846	1st Qu.:1.000	1st Qu.:1.00	1st Qu.:2.00	1st Qu.:1.000
Median :1693	Median :1.000	Median :1.00	Median :2.00	Median :1.000
Mean :1699	Mean :1.218	Mean :1.13	Mean :2.27	Mean :1.001
3rd Qu.:2553	3rd Qu.:1.000	3rd Qu.:1.00	3rd Qu.:3.00	3rd Qu.:1.000
Max. :3397	Max. :2.000	Max. :7.00	Max. :6.00	Max. :2.000

alcanta	gas	basura	acueduc	estrato
Min. :1.00	Min. :1.000	Min. :1.000	Min. :1.000	Min. :0.000
1st Qu.:1.00	1st Qu.:1.000	1st Qu.:1.000	1st Qu.:1.000	1st Qu.:1.000
Median :1.00	Median :1.000	Median :1.000	Median :1.000	Median :2.000
Mean :1.01	Mean :1.251	Mean :1.001	Mean :1.004	Mean :2.198
3rd Qu.:1.00	3rd Qu.:2.000	3rd Qu.:1.000	3rd Qu.:1.000	3rd Qu.:3.000
Max. :2.00	Max. :2.000	Max. :2.000	Max. :2.000	Max. :6.000

tcuartosvi	thogar	teneviv	tcuartos	tdormir
Min. : 1.000	Min. :1.000	Min. :1.000	Min. : 0.000	Min. :0.000
1st Qu.: 2.000	1st Qu.:1.000	1st Qu.:1.000	1st Qu.: 1.000	1st Qu.:1.000
Median : 3.000	Median :1.000	Median :1.000	Median : 2.000	Median :2.000
Mean : 2.937	Mean :1.206	Mean :1.802	Mean : 2.511	Mean :1.802
3rd Qu.: 4.000	3rd Qu.:1.000	3rd Qu.:3.000	3rd Qu.: 4.000	3rd Qu.:2.000
Max. :12.000	Max. :5.000	Max. :4.000	Max. :12.000	Max. :7.000

tsanitar	preparan	lavadora	tpersona	sexo
Min. :0.000	Min. :0.0000	Min. :1.000	Min. : 1.000	Min. :1.00
1st Qu.:1.000	1st Qu.:1.0000	1st Qu.:1.000	1st Qu.: 2.000	1st Qu.:1.00
Median :1.000	Median :1.0000	Median :2.000	Median : 3.000	Median :1.00
Mean :1.173	Mean :0.9875	Mean :1.507	Mean : 3.439	Mean :1.48
3rd Qu.:1.000	3rd Qu.:1.0000	3rd Qu.:2.000	3rd Qu.: 4.000	3rd Qu.:2.00
Max. :4.000	Max. :2.0000	Max. :2.000	Max. :13.000	Max. :2.00

nivel	ingresos	edad	puntaje_sisben_3
Min. :0.000	Min. : 0	Min. :17.00	Min. : 16
1st Qu.:1.000	1st Qu.: 300000	1st Qu.:32.00	1st Qu.:2624
Median :2.000	Median : 515000	Median :43.00	Median :4477
Mean :1.818	Mean : 548518	Mean :44.48	Mean :4248
3rd Qu.:2.000	3rd Qu.: 644350	3rd Qu.:56.00	3rd Qu.:5985
Max. :5.000	Max. :11300000	Max. :96.00	Max. :8571

barrio	upz
Min. : 8401	cristobal:696
1st Qu.: 8516	verbenal :531
Median : 8529	uribe :393
Mean : 14125	usaquen :383
3rd Qu.: 8543	toberin :374
Max. :208203	cedros :303
	(Other) : 41

Paso 4: Visualización del conjunto de datos

Para visualizar gráficos, primero es necesario instalar un par de librerías del repositorio de R (ggplot2 y dplyr).

Después de tener una idea básica de los datos, se amplió esa comprensión con algunas visualizaciones:

1. Gráficos univariados para comprender mejor cada atributo
2. Gráficos multivariados para comprender mejor las relaciones entre atributos

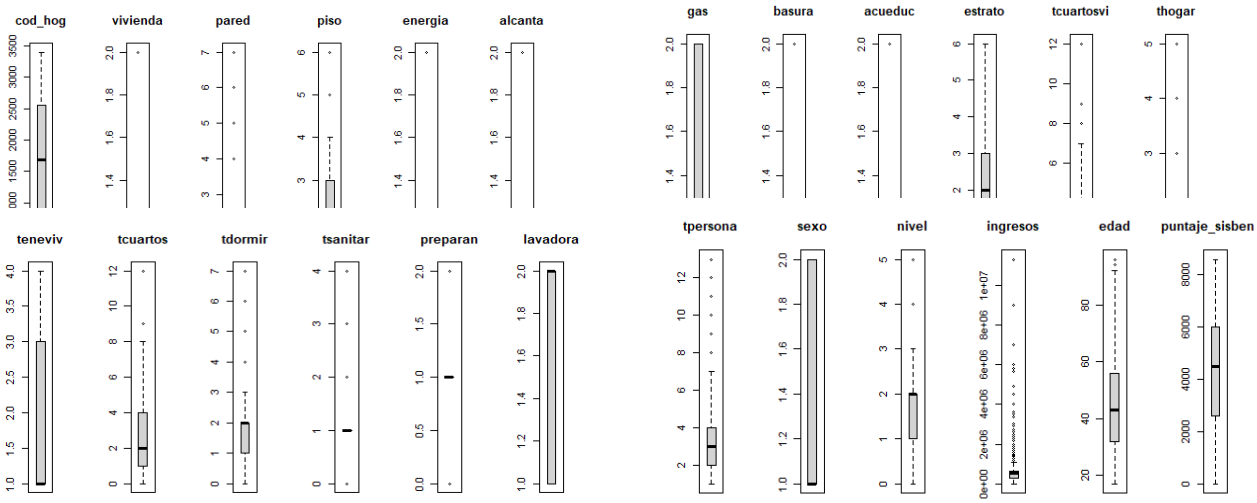
6.9 Gráficos univariados

Se comenzó con gráficos univariados (es decir, gráficos de cada variable individual).

Con la visualización, es útil tener una forma de hacer referencia solo a los atributos de entrada y solo a los atributos de salida. Así que se configuraron variables llamando a los atributos de entrada x y al atributo de salida (o clase de UPZ) y.

Ahora bien, dado que las variables de entrada no son todas numéricas, de los diagramas de cajas y bigotes mostrados en la gráfica 1, solamente resultan importantes para revisión los que tienen asociada este tipo de variable y aportan información para ver la distribución de cada atributo. Se sugiere al lector prestar atención al total de cuartos de la vivienda, total de cuartos que dispone el hogar, total de cuartos usados para dormir, total sanitarios, total personas, nivel académico, edad y puntaje Sisbén.

Gráfica 1. Diagramas de caja



La caja de la variable *tcuartosvi* indica que las unidades de vivienda habitadas por la población de estudio se concentran alrededor de 2 a 4 cuartos, siendo el valor mediano 2 cuartos que como se muestra en la caja de la variable *tcuartos*, estos 2 últimos son de uso exclusivo del hogar encuestado, sin embargo, para dormir no se usan ambos cuartos, como lo señala la caja de la variable *tdormir,* lo que comienza a generar problemas de hacinamiento que sumado al reducido número de sanitarios disponibles para uso del hogar que en promedio es 1 son indicadores de condiciones habitacionales deficitarias.

Por otra parte, estos hogares se componen por grupos de entre 1 y 7 personas, estando el valor mediano alrededor de 3 personas (caja de la variable *tpersona*). Estos hogares tienen una jefatura ejercida por una persona de 41 años promedio cuyo máximo nivel de escolaridad es secundaria (cajas *edad* y *nivel*). El puntaje sisben 3 mediano de estos hogares es de 44,77.

Así las cosas, es posible concluir que las cifras que se muestran en los diagramas de cajas son indicadores reales generalizados del nivel de pobreza de esta población y elementos válidos se selección para el ejercicio de clasificación que se aborda en este estudio.

6.10 Gráficos multivariados

Después de graficar cada atributo individual, se exploraron las interacciones entre las variables observando diagramas de dispersión de todos los pares de atributos con puntos coloreados por clase.

Luego se volvió a los gráficos de caja y bigotes para cada variable de entrada, pero dividiéndolos en gráficos separados para cada clase de UPZ. Esto porque ayuda a desentrañar separaciones lineales obvias entre las clases y por supuesto que hay distribuciones claramente diferentes de los atributos para cada clase de UPZ.

La distribución de cada atributo se exploró más a fondo con gráficos de densidad de probabilidad. Nuevamente, al igual que los diagramas de caja y bigotes anteriores, los diagramas de densidad de probabilidad se desglosaron por clase de UPZ para obtener líneas suaves para cada distribución. Al igual que los diagramas de caja, la diferencia en la distribución de cada atributo por clase fue evidente. También se revisó la distribución de tipo gaussiano de cada atributo.

Algunas visualizaciones por pares de variables (bivariadas) se presentan en el anexo a este documento.

Paso 5: Evaluación del algoritmo

A continuación, se crearon modelos de los datos y se calculó su precisión en datos invisibles. Este fue un proceso de tres pasos:

1. Configuración del arnés de prueba para utilizar una validación cruzada de 10 veces.
2. Construcción de 5 modelos diferentes para predecir la UPZ a partir de las variables de entrada.
3. Selección del mejor modelo.

6.11 Prueba de arnés

Se utilizó una validación cruzada de 10 veces para estimar la precisión. Esto dividió el conjunto de datos en 10 partes (entrenar en 9 y probar en 1) y luego se publicó para todas las combinaciones de divisiones de prueba de tren. El proceso se repitió 3 veces para cada uno de los 5 algoritmos, con diferentes divisiones de los datos en 10 grupos para obtener estimaciones más precisas.

Como se mencionó, se utilizó la métrica de ***Precisión*** para evaluar los modelos. Esta es una proporción del número de instancias predichas correctamente dividido por el número total de instancias en el conjunto de datos multiplicado por 100 para dar un porcentaje (por ejemplo, 95% de precisión).

Ahora se deben instalar las siguientes librerías adicionales necesarias para entrar a las construcciones de los modelos:

- ***E1071***
- ***Kernlab***
- ***Random Forest***

6.12 Construcción de modelos

Inicialmente se desconocía qué algoritmos funcionarían bien en este problema o qué configuraciones usar. Los gráficos sugirieron que algunas de las clases son parcialmente separables linealmente en algunas dimensiones, por lo que en general se esperaban buenos resultados.

Se evaluaron cinco algoritmos diferentes:

1. Análisis discriminante lineal (LDA)
2. Árboles de clasificación y regresión (CART)
3. k-Vecinos más cercanos (kNN)
4. Admite máquinas vectoriales (SVM) con un núcleo lineal
5. Bosque aleatorio (RF)

Esta fue una buena combinación de métodos lineales simples (LDA), no lineales (CART, kNN) y complejos no lineales (SVM, RF). La semilla de número aleatorio se restableció antes de cada ejecución para garantizar que la evaluación de cada algoritmo se realizara utilizando exactamente las mismas divisiones de datos y los resultados fueran directamente comparables.

Los cinco modelos se construyeron y se guardaron como variables en el espacio de trabajo de R Studio Cloud.

- Algoritmos lineales: LDA
- Algoritmos no lineales: CART y kNN
- Algoritmos avanzados: SVM y RF

6.13 Selección del mejor modelo

Una vez que se crearon los cinco modelos y las estimaciones de precisión para cada uno, la siguiente tarea consistió en comparar los modelos y seleccionar el más preciso.

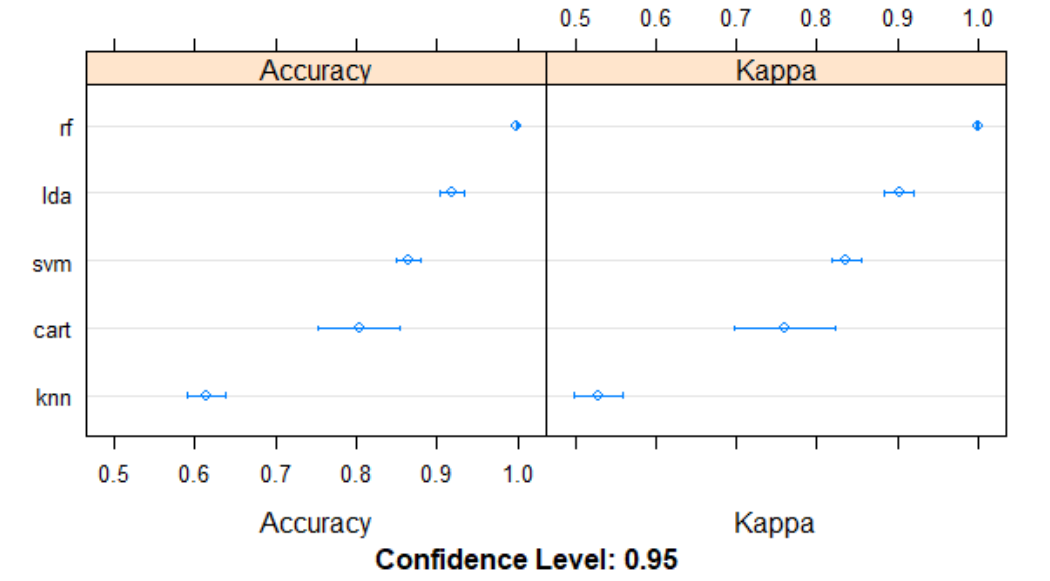
Para hacer esto, se creó una lista de los modelos ajustados y se pasaron estos resultados a la función de resumen para obtener una salida que muestra la precisión de cada clasificador junto con otras métricas, como Kappa.

Tabla 6. Resuman de precisión

```
Call:
summary.resamples(object = results)

Models: lda, cart, knn, svm, rf
Number of resamples: 10

Accuracy
          Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
lda  0.8868613 0.9104414 0.9208134 0.9191350 0.9283088 0.9525547    0
cart 0.7335766 0.7357298 0.8035512 0.8034608 0.8708487 0.8750000    0
knn  0.5620438 0.5891544 0.6143927 0.6141370 0.6297901 0.6752768    0
svm  0.8175182 0.8580591 0.8713235 0.8658709 0.8784518 0.8901099    0
rf   0.9963504 1.0000000 1.0000000 0.9996350 1.0000000 1.0000000    0

Kappa
          Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
lda  0.8631325 0.8912553 0.9039902 0.9019713 0.9130437 0.9425020    0
cart 0.6748911 0.6770209 0.7598314 0.7599363 0.8424209 0.8473524    0
knn  0.4629557 0.4982882 0.5284947 0.5290889 0.5512326 0.6033926    0
svm  0.7772647 0.8269065 0.8427713 0.8362793 0.8514432 0.8659421    0
rf   0.9955780 1.0000000 1.0000000 0.9995578 1.0000000 1.0000000    0
```

Luego se generó un gráfico de los resultados de la evaluación del modelo y se comparó la dispersión, así como la precisión media de cada modelo. Es importante tener en cuenta que existe una población de medidas de precisión para cada algoritmo porque cada algoritmo se evaluó 10 veces (validación cruzada de 10 veces), por lo que las estimaciones de precisión media tuvieron que compararse.

Gráfica 2. Precisión de los modelos



Como se observa en la gráfica anterior, el modelo más preciso para el ejercicio de clasificación requerido de los hogares de Sisben en Usaquén por UPZ fue el Análisis de Discriminante Lineal - LDA, dado que posee la precisión media más alta con la dispersión más pequeña.

Ahora bien, dado que LDA se identificó como el mejor modelo de clasificación, se resumieron los resultados solo para LDA. El resultado brindó un buen resumen de lo que se usó para entrenar el modelo y la precisión media y de desviación estándar lograda, específicamente 97.5% de precisión +/- 4%.

Tabla 7. Resumen modelo óptimo

```
Linear Discriminant Analysis

2721 samples
  25 predictor
   9 classes: 'barbara', 'cedros', 'country', 'cristobal', 'libertadores', 'tober
 'uribe', 'usaquen', 'verbenal'

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 2449, 2451, 2447, 2447, 2450, 2450, ...
Resampling results:

  Accuracy  Kappa
  0.919135  0.9019713
```

Paso 6: Predicciones

Como se evidenció, LDA resulto ser el modelo más preciso en el conjunto de entrenamiento, sin embargo, se debe determinar la precisión del modelo en el conjunto de validación para obtener una verificación final independiente sobre la precisión del mejor modelo: para ello se mantuvo un conjunto de validación solo en caso de sobreajuste al conjunto de entrenamiento o una fuga de datos, ya que ambos habrían resultado en un resultado demasiado optimista.

El modelo LDA se ejecutó directamente en el conjunto de validación y los resultados se resumieron en una matriz de confusión. La precisión fue del 100%.

Es importante recordar que el data set para validación era un pequeño conjunto correspondiente al 20% de la data original, pero este resultado estaba dentro del margen

esperado de 97% +/- 4%, lo que sugiere que LDA definitivamente es un modelo preciso y confiable.

Tabla 8. Matriz de confusión

```
Confusion Matrix and Statistics

              Reference
Prediction     barbara cedros country cristobal libertadores toberin uribe usaquen
  barbara            2      0       1         0             0       0     0       1
  cedros             0     48       0         0             0       8     0       6
  country            2      0       1         0             0       0     0       0
  cristobal          0      0       0       166             0       0     7       0
  libertadores       0      0       0         2             0       0     1       0
  toberin            0      5       0         0             0      85     0       0
  uribe              0      0       0         6             0       0    90       0
  usaquen            1     22       1         0             0       0     0      88
  verbenal           0      0       0         0             1       0     0       0
              Reference
Prediction     verbenal
  barbara              0
  cedros               0
  country              0
  cristobal            0
  libertadores         1
  toberin              0
  uribe                7
  usaquen              0
  verbenal           124

Overall Statistics

               Accuracy : 0.8935
                 95% CI : (0.8678, 0.9157)
    No Information Rate : 0.2574
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.8708
```

La matriz de confusión deja ver como LDA predice con muy buena exactitud la clasificación del hogar en la UPZ correcta a través de las variables de entrada consideradas en el estudio.

Para terminar con la revisión de las predicciones, se mostrará en la tabla 9 las estadísticas resultantes para cada UPZ.

Tabla 9. Estadísticas por clase para modelo óptimo

```
 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: barbara Class: cedros Class: country Class: cristobal
Sensitivity                0.400000       0.64000       0.333333           0.9540
Specificity                0.997019       0.97671       0.997028           0.9861
Pos Pred Value             0.500000       0.77419       0.333333           0.9595
Neg Pred Value             0.995536       0.95603       0.997028           0.9841
Prevalence                 0.007396       0.11095       0.004438           0.2574
Detection Rate             0.002959       0.07101       0.001479           0.2456
Detection Prevalence       0.005917       0.09172       0.004438           0.2559
Balanced Accuracy          0.698510       0.80835       0.665181           0.9700
                     Class: libertadores Class: toberin Class: uribe Class: usaquen
Sensitivity                     0.000000         0.9140       0.9184          0.9263
Specificity                     0.994074         0.9914       0.9775          0.9587
Pos Pred Value                  0.000000         0.9444       0.8738          0.7857
Neg Pred Value                  0.998512         0.9863       0.9860          0.9876
Prevalence                      0.001479         0.1376       0.1450          0.1405
Detection Rate                  0.000000         0.1257       0.1331          0.1302
Detection Prevalence            0.005917         0.1331       0.1524          0.1657
Balanced Accuracy               0.497037         0.9527       0.9479          0.9425
                     Class: verbenal
Sensitivity                   0.9394
Specificity                   0.9982
Pos Pred Value                0.9920
Neg Pred Value                0.9855
Prevalence                    0.1953
Detection Rate                0.1834
Detection Prevalence          0.1849
Balanced Accuracy             0.9688
```

Paso 7: Visualizaciones

Para no mostrar visualizaciones estáticas, las visualizaciones de los resultados más relevantes e interesantes se encuentran en el informe de Power BI anexo. Se comparte el archivo web mediante link:

https://app.powerbi.com/groups/me/dashboards/5fa592ab-1d6b-4b10-a36a-b1d2acd7a14c?ctid=25a51937-6bbd-469d-9e25-d53cbb4bf3a2

7. CONCLUSIONES

Al final del ejercicio fue posible concluir que para resolver el problema de clasificación de los hogares del Sisben por UPZ mediante inteligencia artificial, resulta muy apropiado el uso de la técnica de clasificación referida al tipo de aprendizaje supervisado de Machine Learning, haciendo uso del sofware R Studio Cloud, sin embargo, es recomendable explorar el ejercicio sobre R Studio para determinar cuál escala mejor la data y logra trabajar más rápido en términos de procesamiento.

El modelo basado en Análisis de Discriminante Lineal – LDA conserva las características del mejor modelo de clasificación para este ejercicio de clasificación, incluso superando a Random Forest en el valor de la media de precisión a la vez que mostró ser el de menor variabilidad.

Para próximas versiones del ejercicio es importante probar una disminución de la cantidad de la data destinada para el entrenamiento, se sugiere un 70% a 75%, esto con el fin de reducir la probabilidad de sobre entrenamiento de los modelos.

ANEXO

CLASIFICACIÓN SOCIOECONÓMICA – ESPACIAL DE HOGARES DEL SISBEN POR UPZ EN BOGOTÁ

CÓDIGO DE PROGRAMACIÓN EN R STUDIO

```
#cargar la data
data (usaq_catalu)
#renombrar data set
dataset <- usaq_catalu

# Cargar paquete
install.packages("caret")
library (caret)


# 80% de registros del conjunto de datos original para entrenamiento
validation_index<-createDataPartition(dataset$upz, p =0.80, list = FALSE)

# Selección del 20% de los datos para validación
validation<-dataset[-validation_index, ]

# 80% restante de los datos para entrenar y probar los modelos
dataset<-dataset[validation_index, ]


# dimensión del data set

dim(dataset)


# Tipos de lista para cada atributo
sapply(dataset,class)


# Ver las primeras 6 filas de los datos
head(dataset)
```

```
# niveles de variable upz
levels(dataset$upz)

# Resumen de la distribución de clases
percentage<-prop.table(table(dataset$upz)) * 100
cbind(freq = table(dataset$upz), percentage = percentage)

# Resumen de distribuciones de atributos
summary(dataset)

install.packages("ggplot2")
library(ggplot2)
install.packages("dplyr")
library(dplyr)

# Entrada y salida divididas
x<-dataset[,1:24]
y<-dataset[ ,26]

# Diagrama de caja para cada atributo
par(mfrow=c(1,6))
for(i in 1:24) {boxplot(x[,i], main=names(usaq_catalu)[i])}

# matriz de diagrama de dispersión
featurePlot (x = x, y = y, plot = "elipse")

# matriz de diagrama de dispersión
featurePlot (x = x, y = y, plot = "elipse")

# Gráficos de caja y bigotes para cada atributo
```

```
featurePlot (x = x, y = y, plot = "caja")

# Gráficas de densidad para cada atributo por valor de clase de upz
escalas <-lista (x = lista (relación = "libre"), y = lista (relación = "libre"))
featurePlot (x = x, y = y, plot = "densidad", escalas = escalas)

# Ejecutar algoritmos usando una validación cruzada de 10 veces control
control<-trainControl(method = "cv", number = 10)
metric<-"Accuracy"

install.packages("e1071")
library(e1071)
install.packages("kernlab")
install.packages("randomForest")
library(kernlab)
library(randomForest)

# Análisis discriminante lineal (LDA)
set.seed(7)
fit.lda <- train(upz~., data=dataset, method="lda", metric=metric, trControl=control)
# Árboles de clasificación y regresión (CART)
set.seed(7)
fit.cart <- train(upz~., data=dataset, method="rpart", metric=metric, trControl=control)

# k-Vecinos más cercanas (kNN)
set.seed(7)
fit.knn <- train(upz~., data=dataset, method="knn", metric=metric, trControl=control)

# Máquinas de vectores de soporte (SVM)
set.seed(7)
```

```
fit.svm <- train(upz~., data=dataset, method="svmRadial", metric=metric, trControl=control)

# Bosque aleatorio (RF)
set.seed(7)
fit.rf <- train(upz~., data=dataset, method="rf", metric=metric, trControl=control)
# k-Vecinos más cercanos (kNN)


# Resumen de la precisión del modelo para cada modelo
results <- resamples(list(lda=fit.lda,cart=fit.cart, knn=fit.knn, svm=fit.svm, rf=fit.rf))
summary(results)


# Compación grafica de la precisión de los modelos

dotplot(results)

# Resumen del mejor modelo (LDA)
print(fit.lda)


# Estimar la habilidad de LDA en el conjunto de datos de validación
predictions<-predict(fit.lda, validation)
confusionMatrix(predictions, validation$upz)
```