COMPARACION DE SISTEMAS DE

ATRACCION-GENERACION

GRUPO 10

Joao Paulo Scabora | Lucas Ezequiel | Julen Larranaga | Lucia Lopez | Juan Carlos Rubio

INDICE

I. INTRODUCCION Y OBJETIVOS

El objetivo principal de este proyecto es establecer una comparacion entre dos tipos de modelos (SVR y Redes neuronales) para predecir los viajes diarios atraidos y generados por una zona concreta.

Para el analisis se han utilizado datos economico-demograficos de un total de 256 zonas diferentes. Los datos provistos en el enunciado del proyecto han sido generados sinteticamente para poder realizar el ejercicio academico. Asi, estos datos incluyen informacion detallada sobre la poblacion, como la distribucion por edades y el porcentaje de personas en diferentes rangos de ingresos. Ademas, tambien se disponen datos sobre la superficie de areas especificas destinadas a diferentes usos (residencial, comercial e industrial). Por ultimo, se incluyen datos actuales de poblacion y de atraccion y generacion de viajes.

Se parte de la hipotesis de que estos datos proporcionan una base suficiente para entender las caracteristicas socioeconomicas y demograficas de cada zona, y relacionarlas con sus valores de atraccion y generacion de viajes.

Con este objetivo, el proyecto se ha dividido en varias fases.

La primera fase consiste en la **limpieza de los datos**. Esto incluye la carga y limpieza inicial, para poder trabajar con los datos. Ademas, se lleva a cabo la imputacion de datos faltantes para asegurar la integridad del analisis.

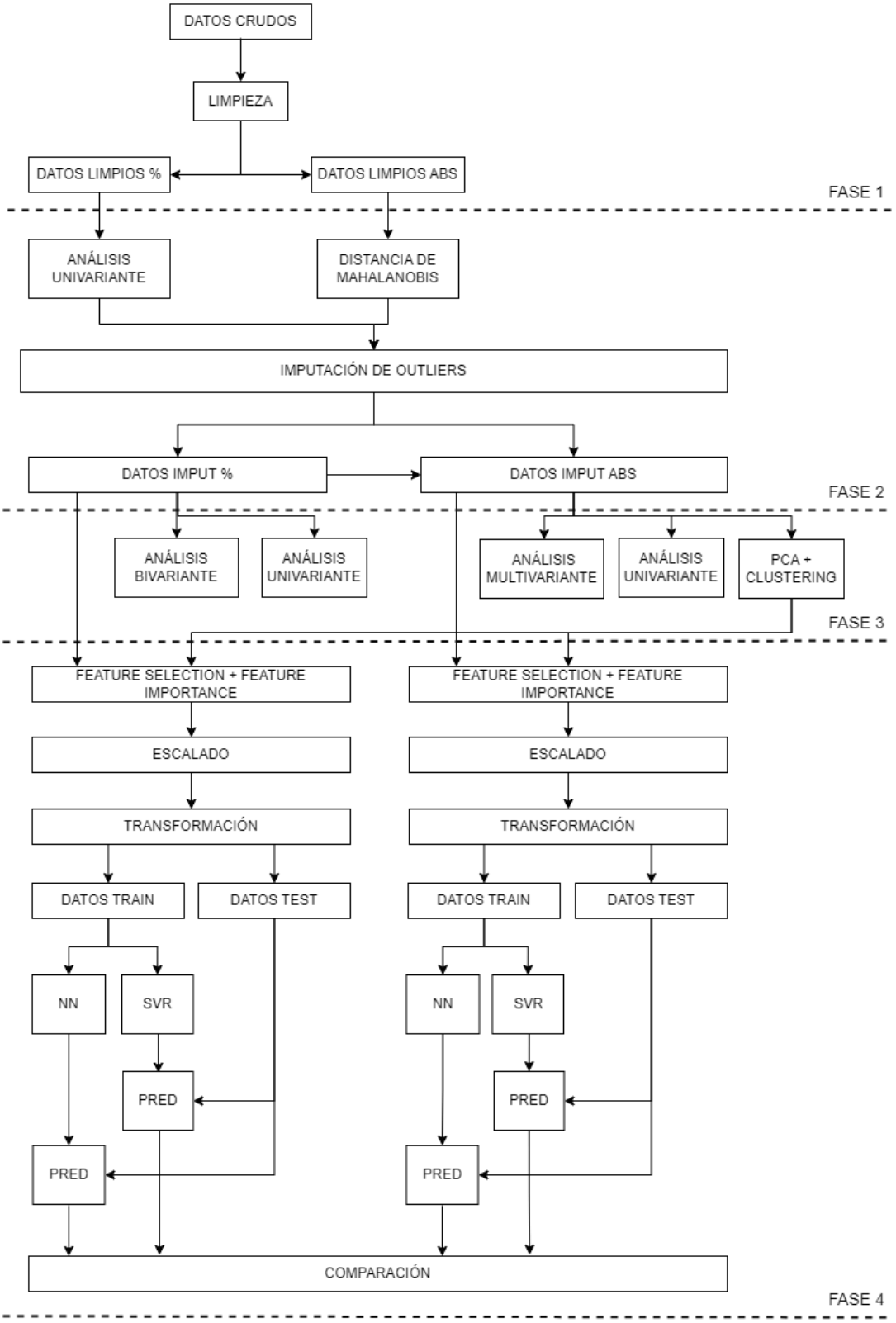
La segunda fase procede con el **analisis univariante y multivariante para la deteccion de outliers**. Aqui, por un lado se examina cada variable individualmente para entender su distribucion y caracteristicas. Por otro lado, se utiliza el concepto de la distancia de Mahalanobis para identificar outliers multivariantes entre los datos. A continuacion, se evaluan los distintos analisis de forma conjunta y se decide la eliminacion e imputacion de los valores asumidos como outliers.

En la tercera fase se vuelve a realizar los analisis univariante y multivariante con los datos imputados con el objetivo de comprobar su correcto tratamiento. Igualmente, para una mejor interpretacion de los datos, se hace un **analisis bivariante y un Analisis de Componentes Principales (PCA).** Asimismo, se agrupan los datos en **clusters** para posteriormente valorar si es conveniente anadirlo a la ingesta de los modelos y mejorar su desempeno.

Finalmente, en la cuarta fase, se construyen, entrenan y optimizan los modelos de **Redes Neuronales (NN)** y metodos de **Maquinas de Vectores de Soporte (SVR)** para predecir la atraccion y generacion de movimiento en un

lugar. Las predicciones obtenidas mediante los dos metodos se comparan con los datos test para ver su precision y efectividad y determinar cual proporciona mejores resultados.

Los resultados de este analisis pueden ser utilizados para mejorar la planificacion urbana y la toma de decisiones en politicas publicas, basandose en predicciones mas precisas sobre la atraccion y generacion de movimiento en diferentes areas.


DATOS CRUDOS
LIMPIEZA
DATOS LIMPIOS %
DATOS LIMPIOS ABS
FASE 1
ANÁLISIS UNIVARIANTE
DISTANCIA DE MAHALANOBIS
IMPUTACIÓN DE OUTLIERS
DATOS IMPUT %
DATOS IMPUT ABS
FASE 2
ANÁLISIS BIVARIANTE
ANÁLISIS UNIVARIANTE
ANÁLISIS MULTIVARIANTE
ANÁLISIS UNIVARIANTE
PCA + CLUSTERING
FASE 3
FEATURE SELECTION + FEATURE IMPORTANCE
FEATURE SELECTION + FEATURE IMPORTANCE
ESCALADO
ESCALADO
TRANSFORMACIÓN
TRANSFORMACIÓN
DATOS TRAIN
DATOS TEST
DATOS TRAIN
DATOS TEST
NN
SVR
NN
SVR
PRED
PRED
PRED
PRED
COMPARACIÓN
FASE 4

II. LIMPIEZA DE DATOS

A. DATOS

El conjunto de datos original es un archivo en formato de texto (.txt) que contiene 256 filas de datos mas una fila de encabezado, definida por un espacio y una tabulacion. Los datos estan distribuidos en 14 columnas, incluida una columna de identificacion (nz).

Los datos incluyen informacion sobre poblacion y superficie, analizados segun tres caracteristicas: renta, grupo de edad y uso de la superficie. Los porcentajes de personas segun su renta y edad pertenecen a la parte relacionada con la poblacion, mientras que los porcentajes de tipos de superficie se relacionan con la superficie total. Cada una de estas tres caracteristicas se presenta en grupos de tres fracciones, donde la suma de un triplete debe ser a 1.

nz	Pob	Ralta	Rmedia	Rbaja	PirPob25	PirPob65	PirPobJ	SupTOT	supSR	supC	supI	A	G
1	12356	0.354	0.1103	0.5357	0.2135	0.6154	0.1711	8.17	0.9161	0.0696	0.0144	119310	121922
2	13198	0.4309	0.2903	0.2788	0.2655	0.5852	0.1492	7.861	0.8559	0.0864	0.0577	150816	148812
3	11983	0.571	0.3933	0.0357	0.2695	0.5752	0.1553	7.956	0.8547	0.0797	0.0656	125061	120782
4	9748	0.3679	0.3796	0.2525	0.2336	0.634	0.1324	7.783	0.9064	0.0767	0.0169	130876	132352
5	7803	0.0695	0.6726	0.2578	0.2522	0.597	0.1507	6.825	0.8972	0.0599	0.0429	68918	69780
6	6763	0.3628	0.3068	0.3305	0.2132	0.6338	0.153	8.161	0.8411	0.0802	0.0787	113002	115793

B. MODIFICACIONES

El primer paso consiste en convertir las columnas inicialmente dispuestas en formato de texto o con comas en lugar de puntos decimales a **formato numerico**. Esto es importante ya que las operaciones matematicas y estadisticas no pueden realizarse correctamente en columnas que no esten en formato numerico.

Tras una primera ojeada al contenido del archivo se observa que hay 8 , representados con asteriscos.

nz	Pob	Ralta	Rmedia	Rbaja	PirPob25	PirPob65	PirPobJ	SupTOT	supSR	supC	supI	A	G
29	13905	0.676	0.1269	******	0.259	0.6004	0.1406	9.202	0.8046	0.1222	0.0732	97962	98231
53	30574	******	0.3693	0.57	0.2527	0.5833	0.164	8.38	0.6846	0.2404	0.075	154867	152287
74	30801	0.1209	0.2687	0.6104	0.466	0.6055	******	8.772	0.6216	0.3398	0.0386	197305	198600
122	39941	0.1597	0.4474	0.3929	0.263	0.5772	0.1598	11.252	0.6938	0.2638	******	42823	42657
130	15614	0.3427	0.4058	******	0.6199	0.6097	0.1705	7.753	0.6475	0.2802	0.0723	55215	57086

229	1101	0.1667	0.3333	******	0.2587	0.5769	0.165	3.571	0.0391	0.4698	0.4911	36758	37284
234	1464	******	0.2975	0.407	0.2564	0.5908	0.1528	2.604	0.026	0.2752	0.6988	49863	51037
246	941	0.3103	0.2759	******	0.2427	0.6072	0.1501	2.31	0.0544	0.3949	0.5506	6657	7024

Estos valores faltantes se han calculado teniendo en cuenta los grupos de porcentajes a los que pertenecen. Por ejemplo, si *Ralta* es el dato ausente, se ha imputado utilizando el complementario a la suma de *Rmedia* y *Rbaja* para que el total sea 1.

$$Si\ Ralta = NA,\ entonces\ Ralta = 1 - (Rmedia + Rbaja)$$

Este paso es crıtico para garantizar que no haya huecos en los datos que puedan sesgar los resultados del analisis. Para ello, se han creado nuevas columnas nombradas (*% + #nombre columna original*) que a continuacion se insertan en sustitucion de las existentes con datos faltantes.

nz	Ralta	% Ralta	Rbaja	% Rbaja	PirPobJ	% PirPobJ	supl	% supl
29	0.676	0.676	******	0.1971	0.1406	0.1406	0.0732	0.0732
53	******	0.0607	0.57	0.57	0.164	0.164	0.075	0.075
74	0.1209	0.1209	0.6104	0.6104	******	-0.0715	0.0386	0.0386
122	0.1597	0.1597	0.3929	0.3929	0.1598	0.1598	******	0.0424
130	0.3427	0.3427	******	0.2515	0.1705	0.1705	0.0723	0.0723
229	0.1667	0.1667	******	0.5	0.165	0.165	0.4911	0.4911
234	******	0.2955	0.407	0.407	0.1528	0.1528	0.6988	0.6988
246	0.3103	0.3103	******	0.4138	0.1501	0.1501	0.5506	0.5506

El siguiente paso consiste en la deteccion e imputacion de **datos erroneos**. Estos son aquellos valores que son negativos o pertenecientes a grupos de fracciones cuya suma fuese diferente a uno. Para ello se han creado dos tipos de parametros de salida llamados OUT comprobando que:

- OUT1: las sumas de grupos de fracciones dan 1
- OUT2: todos los valores de fraccion son no negativos (≥ 0)

Un parametro de salida igual a 1 indica que los datos estan correctos. En cambio, un parametro de salida igual a 0 denota la presencia de datos erroneos. Al desconocer el procedimiento de toma de datos y considerando el repetitivo e incierto proceso que puede requerir un analisis siguiendo la casuıstica de cada uno de los errores, se ha

optado por aplicar un mismo tratamiento general a todos ellos. El procedimiento cuando existe un fallo en **Out1** y/u **Out2** es el indicado a continuacion.

- Eliminacion del conjunto de fracciones
- Imputacion de todos los valores usando MICE
- Reajuste para que la suma sea 1

Se muestran todos los datos erroneos en la tabla a continuacion:

FILA	INDICADOR OUT	GRUPO DE DATOS	PROBLEMA
11	OUT1P	Edad/Poblacion	Las fracciones de edad/poblacion suman 1,004
74	OUT2P2565	Edad/Poblacion	La fraccion de la poblacion entre 25 y 65 anos es negativa
113	OUT1S	Uso de Superficie	Las fracciones de superficie suman 1,4
130	OUT1P	Edad/Poblacion	Las fracciones de edad/poblacion suman 1,4
145	OUT1R	Renta	Las fracciones de renta suman 1,089
191	OUT1R	Renta	Las fracciones de renta suman 1,1
231	OUT1P	Edad/Poblacion	Las fracciones de edad/poblacion suman 1,07
240	OUT1R	Renta	Las fracciones de renta suman 1,0081

El **metodo de imputacion** empleado es "Predictive Mean Matching" (pmm), que en lugar de imputar directamente los valores faltantes utilizando el modelo de regresion, pmm selecciona individuos observados en los datos que tienen valores similares.

C. EXPORTACION DE DATOS

Finalmente se cambio el nombre a las columnas para facilitar el entendimiento en los analisis posteriores:

nz → nz

Pob → pob

Ralta	→	r_alta
Rmedia	→	r_media
Rbaja	→	r_baja
PirPob25	→	pob_25
PirPob65	→	pob_2565
PirPobJ	→	pob_65
SupTOT	→	sup_tot
supSR	→	sr_sup
supC	→	c_sup
supI	→	i_sup
A	→	atraccion
G	→	generacion

Hay que tener en cuenta que, por un lado, los datos en fracciones y los datos absolutos tienen interpretaciones fisicas distintas. Asi, un *pob_25* en fraccion indica la presencia de jovenes en el distrito respecto a la poblacion del mismo e independientemente de cual sea esta, mientras que *pob_25* en terminos absolutos denota simplemente el numero total de jovenes en el distrito. El uso de datos en multiples dimensiones con valores en rangos diferentes en ordenes de magnitud puede alterar el analisis multivariante y el funcionamiento de los modelos. En consecuencia, se han generado **dos dataset** de partida, uno manteniendo las columnas en **fracciones** y otro convirtiendolas en valores **absolutos**. Por tanto, de ahora en adelante se dispone de:

- datos_porcentajes
- datos_absolutos

Los valores absolutos se han calculado multiplicando los porcentajes de los grupos de datos con los valores a los que hacen referencia (Poblacion/Superficie), y los datos obtenidos para las caracteristicas relacionadas con la poblacion se han redondeado a su valor entero mas cercano.

III. OUTLIERS

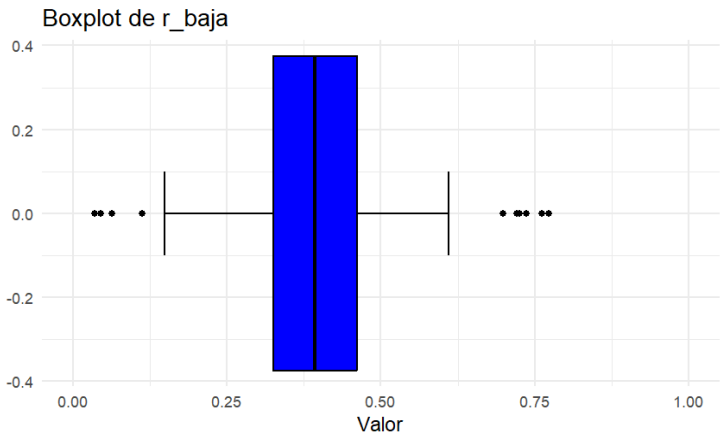
A. ANALISIS UNIVARIANTE

El analisis univariante se enfoca en examinar cada una de las variables para comprender su distribucion y detectar posibles outliers. En este apartado en concreto solo se emplea para la deteccion de posibles outliers, dejando para mas adelante el analisis y descripcion de la distribucion de cada variable.

La deteccion de outliers se realiza principalmente con la herramienta conocida como boxplot. Los datos introducidos en los boxplots son los datos en fracciones (*datos_porcentajes*) y se evaluan mediante el paquete *skimr*. Los resultados obtenidos para las diferentes caracteristicas se muestran a continuacion, donde se ha optado por omitir los datos sin relevancia interpretativa.

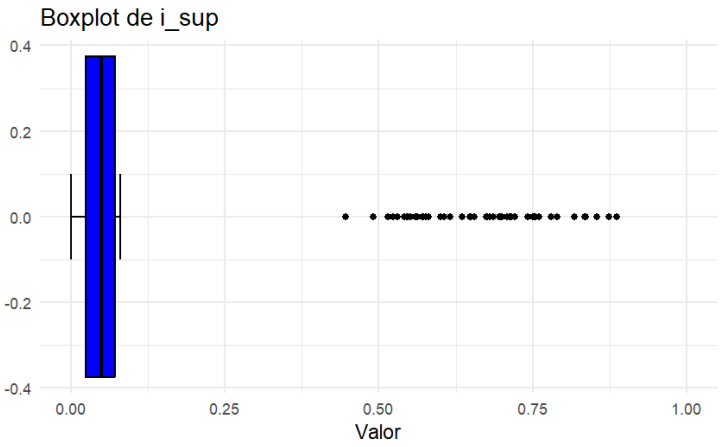
RENTA

En el analisis de la variable *r_baja*, se pueden ver algunos outliers. Estos valores, a pesar de poder ser identificados como outliers, se asume que son valores *reales* que contienen informacion sensible sobre los datos.

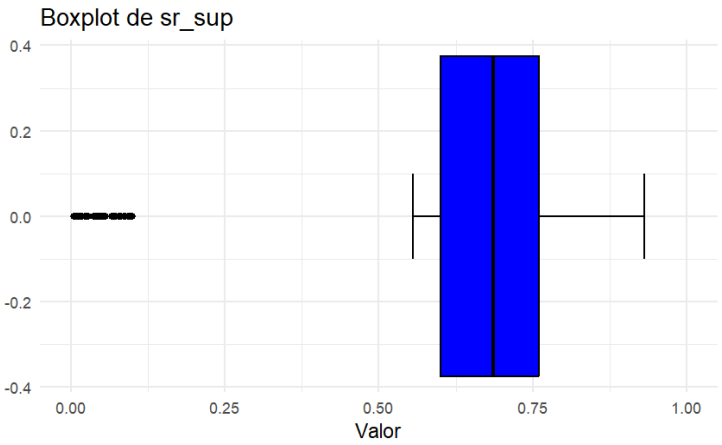


USO DE LA SUPERFICIE

La variable del porcentaje de uso industrial de la superficie, *i_sup,* revela que una gran cantidad de distritos tienen un caracter predominantemente industrial.



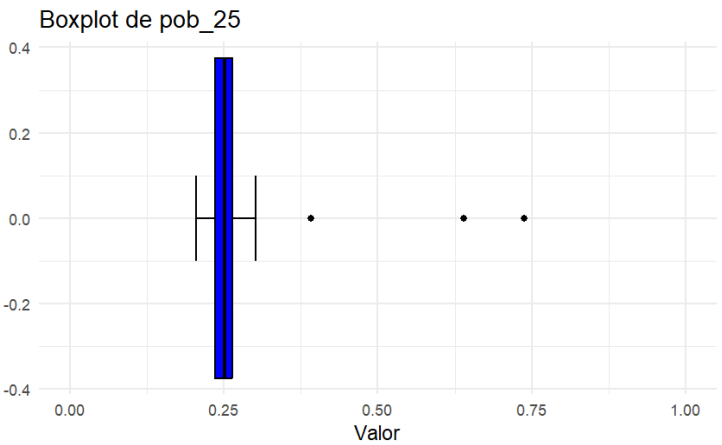
Este hallazgo es corroborado por las fracciones residenciales y de servicio *sr_sup*.



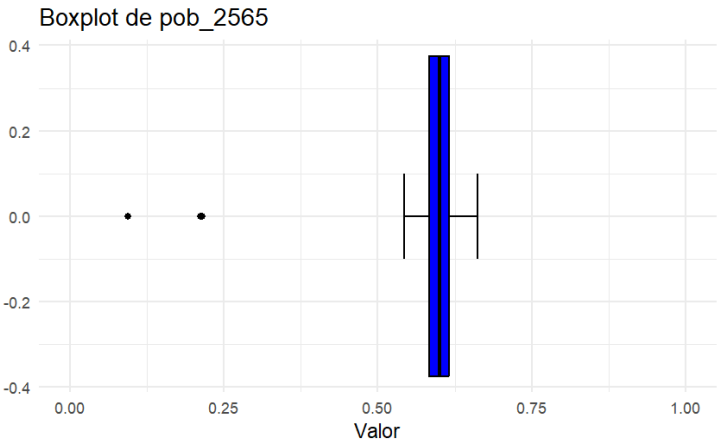
En cuanto a la distribucion comercial, no se observan valores extremos destacados.

DISTRIBUCION POR EDADES

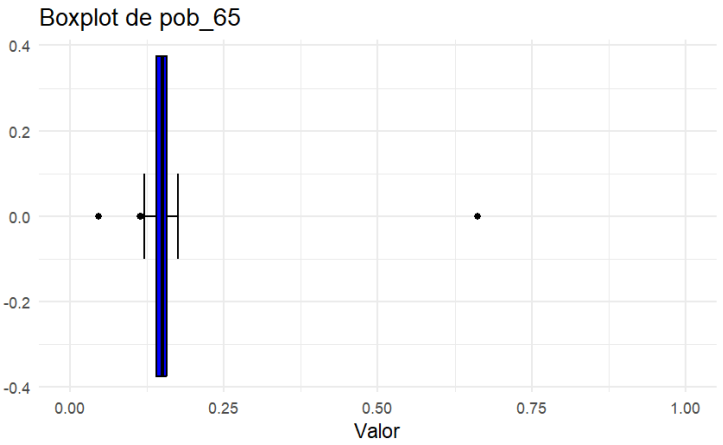
La fraccion de poblacion inferior a 25 anos *pob_25* muestra que tres distritos tienen valores considerablemente superiores a lo normal. Los distritos en cuestion son los correspondientes a las filas 35, 45 y 74.



Evaluando la variable porcentual de poblacion entre 25 y 65 anos, *pob_2565*, se observan tres valores fuera de lo comun que corresponden a las filas 18, 35 y 45.



Por ultimo, la fraccion de la poblacion mayor a 65, pob_65, muestra datos llamativos en las filas 35, 74, 94 y 18.



Las variables objetivo, atraccion y generacion, no presentaban outliers en el analisis. Esto indica que los datos son consistentes y no contienen valores atıpicos que puedan distorsionar los resultados del modelo. Asimismo, los datos de superficie total y poblacion total no presentan outliers, lo que asegura la calidad y la fiabilidad de estos datos para su uso en el analisis y modelado posterior.

B. DISTANCIA DE MAHALANOBIS

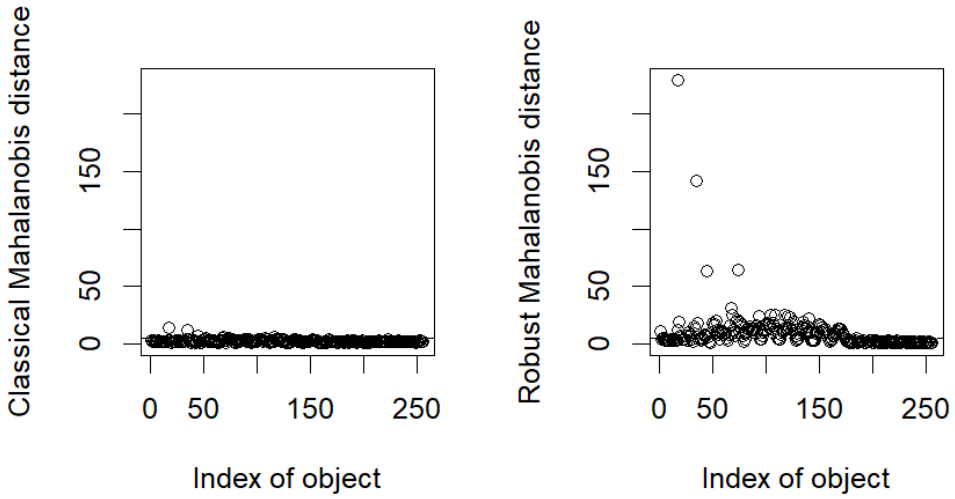
La distancia de Mahalanobis es una medida estadıstica que mide la distancia entre un punto y un conjunto de puntos en un espacio multidimensional, teniendo en cuenta la estructura de covarianza de los datos.

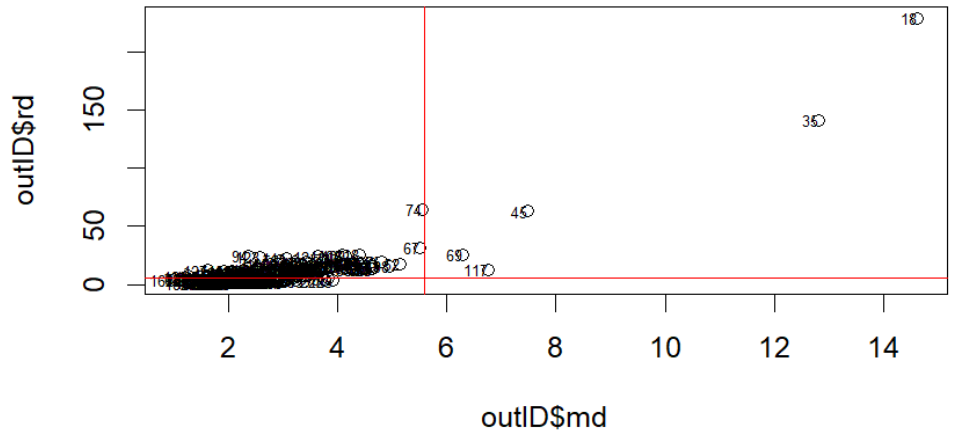
Para identificar outliers en un conjunto de datos utilizando la distancia de Mahalanobis en R, primero se calcula esta distancia para cada observacion respecto al centroide del conjunto de datos, utilizando la media y la matriz de covarianza. Luego, se establece un umbral basado en la distribucion chi-cuadrado, que determina cuan lejos debe estar una observacion para ser considerada un outlier, ajustando el nivel de significancia

En este caso se ha creado un grafico de dispersion de las distancias de Mahalanobis (md) vs. las distancias de Robust Mahalanobis (rd) que permite visualizar como se distribuyen las observaciones en funcion de los dos tipos de distancias. Aquellas observaciones cuya distancia de Mahalanobis o Robust Mahalanobis supere este umbral se identifican como outliers.

Para realizar este calculo, debido al analisis conjunto de las diversas dimensiones, se ha optado por emplear los datos absolutos (*datos_absolutos*). Se han empleado todas las columnas de datos excepto 'id' y una fraccion de cada grupo, debido a que esta inclusion resulta redundante y hace que el sistema sea singular e irresoluble.

Los outliers obtenidos corresponden a las filas id: 18, 35, 45, 69 y 117 (*outliers_rows* en R).





C. ELIMINACION Y CORRECCION DE OUTLIERS

Los outliers obtenidos tras el analisis univariante y la distancia de Mahalanobis son los siguientes:

FILA (ID)	OUTLIER
18	Univariante + Mahalanobis
35	Univariante + Mahalanobis
45	Univariante + Mahalanobis
69	Mahalanobis (Cerca del límite)
74	Univariante (Cerca del límite)
94	Univariante (Cerca del límite)
117	Mahalanobis (Cerca del límite)

Tras consultar con la fuente de los datos, se confirma que ciertos datos son extremos mientras que otros se encuentran cerca del límite, por lo que se decide eliminar los datos de fracciones de poblacion de las filas 18, 35 y 45 del dataset (*dataset_porcentajes_F1*).

No se han eliminado todos los outliers debido a la escasez de datos, ya que eliminar varios de estos podría suponer perdida de informacion crucial. Ademas, los outliers pueden contener informacion relevante que podría mejorar la

capacidad predictiva del modelo, ya que pueden representar casos especiales o extremos que el modelo debe aprender a manejar. En contextos reales, los valores extremos podrıan ser representativos de situaciones importantes y no deberıan ser descartados automaticamente.

Se procedio a su imputacion utilizando 'Mice' nuevamente y se ajusto la suma de las fracciones. Los valores imputados fueron los siguientes:

FILA	VALOR ORIGINAL	VALOR IMPUTADO
18	pob_25 = 0,2436 pob_2565 = 0,0951 pob_65 = 0,6613	pob_25 = 0,2663 pob_2565 = 0,5921 pob_65 = 0,1421
35	pob_25 = 0,7381 pob_2565 = 0,2148 pob_65 = 0,0471	pob_25 = 0,2688 pob_2565 = 0,5986 pob_65 = 0,1338
45	pob_25 = 0,6397 pob_2565 = 0,2122 pob_65 = 0,1481	pob_25 = 0,2148 pob_2565 = 0,6103 pob_65 = 0,1730

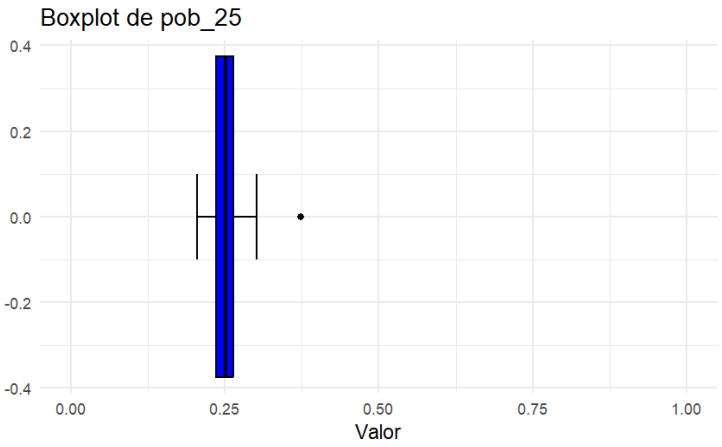
Una vez realizada la imputacion han modificado tanto los datos en fracciones (*dataset_porcentajes_F2*) como los datos absolutos (*dataset_absolutos_F2*).

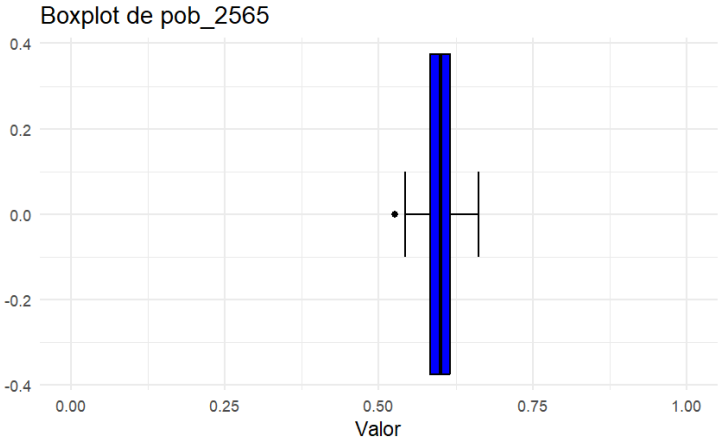
IV. ANALISIS EXPLORATORIO DE LOS DATOS

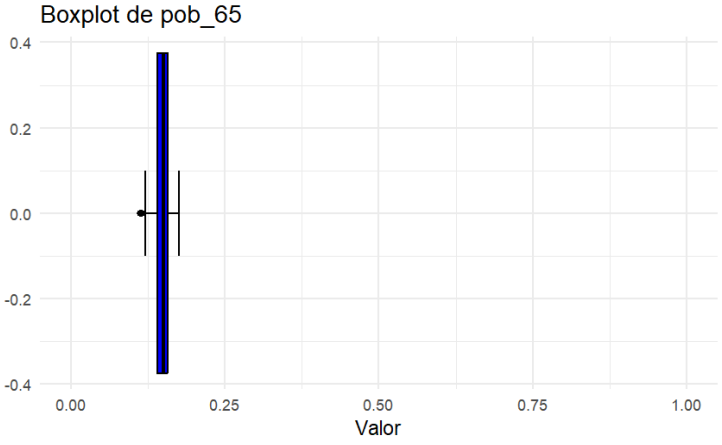
Con los datos ya imputados se realizan los diferentes analisis para tener una comprension mas completa y detallada del conjunto de datos y las relaciones entre las variables.

A. ANALISIS UNIVARIANTE

Primero, se vuelven a graficar los boxplots de los parametros fraccionales de poblacion con el objetivo de comprobar la correcta imputacion de los datos.







Una vez realizada la comprobacion, se procede a resumir los datos de los que se dispone y se emplean histogramas para visualizar la distribucion de cada variable. Estos muestran la frecuencia por rangos de los valores de la variable.

Los datos se presentan en un resumen estadistico ("summary") para proporcionar una vision general de las caracteristicas principales de los datos, como la media, mediana, desviacion estandar y rangos percentiles

Summary de datos en fracciones:

Variable	Media A	SD	Min	p25	Mediana	p75	Max	Hist
pob	17.407	14.660	95	2.002	15.064	31.058	43.416	
r_alta	0,292	0,13	0,0371	0,175	0,296	0,375	0,676	
r_media	0,321	0,104	0,0755	0,258	0,318	0,378	0,724	
r_baja	0,388	0,118	0,0357	0,326	0,393	0,462	0,773	
pob_25	0,251	0,0204	0,205	0,236	0,252	0,264	0,373	
pob_2565	0,6	0,0227	0,526	0,584	0,6	0,616	0,661	
pob_65	0.149	0,0117	0,113	0,141	0,15	0,158	0,176	
sup_tot	7,48	2,67	2,11	6,38	7,81	9,02	14,3	
sr_sup	0,595	0,274	0,0046	0,6	0,685	0,76	0,931	
c_sup	0,246	0,0931	0,0369	0,192	0,252	0,32	0,481	
i_sup	0,159	0,252	0,0002	0,0246	0,0493	0,0727	0,886	
atraccion	76.593	55.881	5.522	31.231	65.265	107.264	267.001	
generacion	76.593	55.642	5.266	31.182	66.168	106.121	261.332	

Summary de datos en fracciones:

Variable	Media A	SD	Min	p25	Mediana	p75	Max	Histogram
pob	17.407	14.660	95	2.002	15.064	31.058	43.416	
r_alta	5.256	5.463	16	468	3.694	8.174	22.914	
r_media	5.608	5.269	32	588	4.587	8.708	20.120	
r_baja	6.543	6.142	48	721	4.600	11.296	27.400	
pob_25	4.355	3.704	19	494	3.774	7.874	11.499	
pob_2565	10.461	8.816	59	1.216	9.068	18.776	26.258	
pob_65	2.593	2.203	17	302	2.174	4.503	7.089	
sup_tot	7,48	2,67	2,11	6,38	7,81	9,02	14,3	
sr_sup	4,98	2,62	0,0167	4,34	5,53	6,76	10,2	
c_sup	1,83	0,997	0,255	1,04	1,78	2,4	5,03	
i_sup	0.662	0,725	0,00153	0,204	0,415	0,645	3,22	

atraccion	76.593	55.881	5.522	31.231	65.265	107.264	267.001	
generacion	76.593	55.642	5.266	31.182	66.168	106.121	261.332	

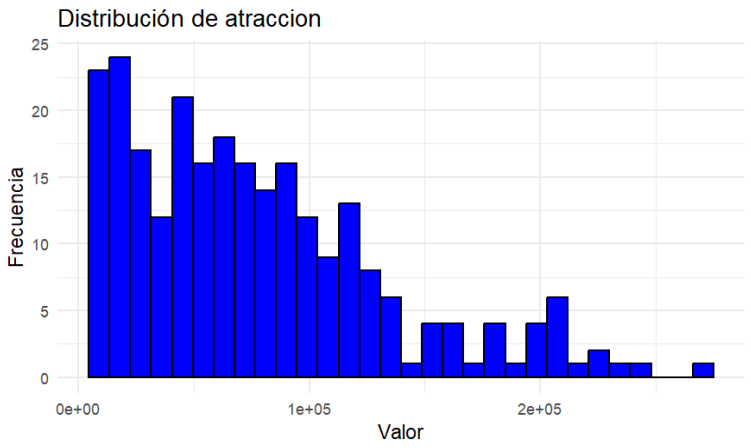
A partir de este resumen se llega a las siguientes conclusiones:

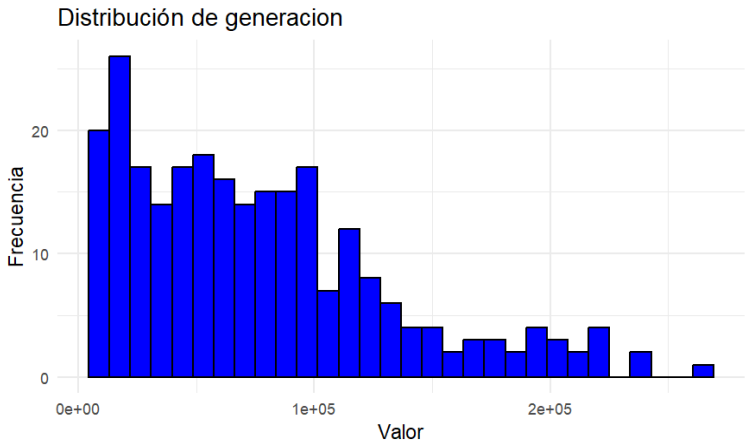
Las areas tienen una media de 17.400 habitantes, pero con una gran variabilidad (SD de 14660). El 25% de las zonas mas pequenas tiene menos de 2.000 habitantes, mientras que el 25% mas grande sobrepasa los 30.000. En cuanto a la renta de dicha poblacion, la mayoria se encuentra en el rango de renta media. Por edades, la mayoria se ubica en el rango de 25 a 65 anos.

La superficie media es de 7,48 kilometros cuadrados, y se utiliza principalmente para servicios y uso residencial. Los datos resultan en una densidad media de entorno a 2.300 habitantes por kilometro cuadrado. El dato de densidad es similar al valor de una poblacion como Tarragona.

Una vez obtenida una imagen general de los datos, seguidamente se comentan las distribuciones de los valores para las variables evaluadas.

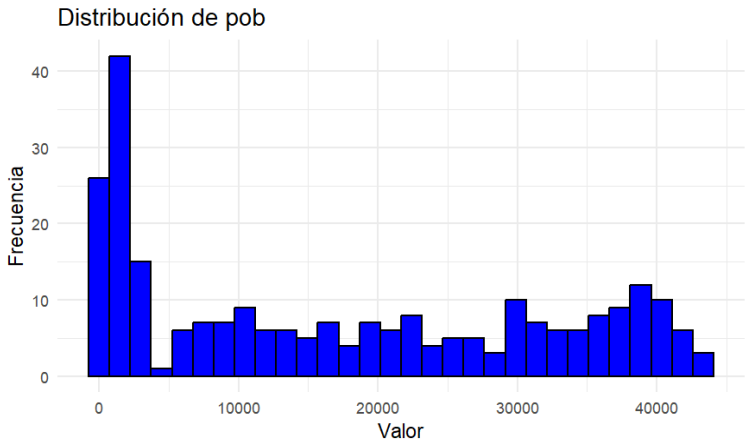
Ambas **targets (atraccion y generacion)** tienen una distribucion sesgada a la derecha.



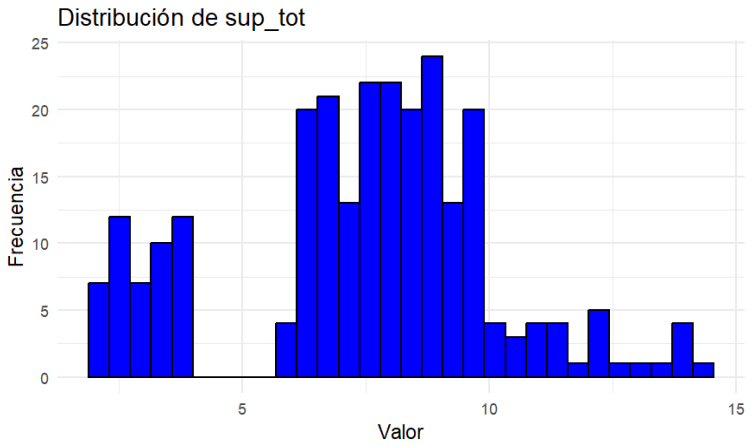


Se sugiere aplicar una transformacion logarıtmica (log1p) para reducir la asimetrıa y hacer las distribuciones mas normales, lo que puede mejorar la performance de los modelos NN y SVR. Esto podrıa ser beneficioso en el sentido de que la transformacion logarıtmica estabiliza la varianza, reduce el efecto de valores extremadamente altos y mejora la capacidad de los modelos para capturar patrones. Las desventajas son que se complica la interpretacion de los resultados y se requiere una transformacion inversa para interpretar las predicciones en la escala original.

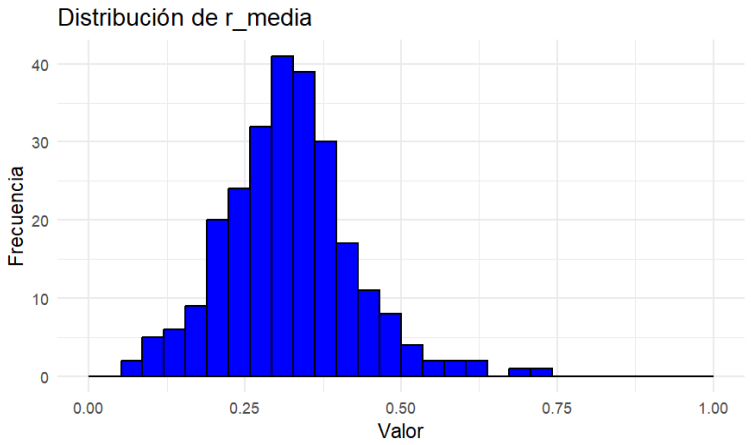
Respecto a la **poblacion total**, esta tiene tambien una distribucion sesgada hacia la derecha con valores extremos. Por lo tanto, tambien se propone una distribucion logarıtmica.

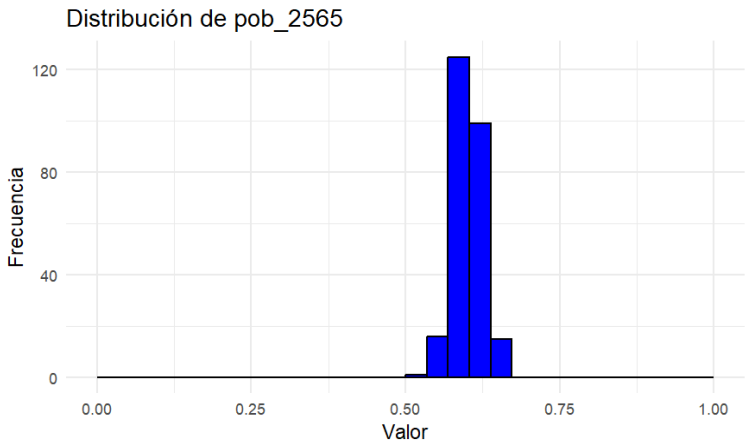


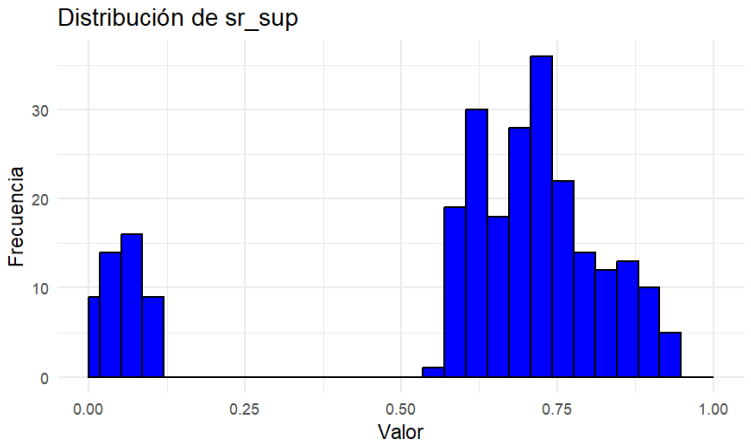
La superficie total muestra algunos valores muy bajos pero tambien otros muy altos. No se propone ningun tipo de transformacion.



Los valores de **fracciones**, en general, estan bien escalados entre 0 y 1, o el correspondiente valor mínimo y maximo. Aun así, algunas muestran ligeros sesgos. Seguidamente se procede a mostrar algunos ejemplos.







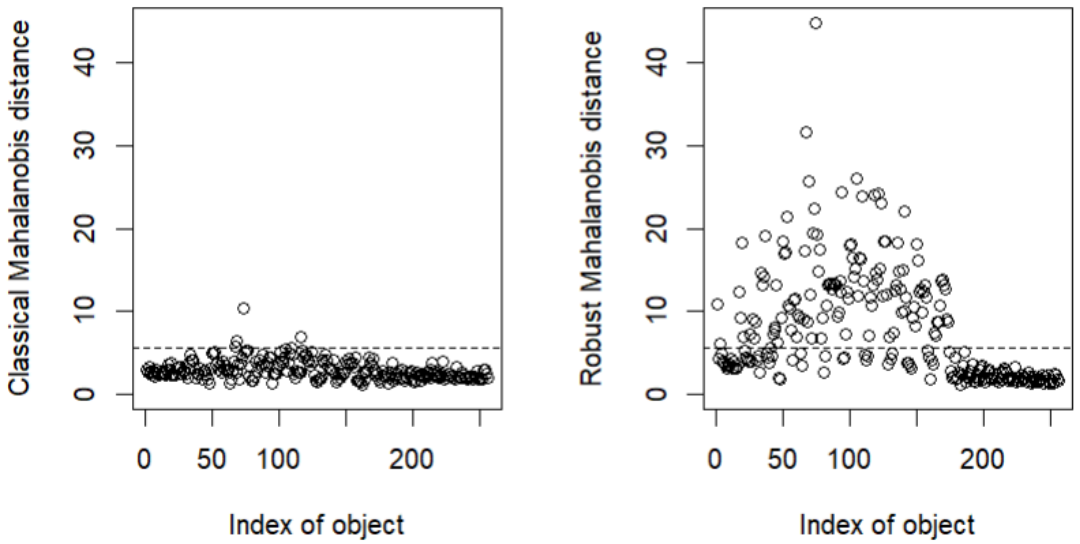
Se propone mantener las distribuciones actuales, ya que mantener las características en su forma actual simplifica el preprocesamiento y facilita la comparacion sin comprometer el rendimiento del modelo.

De este analisis previo se deduce que se podría tratar de aplicar transformaciones logarítmicas a las características sesgadas y que existe una necesidad de escalar o estandarizar los datos para asegurar que todas las características contribuyan equitativamente al modelo.

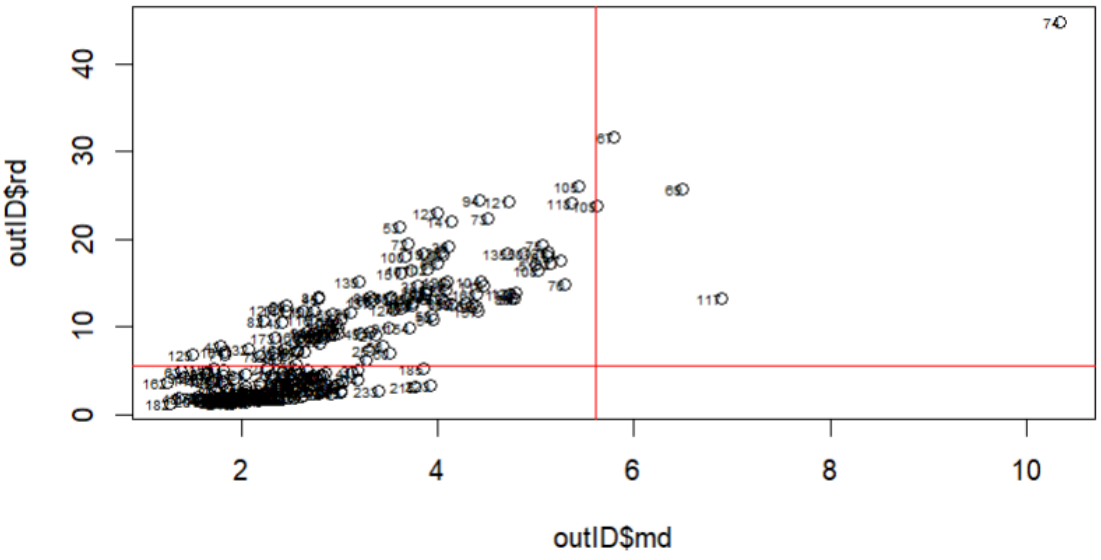
B. ANALISIS MULTIVARIANTE: MAHALANOBIS

Las distancias de Mahalanobis se revisan para verificar la eficiencia de la limpieza de datos. Al recalcular las distancias, se confirma que los outliers han sido efectivamente eliminados y detectar otros potenciales que no eran visibles antes.

El primero representa una comparacion entre la distancia de Mahalanobis para todos los datos y la distancia robusta de Mahalanobis.



La segunda grafica cuenta con dos ejes que representan las distancias de Mahalanobis (eje X) y las distancias robustas (eje Y) para cada observacion. Las lıneas rojas vertical y horizontal representan el umbral de corte para considerar una observacion como outlier. Aquellos ids (puntos) que quedan fuera de estas lıneas se consideran posibles outliers.



Como se observa, los valores de las filas 69, 74 y 111 suponen los posibles outliers, que son precisamente los descartados en la fase 2. Esta vez salen mas lejanos a la nube de datos ya que se han modificado algunos de ellos.

Para evitar dinamicas de eliminar datos de forma excesiva se ha optado por no volver a valorar posibles outliers con los datos imputados.

Por otro lado, se ve que los datos declarados outliers anteriormente han sido imputados de forma correcta, permitiendo proceder con modelos predictivos mas confiables y robustos.

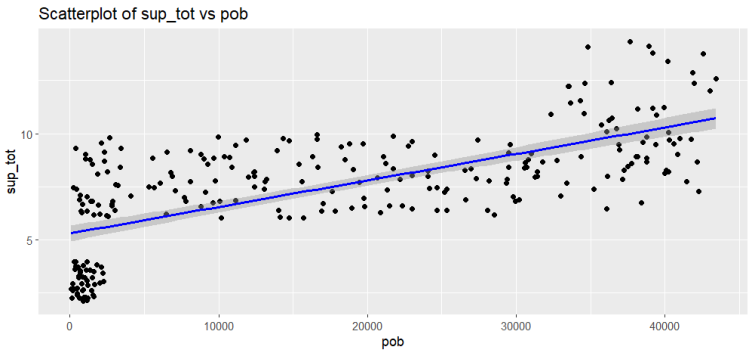
C. ANALISIS BIVARIANTE

El analisis bivariante proporciona una comprension detallada de como dos variables se relacionan entre si, tanto visualmente como cuantitativamente lo que resulta util para entender mejor las dinamicas dentro de los datos.

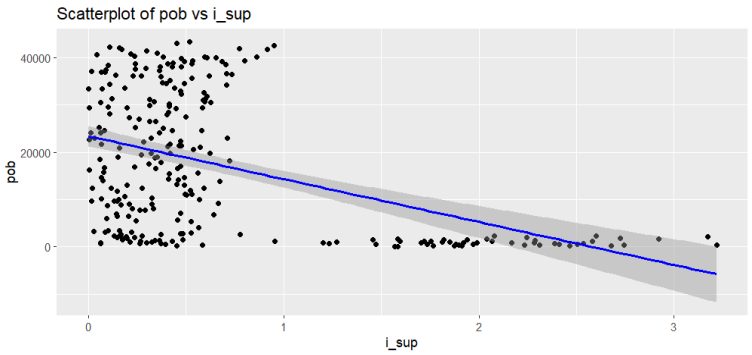
Para el analisis bivariante, se utilizan dos datasets diferentes: datos absolutos (*dataset_absolutos_F2*) para graficos de dispersion y datos fraccionados (*dataset_porcentajes_F2*) para mapas de calor de correlaciones de Pearson. Esto es debido a que los coeficientes de correlacion se relacionaban altamente entre si cuando se utilizan datos absolutos.

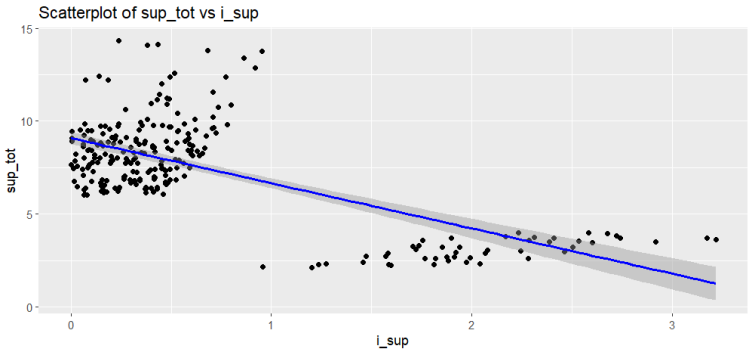
Los **graficos de dispersion** proporcionan una visualizacion clara de la relacion entre dos variables y ayudan a identificar patrones en los datos. Los graficos se realizan para las variables objetivo (atraccion y generacion) y para la poblacion y la superficie total contra todas las demas variantes.

A continuacion, se comentan las figuras mas relevantes:

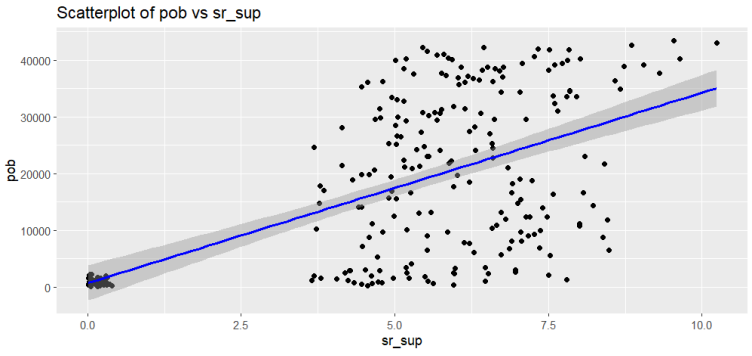


En el grafico superior se muestra una caracteristica peculiar de la base de datos: algunas zonas estan escasamente pobladas, ademas de formar un grupo diferenciado del resto, con una superficie menor en km2. Dentro del segundo grupo mas grande, tambien se observa un subgrupo caracterizado por altas poblaciones y superficies totales.



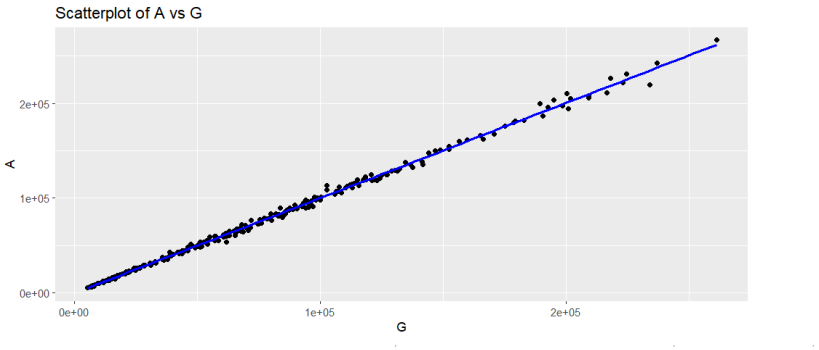


En los dos graficos anteriores, se muestra la relacion entre poblacion y superficie total frente a la superficie industrial, donde se observan dos grupos diferenciados. El grupo mas grande se caracteriza por una gran variacion en poblacion y superficie total en areas con una pequena fraccion de area industrial. El otro grupo esta compuesto por areas con una gran concentracion de km² destinado a actividad industrial, reconocidas por ser areas con baja poblacion y, en este caso, un tamano menor que el resto.

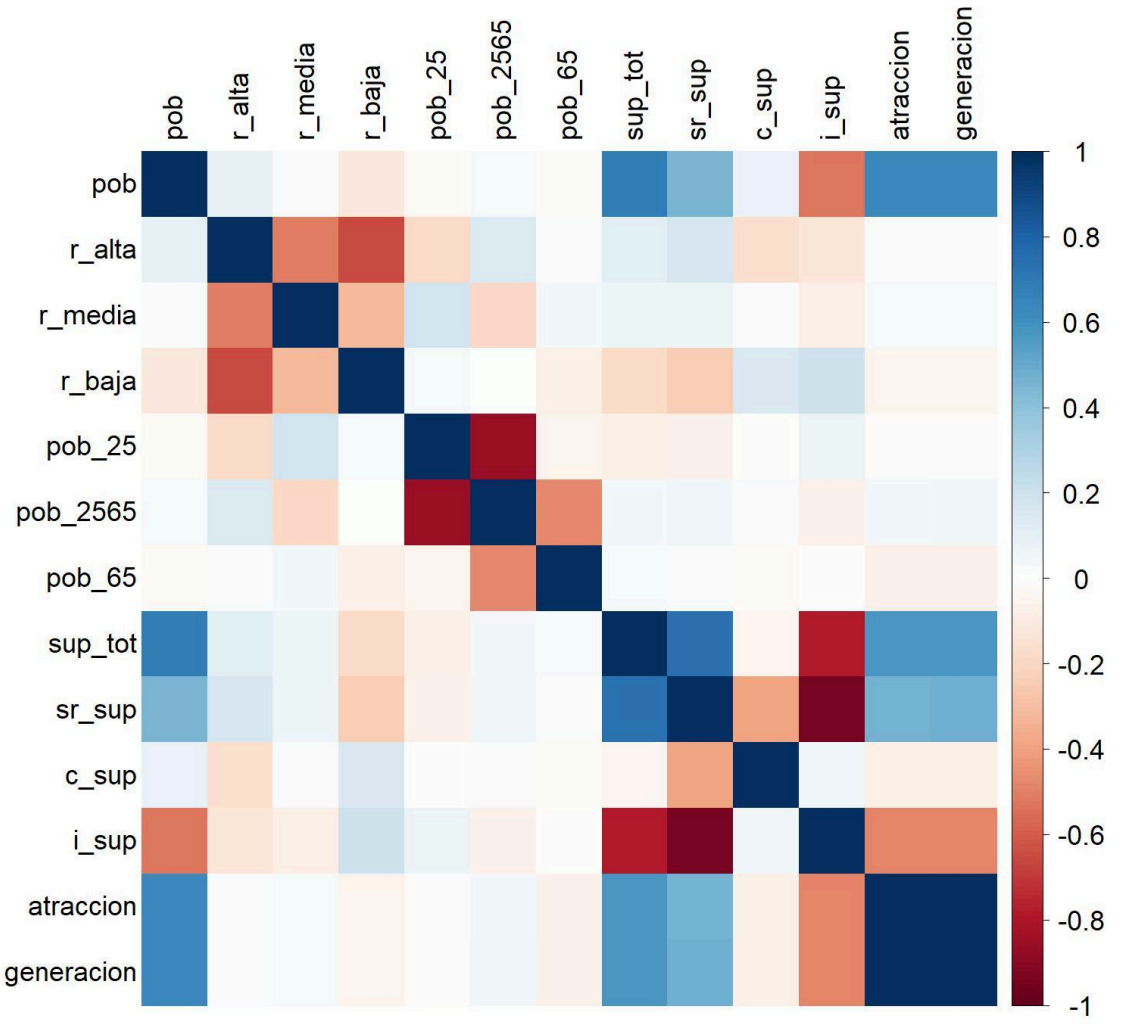


Una segunda forma de observar las areas industriales es a traves del grafico de poblacion versus areas residenciales, donde se observa el pequeno grupo de areas que se encuentran escasamente pobladas.

Finalmente, se observa que la atraccion y generacion de viajes estan altamente correlacionadas y, cuanto mayor sea la atraccion de una zona, mas viajes generara.



El **mapa de calor** facilita la interpretacion de la matriz de correlacion. Colores mas oscuros indican correlaciones mas fuertes, ayudando a identificar rapidamente las relaciones mas importantes.

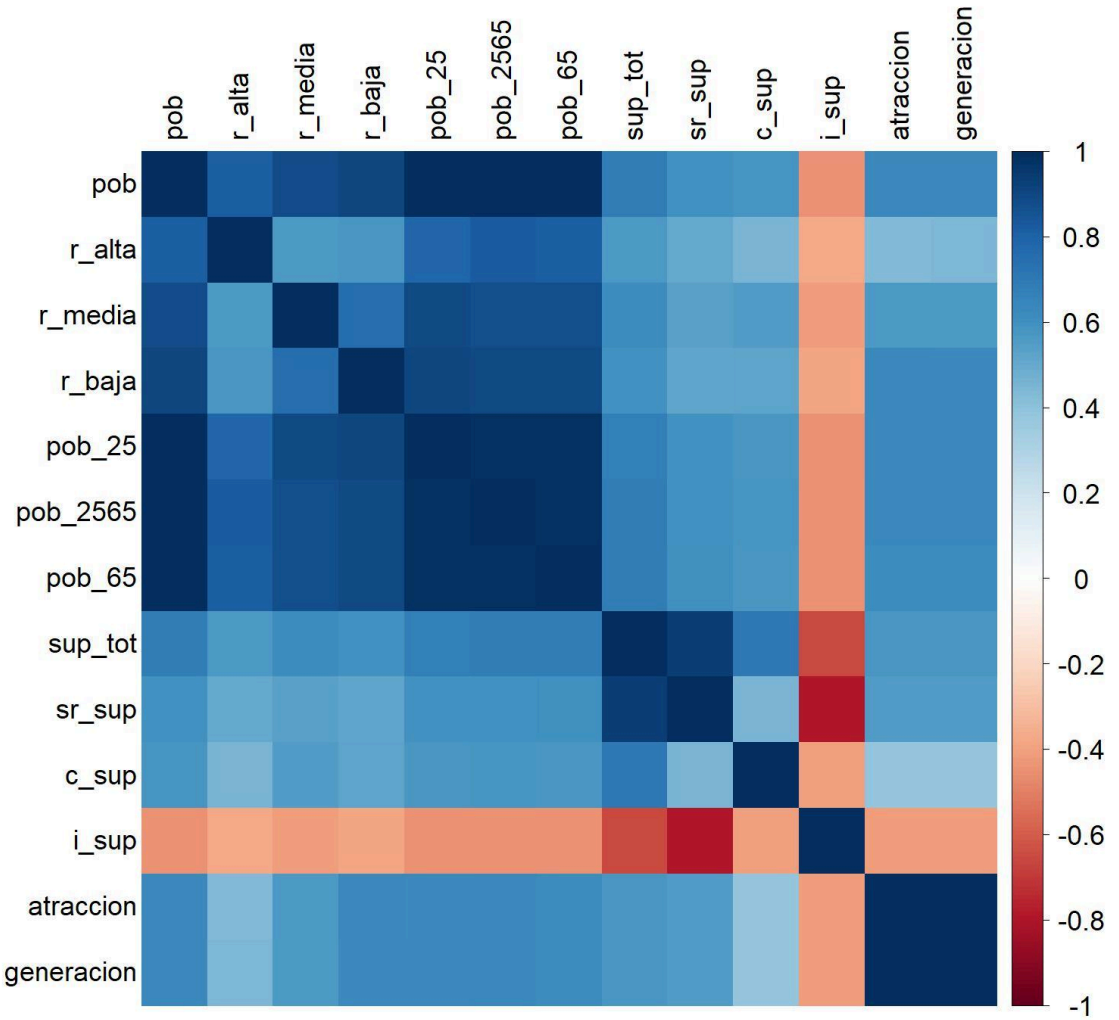


Mapa de calor con correlación de Pearson entre todas las variables. Datos Porcentajes.

Al realizar este mapa se demuestra que las zonas industriales con mayor fraccion de superficie poseen correlaciones negativas con la mayor parte de las variables, con excepcion de *r_baja*. Una explicacion seria que las personas con

renta baja generalmente viven en regiones mas perifericas, donde se encuentran la mayoria de las zonas industriales.

En general, para las metricas de superficie, se observa que cuanto mas ocupada esta la superficie con actividades industriales, menor es la poblacion presente y la atraccion y generacion de viajes se ven negativamente afectadas en estas areas. La variable poblacion muestra altas correlaciones positivas con la superficie total (*sup_tot*), las zonas residenciales, y la atraccion y generacion de viajes. Para el resto de variables, los coeficientes solo son significativos entre grupos de fracciones y, al ser fracciones demograficas del total, no resultan relevantes para el analisis.



Mapa de calor con correlación de Pearson entre todas las variables. Datos Absolutos.

Se opta por utilizar el dataset con fracciones, ya que al considerar la base con valores absolutos se ve que todo el conjunto tiene correlacion con la salvedad que destaca de la correlacion negativa de la superficie industrial (*i_sup*) con las demas variables.

D. ANALISIS DE COMPONENTES PRINCIPALES

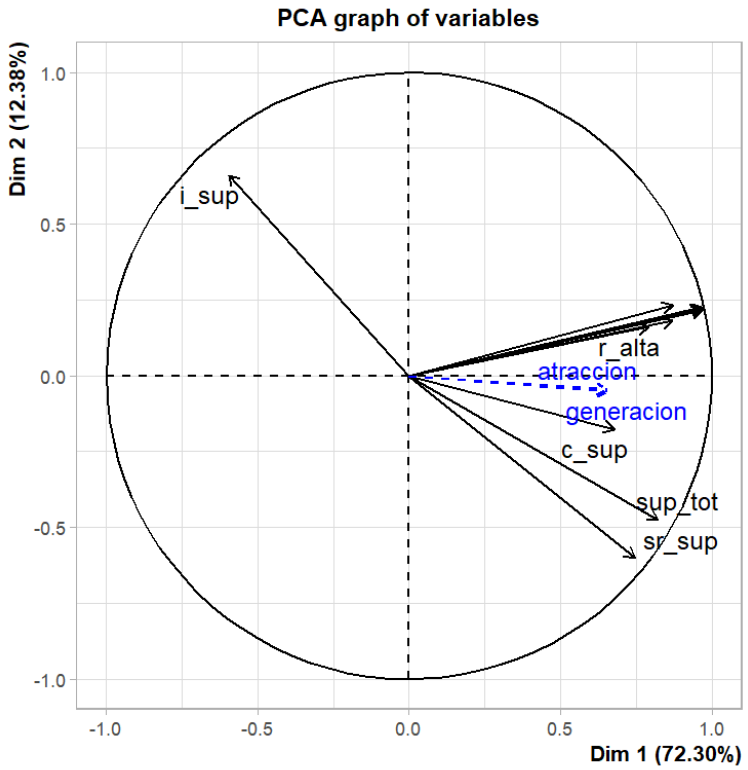
El PCA se utiliza para reducir la dimensionalidad de los datos mientras se conserva la mayor cantidad posible de su variabilidad explicada. En el analisis se utilizan las librerias *FactoMineR* y *factoextra*. Se utilizan los datos en valores absolutos.

En la siguiente tabla se puede observar, por un lado, cuantas de las 11 variables explicativas antes introducidas sustituyen cada una de las nuevas variables factor creadas. Por lo tanto, se puede ver que la primera variable factor recoge, por si misma, el comportamiento de casi 8 de las variables introducidas.

	Dim.1	Dim.2	Dim.3	Dim.4	Dim.5	Dim.6	Dim.7	Dim.8
Variance	7,953	1,363	0,628	0,484	0,297	0,247	0,016	0,014
% of var.	72,299	12,380	5,709	4,398	2,701	2,243	0,147	0,124
Cumulative % of var.	72,299	84,678	90,387	94,785	97,486	99,729	99,876	100

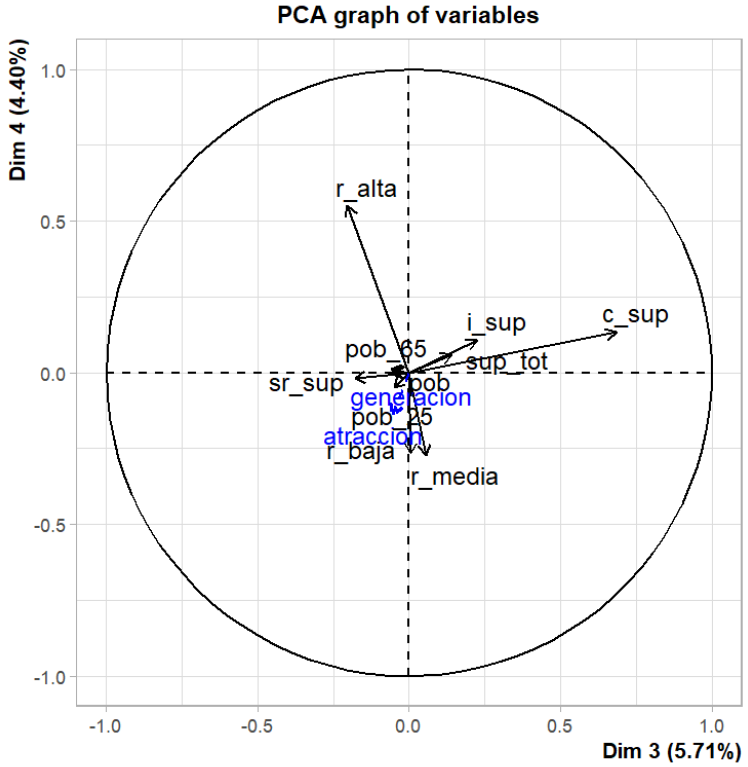
Siguiendo el criterio de Kaiser, se concluye que se deberian conservar solo las dos primeras variables factores.

En la primera grafica en la que se comparan la informacion de las dos primeras componentes principales se observa la correlacion entre: las variables referentes a la poblacion entre si, las variables de generacion y atraccion y las de superficie, salvo i_sup que correla inversamente.

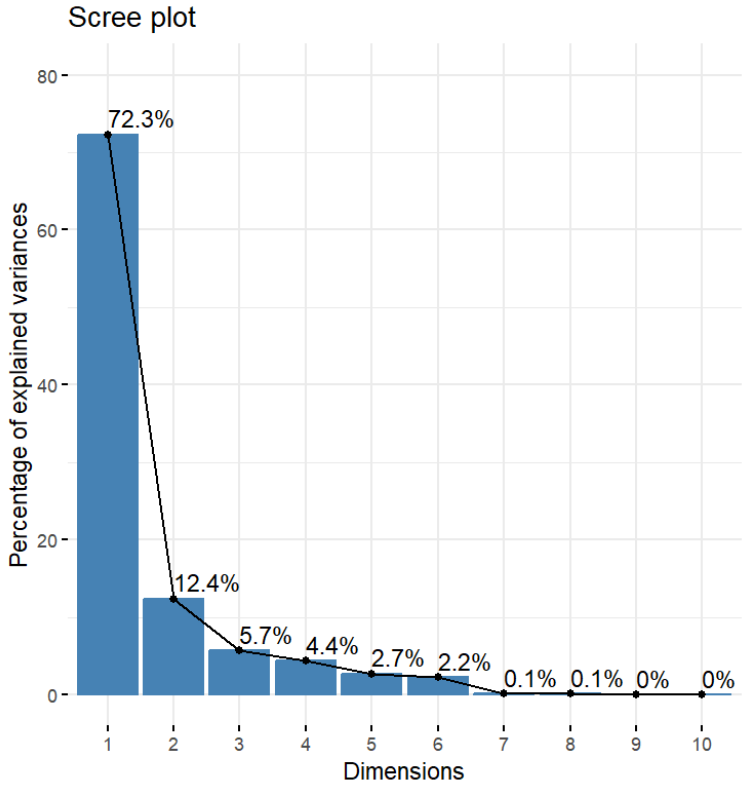


Las caracteristicas superpuestas se refieren a las de poblacion: total, por edades y por renta.

En lo que se refiere a las variables factor 3 y 4, ya se conoce que estas recogen menos informacion que uno solo de los parametros introducidos como variables activas. Ademas, a esta reducida importancia se le suma la poca interpretabilidad de las relaciones entre variables que se puede ver en la grafica. Los parametros se muestran en una estructura bastante confusa que no permite sacar conclusiones.



En esta grafica se observa el porcentaje de variabilidad explicado para cada una de las componentes principales. Seleccionando las dos primeras variables factor se recoge practicamente el 85% de la varianza de los datos.



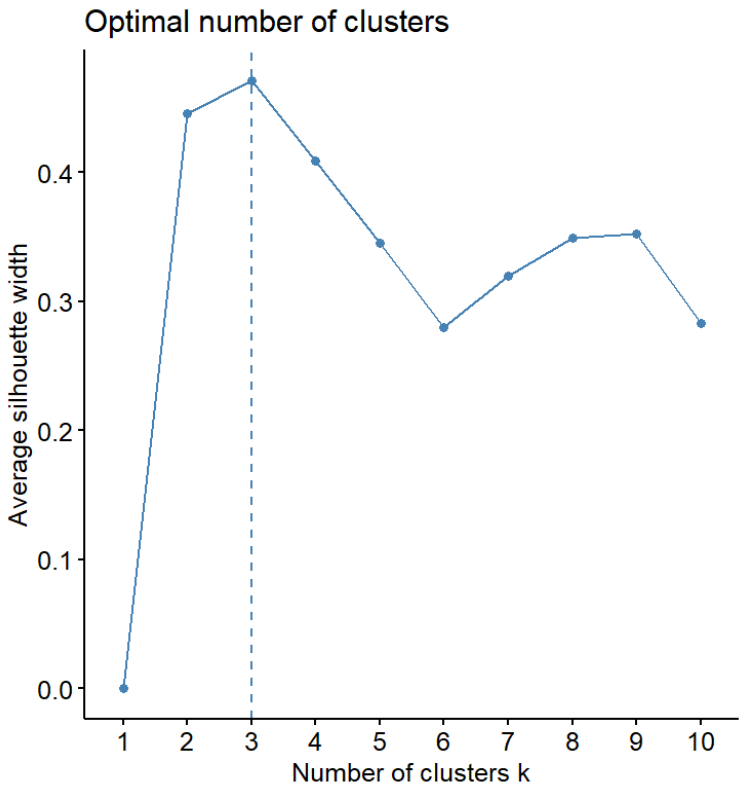
Como **conclusion**, el analisis sugiere que se pueden utilizar un numero reducido de componentes principales para capturar la mayoria de la informacion en los datos. Sin embargo, se ha decidido utilizar las caracteristicas originales porque:

- Son pocas y se dispone de capacidad de procesamiento.
- Mantener las caracteristicas originales evita posibles perdidas de informacion debido a la reduccion dimensional.
- Facilita la interpretacion de los modelos y resultados evitando tener que revertir la transformacion realizada.

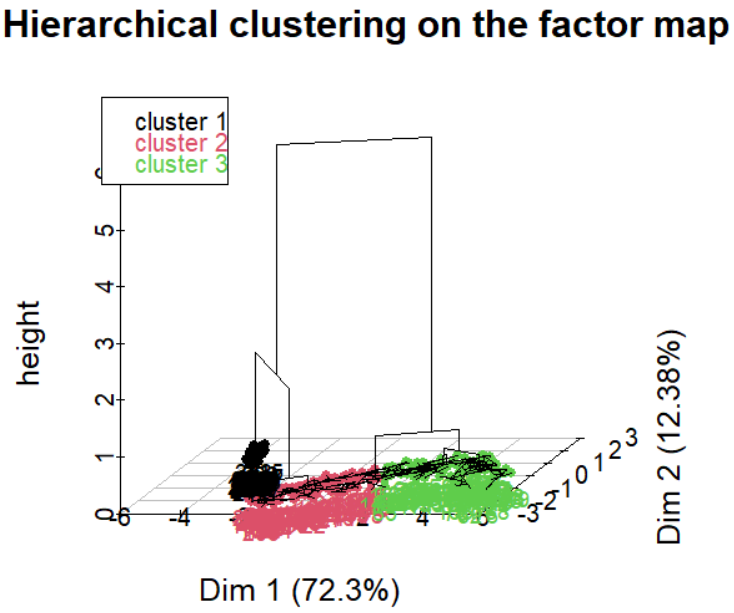
E. CLUSTERING

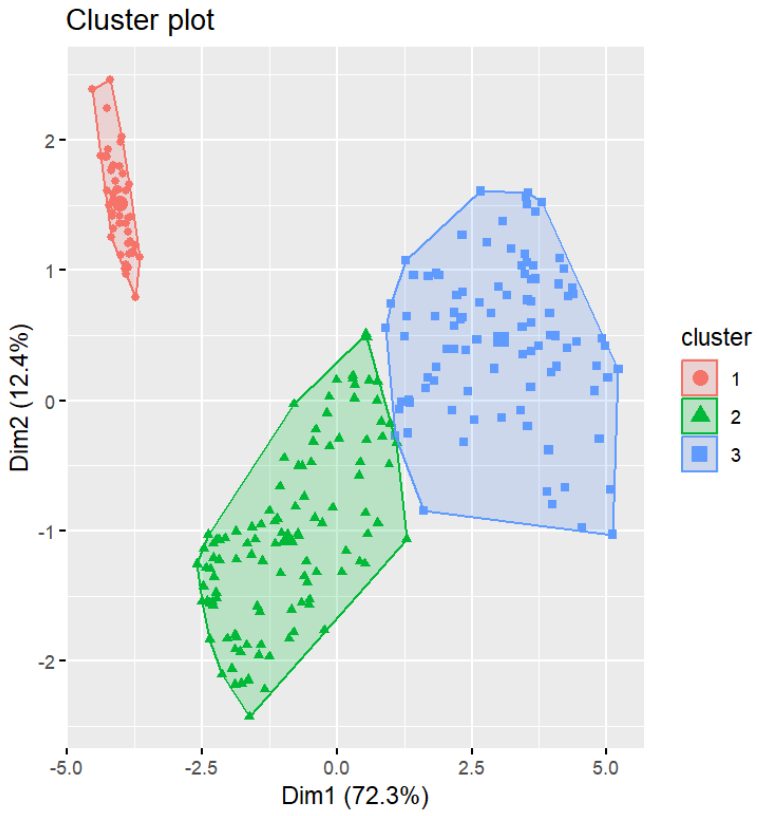
Este apartado de clustering tiene la finalidad, por un lado, de conocer mas en profundidad las posibles agrupaciones y similitudes entre las diferentes areas de estudio, y por otro, generar una nueva columna en el data que haga referencia a la pertenencia a estos clusters con la finalidad de servir de ayuda para el entrenamiento de los modelos. En el analisis se utilizan las librerias *dbscan*, *cluster* y *dplyr*.

Para comenzar se define el numero optimo de clusters en los que clasificar los datos, empleando la metrica del **ancho de la silueta**. En este caso, dicha metrica se maximiza con tres clusters.

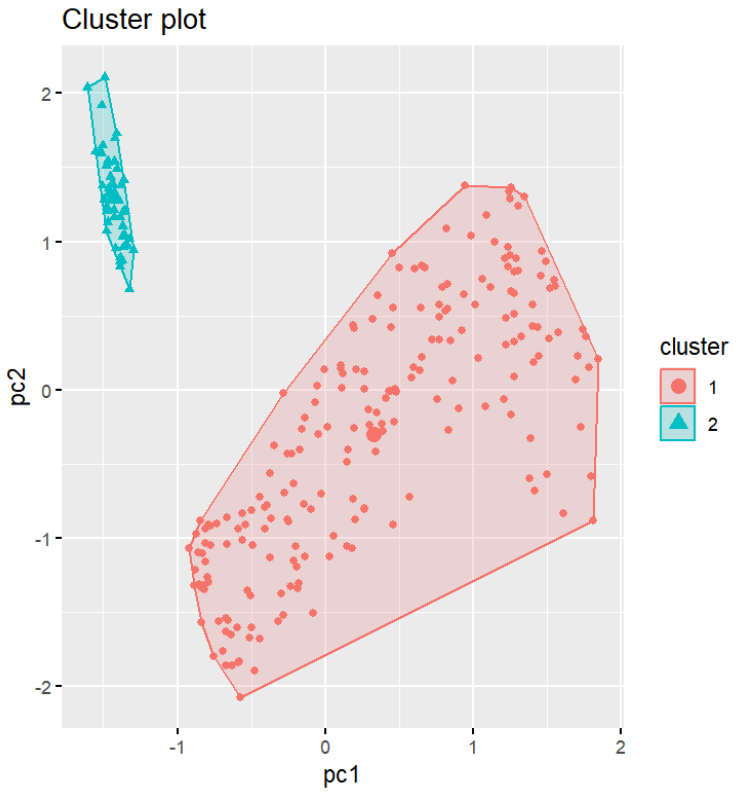


Empleando el metodo de **clustering jerarquico HCPC** se han formado los tres clusters que se muestran en las figuras siguientes.

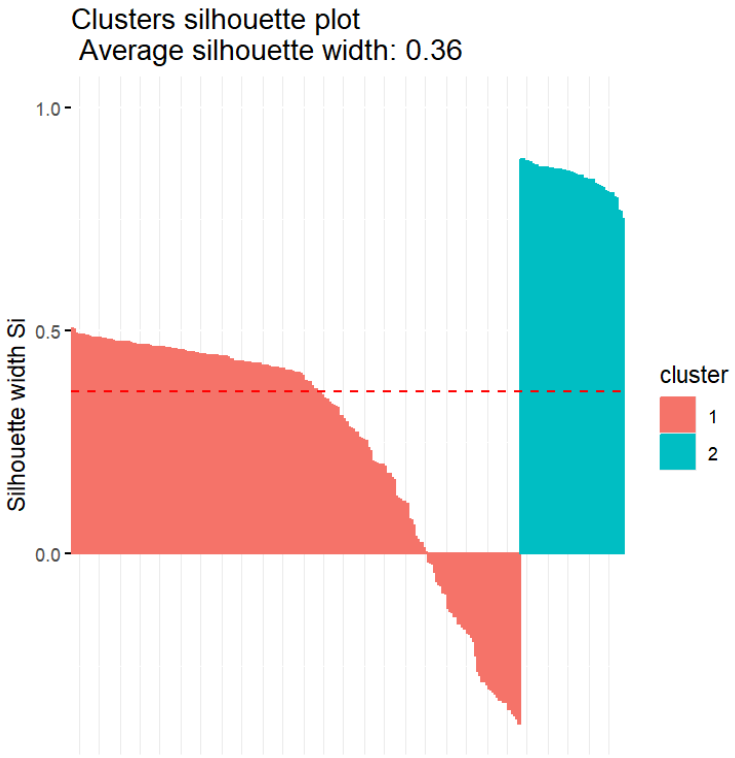




Al concluir por observacion visual que los datos tambien se pueden clasificar en dos unicos grupos, se ha decidido emplear el **clustering no-jerarquico *dbscan*,** formando los dos clusters que se muestran a continuacion.



Esta nueva clasificacion es de calidad moderada, ya que el ancho medio de la silueta queda positivo pero no con gran margen. Es especialmente adecuada la clasificacion del segundo cluster, mientras que el primero si que muestra disonancias en algunos casos con anchos de silueta negativos.



Evaluando todo lo anterior, se decide que la explicabilidad del clustering en 3 grupos puede ser ligeramente mejor y es con el que se decide continuar. Así, se espera que se expliquen con mas precision los datos en lugar de emplear todo un grupo grande.

V. FEATURE IMPORTANCE Y FEATURE SELECTION

A. FEATURE IMPORTANCE

En este apartado se realiza un analisis de la importancia de las características para la prediccion de dos variables objetivo: atraccion y generacion. Para ello se usan los datos en valores absolutos. Los calculos de importancia de características se han realizado sobre modelos de Random Forest usando la medida de *IncNodePurity*. En terminos generales, una característica con un alto valor de *IncNodePurity* se considera mas significativa para predecir las variables objetivo.

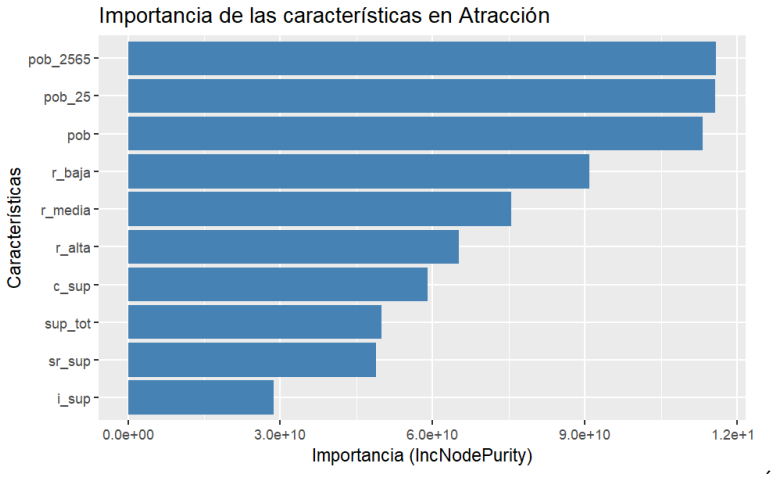
Los Random Forest son una tecnica de ensamblaje que puede mejorar la precision y generalizacion del modelo. Esta formado por un conjunto de arboles de decision, en los cuales las entradas se forman mediante el aprendizaje de cada arbol por un subconjunto de las entradas. Las predicciones finales son el promedio de las predicciones de los arboles.

La importancia de las caracterısticas en Random Forest se mide de las siguientes dos maneras:

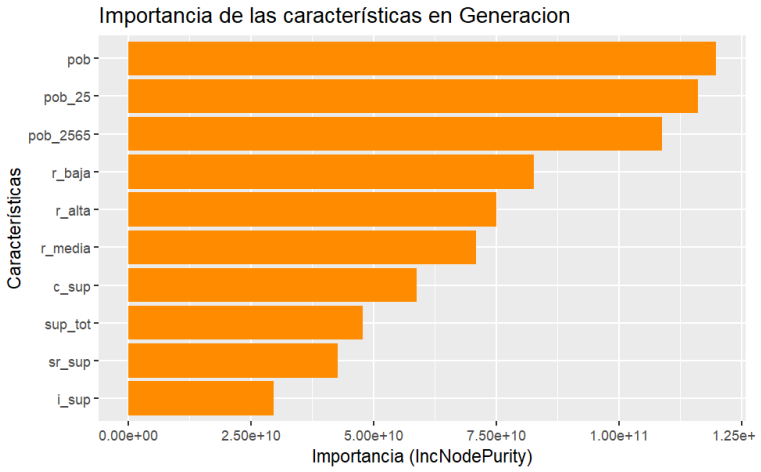
- Mean Decrease in Accuracy (%IncMSE): Mide la perdida de precision del modelo al permutar aleatoriamente una caracterıstica. En consecuencia, si una caracterıstica muestra una gran perdida, se considera muy importante para el modelo.
- Aumento de la Pureza del Nodo (IncNodePurity): Mide cuanto disminuye la impureza del nodo (generalmente medida con el ındice de Gini o la entropıa) al dividir los datos con la caracterıstica. Se considera que la caracterıstica tiene importancia para el modelo si el ındice de pureza del nodo para esa caracterıstica es alto.

Se ha usado ***IncNodePurity*** para determinar la importancia de las caracterısticas. Las caracterısticas con el mayor valor de *IncNodePurity* se consideran mas importantes porque contribuyen mas a la pureza de los nodos en los arboles de decision.

Las caracterısticas mas importantes en la **prediccion de la atraccion** son:



Las caracterısticas mas importantes en el **pronostico de generacion** son:



Este analisis demuestra que, aunque con diferencias, todas las caracterısticas son posiblemente influyentes y de gran utilidad de cara al entrenamiento de los modelos.

B. FEATURE SELECTION, ESCALADO Y TRANSFORMACIONES

En lo que respecta a la **seleccion** de las features, se descarta el id de la zona ya que no tiene relacion alguna con las target, es tan solo un identificador.

Ademas, se decide retirar una caracterıstica de cada grupo de variables que representaban porcentajes y que, sumadas, totalizan el 100%. Esta decision se toma para evitar redundancias y multicolinealidades, ya que estas caracterısticas son linealmente dependientes. Conocer dos de estas caracterısticas permite determinar el valor de la tercera.

En resumen, la eliminacion de una caracterıstica en cada grupo de variables porcentuales mantiene la integridad de la informacion, simplifica el modelo y mejora la eficiencia del entrenamiento, minimizando el riesgo de sobreajuste, sin comprometer la precision y calidad de las predicciones.

La “feature importance” de la Random Forest indica que todas las caracterısticas son posiblemente influyentes y de gran utilidad de cara al entrenamiento de los modelos. Por consiguiente, se mantienen todas las caracterısticas, ya que no se dispone de muchas variables y se pueden manejar todas. De esta manera, el modelo es un poco mas robusto que sin considerar algunas de las variables y su influencia.

Finalmente se decide emplear *pob*, *r_media*, *r_alta*, *pob_25*, *pob_2565*, *sup_tot*, *sr_sup* e *i_sup* para entrenar el modelo. Asimismo, en algunos modelos la ingesta de datos incluye los datos de clusterizacion con *cluster_1*, *cluster_2* y *cluster_3*.

Por ultimo y antes de introducir los datos, estos se separan en dos para tener un dataset de train y otro de test.

A la hora de alimentar los modelos, los datos han de ir **escalados**. Para ello se proponen distintos tipos de escalados que modificaran de manera diferente los datos de ingesta de los modelos. Aquı los que se usan:

- El modelo de escalado de datos mediante el "**Standard Scaler**" o escalador estandar consiste en transformar los datos de manera que tengan una media de 0 y una desviacion estandar de 1. Este proceso se conoce tambien como **centrado y reducido**.

$$x = (x_0 - \mu) / \sigma$$

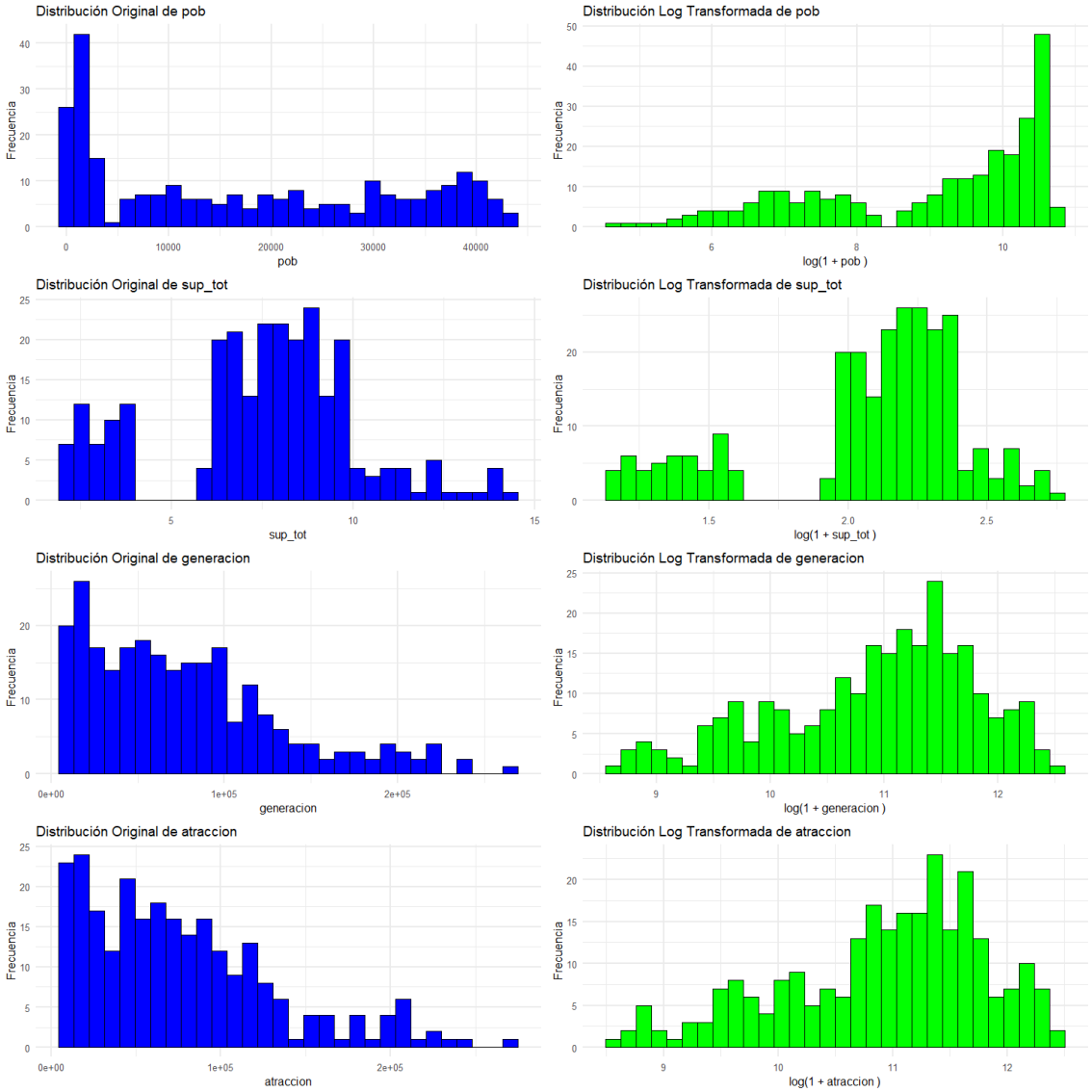
  Centrado: Restar la media (μ) de cada caracterıstica de los datos.

  Escalado: Dividir cada caracterıstica por su desviacion estandar (σ).

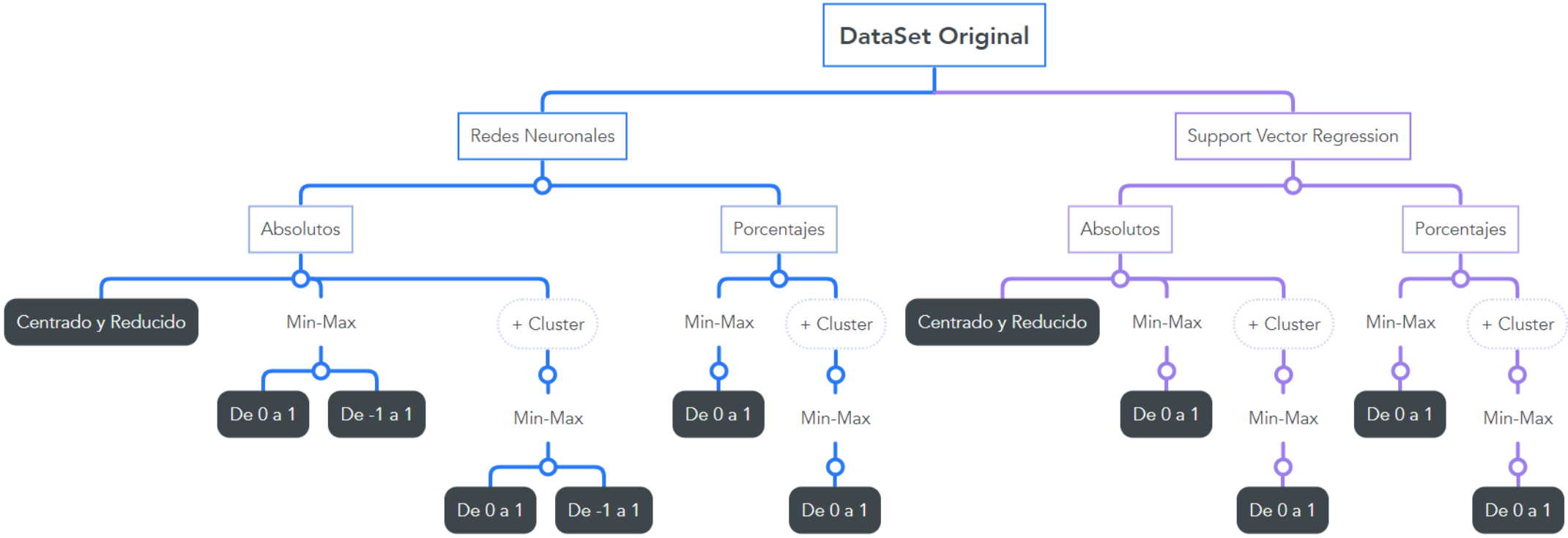
- El escalado **Min-Max**, es una tecnica de preprocesamiento de datos que transforma las caracterısticas de los datos para que se encuentren dentro de un rango especıfico, en este caso se usan dos rangos diferentes: de 0 a 1 y de -1 a 1. Este ultimo solo aplica a las redes neuronales.

$$x = (x_0 - x_{min})/(x_{max} - x_{min})$$

Atendiendo a la distribucion de los valores en *pob*, *sup_total*, *generacion* y *atraccion* se evalua si una **transformacion** logarıtmica conseguirıa normalizar las variables. Estos son los histogramas de las variables:



Como se observa, las transformaciones logarıtmicas no mejoran demasiado la no normalidad de las caracterısticas ası que, se descarta aplicarlas para tambien evitar dificultad de interpretacion.



Este grafico muestra los diferentes datasets que han tenido los modelos como ingesta y sus propiedades.

VI. REDES NEURONALES

A. INTRODUCCION

Las redes neuronales son algoritmos fundamentados en el funcionamiento del cerebro humano para el reconocimiento de patrones y la realizacion de predicciones en datos complejos. En su estructura basica, una red neuronal esta compuesta por unidades fundamentales llamadas neuronas artificiales, las cuales imitan el comportamiento de las neuronas biologicas. Cada neurona artificial recibe multiples entradas, las procesa mediante una funcion de activacion no lineal (como sigmoid, ReLU, etc.) y genera una salida. La eleccion de la funcion de activacion es crucial, ya que influye en la capacidad de la red para modelar relaciones complejas en los datos.

B. DEFINIR Y ENTRENAR LA RED NEURONAL

Se utiliza la funcion *neuralnet* de la biblioteca *neuralnet* para entrenar redes neuronales artificiales. Inicializa la red con una estructura y pesos definidos, realiza propagacion hacia adelante para calcular salidas, calcula el error comparando con salidas reales, ajusta los pesos mediante retropropagacion, y repite el proceso hasta que el error sea mınimo o se alcance el lımite de iteraciones.

Primero, se definen las variables objetivo. Como son dos (atraccion y generacion), se han creado dos modelos de red neuronal, uno para cada variable objetivo. Las demas variables (*pob, r_alta, r_media, pob_25, pob_2565, sup_tot, sr_sup, i_sup*) actuan como variables predictoras.

Posteriormente, se determina el numero de neuronas en la capa oculta de la red neuronal. La salida de la red neuronal se establece como lineal, y el error se evalua utilizando la metrica *sse* (suma de los errores cuadrados), adecuada para problemas de regresion. Ademas, se establece un umbral de convergencia (*threshold*) y un lımite maximo de iteraciones (*stepmax*) para el proceso de entrenamiento.

C. PREDICCION, DESESCALADO Y EVALUACION

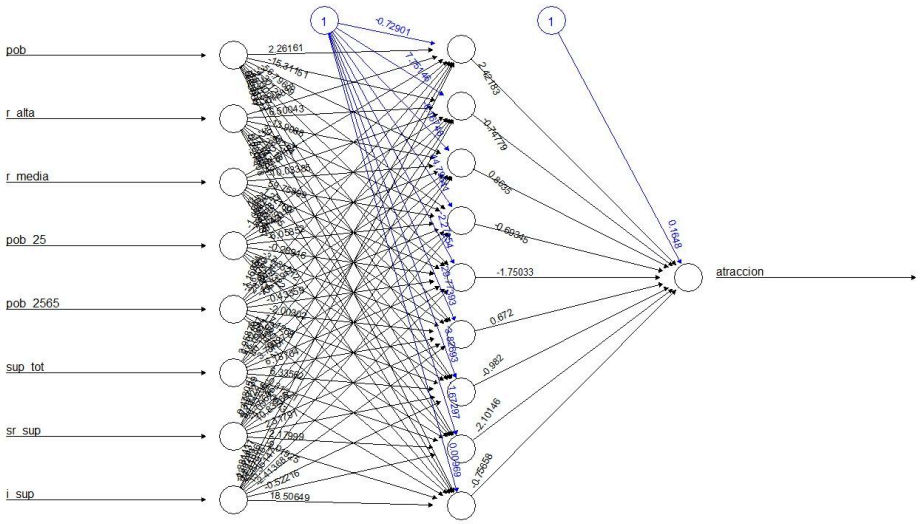
Una vez finalizado el entrenamiento, se realizan predicciones utilizando la funcion *predict()* para tanto los datos de entrenamiento como los de prueba. Los valores predichos se desescalan utilizando la misma formula que para escalar los datos pero de forma inversa, asegurando que las predicciones del modelo esten en la misma escala que

los datos originales del problema. Esto permite una comparacion directa e interpretable entre las predicciones del modelo y los valores reales.

La evaluacion de estas predicciones se lleva a cabo mediante un script externo denominado *EvaluacionResultados.R*, el cual contiene una funcion personalizada disenada para analizar y comparar los resultados obtenidos por el modelo. Las metricas utilizadas para la evaluacion incluyen el *RMSLE* (Root Mean Squared Logarithmic Error) y el R^2 (Coeficiente de Determinacion), que proporcionan una vision integral del rendimiento del modelo. Asimismo, se ha analizado la ecuacion de la regresion lineal de las predicciones en el espacio compuesto por los valores reales en el eje horizontal y los predichos en el vertical.

D. MODELOS

Para explorar y comparar diferentes configuraciones de la red neuronal, se crean modelos variando el numero de neuronas en la capa oculta (3, 5, 7 y 9). Cada uno de estos modelos se entrena, predice y evalua, permitiendo asi una comparacion directa del rendimiento entre diferentes arquitecturas de red. En la siguiente figura se puede ver la estructura de las redes neuronales en el caso de 9 neuronas en la capa intermedia.



Para cada numero de neuronas se han creado multiples modelos, comparado las predicciones con los valores test, y al final se ha escogido el modelo de menor error.

Los escenarios evaluados son los siguientes:

ABSOLUTOS:

CR: Datos normalizados mediante *Standar Scaler* (centrados y reducidos).

range01: Datos escalados en un rango de 0 a 1.

range11: Datos escalados en un rango de -1 a 1.

C. range 01: Datos escalados en un rango de 0 a 1, incluyendo clusters.

C. range11: Datos escalados en un rango de -1 a 1, incluyendo clusters.

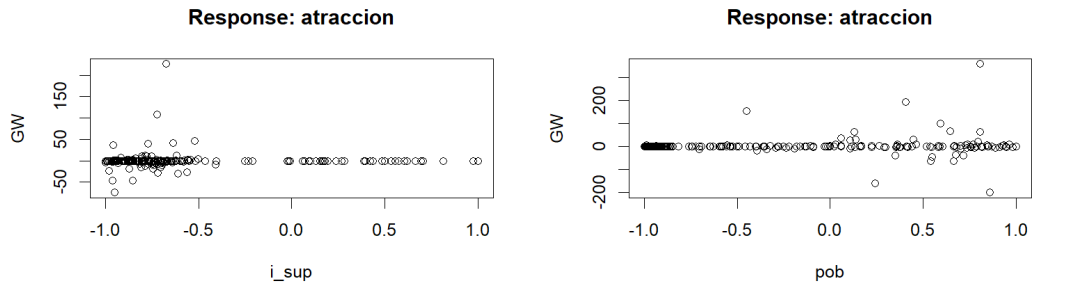
PORCENTAJES:

range01: Las fracciones sin cambios y el resto de parametros escalados de 0 a 1.

C. range 01: Las fracciones sin cambios y el resto de parametros escalados de 0 a 1, incluyendo clusters.

Para cada tipo de escenario se han evaluado modelos de 3, 5, 7 y 9 neuronas, cinco de cada. Es decir, en total 20 modelos neuronales para cada tipo de escalado. Se ha creado un loop en el archivo para que este se quede con el que menor suma de errores cuadrados (escalados) tiene respecto a los datos test.

A la hora de construir los modelos, tambien se ha evaluado la varianza de los pesos para cada uno de los parametros. Una mayor varianza indica que el parametro es mas relevante a la hora de definir la variable target, mientras que unos pesos constantes y en torno al 0 indican que el parametro no afecta a la target. A continuacion se muestran dos de las graficas de todas las que se han evaluado, donde se puede ver que en este caso los parametros si que ayudan a definir la target.



En todos los modelos con los que se ha trabajado se ha observado una mınima variacion de los pesos para todos los parametros.

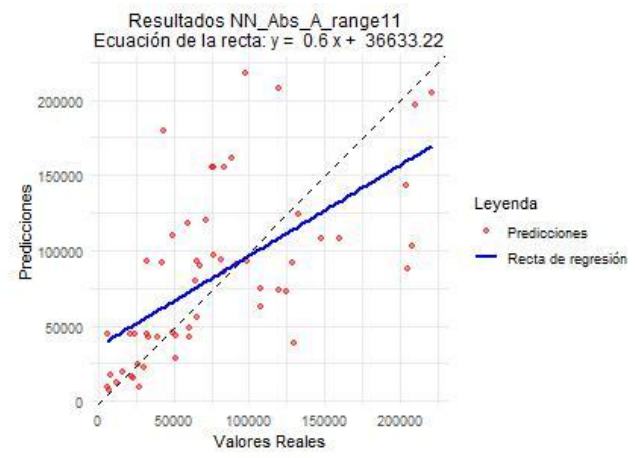
E. CONCLUSIONES

Atracción

ATRACCION							
5 de cada	ABSOLUTOS					PORCENTAJES	
	CR	range01	range11	C. range 01	C. range11	range01	C. range 01
Num. Neuronas optimo	9	9	9	9	9	9	9
RMSLE	2,403	0,868	0,600	2,185	1,482	0,992	0,647

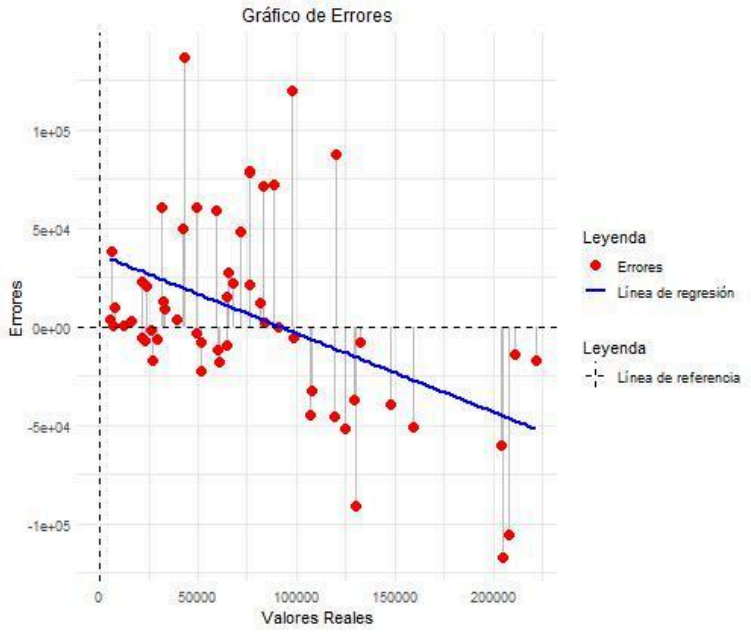
Para los modelos de atraccion la cantidad optima de neuronas en la capa oculta,

evaluada en todos los escenarios, es 9. El modelo que proporciona el valor mas bajo de la metrica de error RMSLE es aquel que utiliza los datos normalizados en un rango de -1 a 1, con un valor de error de 0,6.



El grafico de dispersion del "mejor" modelo muestra un coeficiente RMSLE de 0.6, lo que indica que las predicciones del modelo no coinciden de manera precisa con los valores reales, tal y como se puede observar en la grafica superior. Si se observa la ecuacion de la linea, la constante de alto valor de 36.633,22 sugiere una tendencia a sobrestimar las predicciones iniciales. En cambio, la poca pendiente de la recta (0,6) hace que las predicciones de distritos mayores infraestimen la atraccion real.

Esto se puede ver con todavia mas claridad en la grafica de error mostrada seguidamente. Se ve una tendencia clara de sobreestimar los valores mas pequenos, algo que se repite e incluso se acentua en el caso de los valores medianos; y en cambio, los valores mas altos son ampliamente infraestimados.

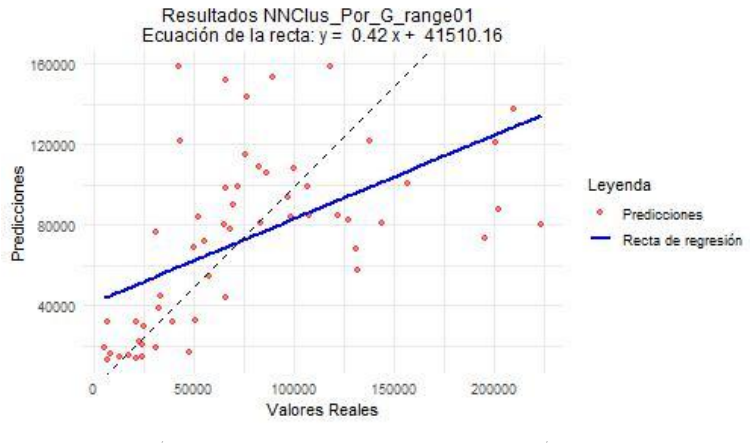


Generación

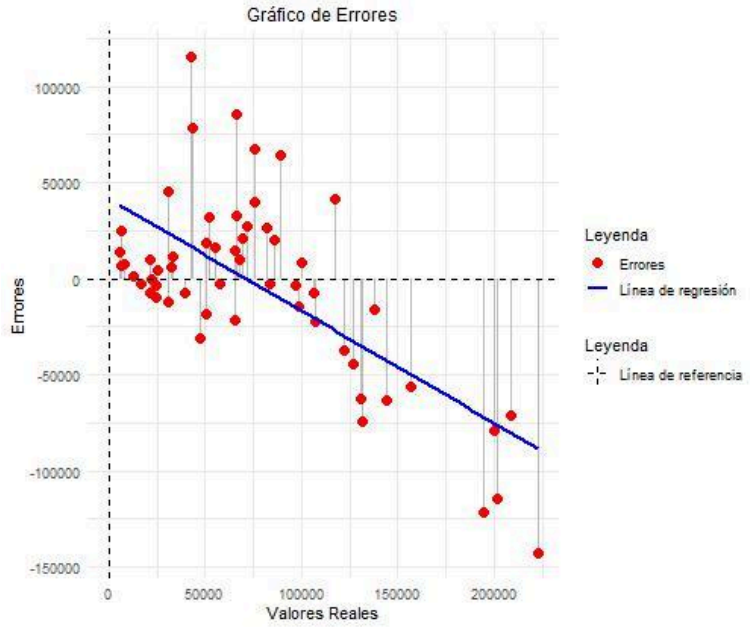
GENERACION							
5 de cada	ABSOLUTOS					PORCENTAJES	
	CR	range01	range11	C. range 01	C. range11	range01	C. range 01
Num. Neuronas optimo	9	9	9	9	9	9	9
RMSLE	2,027	0,606	0,761	0,588	0,626	1,234	**0,584**

Para los modelos de generacion la cantidad optima de neuronas en la capa oculta continua siendo 9, Sin embargo, los valores obtenidos para la metrica de error RMSLE son mas bajos que los obtenidos en los modelos de atraccion. Debido a que los modelos neuronales practicamente nunca son iguales y por tanto el error tambien varia, se cree que los valores del error estan muy cerca y que un modelo u otro sea mejor es mas el resultado del azar que de las caracterısticas que tenga. En este caso en concreto, resulta que el modelo que emplea datos en fracciones incluyendo clusters y escalado de 0 a 1 funciona ligeramente mejor que el resto.

Una vez mas, en la grafica inferior se observa que el modelo tiende a infravalorar la generacion en los distritos donde los valores reales son mayores.



Evaluando la grafica del error mostrada a continuacion, se puede ver que a pesar de emplear un reescalado distinto, incluir los datos de clustering y tratar esta vez de predecir la generacion, las tendencias son las mismas que en el modelo de atraccion. Se sobreestiman los valores mas pequenos y especialmente los medianos, y en cambio infraestimar los mayores.



Es notable que los modelos realizados con clusters tienden a tener mejores resultados en generacion y peores en atraccion, pero se cree que esto tambien es resultado del azar y que la respuesta a si la inclusion de clusters mejora el modelo es inconcluyente. Lo que si que se observa de forma bastante clara en los datos tanto de generacion como de atraccion es que el escalado centrado y reducido puede no ser el mas adecuado. Se considera que esto es debido a los reducidos gradientes en los valores de las funciones de activacion a partir de

valores de x menores a -1 o 2 o mayores a 1 o 2. Esto es algo bastante evidente, por ejemplo, en el caso de la tangente hiperbolica.

En conclusion, los modelos de redes neuronales actuales muestran un bajo rendimiento de regresion indicando la necesidad de mas iteraciones y de explorar diferentes arquitecturas para obtener predicciones mas precisas.

VII. SVR

A. DEFINICION DEL MODELO

SVR (Support Vector Regression, un tipo de Support Vector Machine) es una tecnica de machine learning utilizada para problemas de regresion que predice valores continuos minimizando errores dentro de un margen de tolerancia y controla la complejidad del modelo para evitar sobreajuste.

En este caso se ha optado por realizar una regresion tipo nu-regression con formulacion:

$$\frac{1}{2}\|w\|^2 + C\left(\nu\xi + \frac{1}{l}\sum_{i=1}^{l}(\xi_i + \xi_i^*)\right)$$

Y kermel, radial basis:

$$e^{-\gamma|\mu-\nu|^2}$$

B. METRICA EVALUACION

El primer paso consiste en seleccionar una metrica para medir la diferencia entre los valores predichos por el modelo y los valores reales, proporcionando una evaluacion de la precision del modelo. La funcion escogida es RMSE (Root Mean Squared Error), ya que es facil de interpretar facilitando la optimizacion del modelo. Ademas, es una metrica estandar ampliamente aceptada, lo que permite comparaciones y comunicacion efectiva de resultados. Esta calcula la raiz cuadrada de la media de los cuadrados de las diferencias entre los valores predichos y reales.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

C. GRID DE HIPERPARAMETROS

El siguiente paso consiste en definir el grid de hiperparametros. El grid de hiperparametros permite encontrar la mejor configuracion de parametros para optimizar el rendimiento del modelo SVR, asegurando que se obtiene el mejor modelo posible para los datos dados. En este caso se utilizan los hiperparametros (*nu*, *gamma*, *cost*, *epsilon*).

nu: Controla la fraccion de errores permitidos durante el entrenamiento y el numero de vectores de soporte. Un valor mas alto de nu permite mas errores, util en datos ruidosos.

gamma: Regula la influencia de un solo ejemplo de entrenamiento en el modelo RBF. Bajo *gamma* suaviza el modelo, menos sensible a ruidos; alto gamma ajusta estrechamente, captura mas detalles pero puede sobreajustar.

cost (C): Ajusta el balance entre el margen mas amplio y la clasificacion correcta de puntos de entrenamiento. Alto C prioriza clasificacion correcta, margen mas estrecho; bajo C favorece margen mas amplio, mejor generalizacion.

epsilon: Define una zona de tolerancia alrededor de la funcion objetivo donde no se penalizan errores. Bajos valores de epsilon ajustan el modelo mas cerca de los datos; altos valores aumentan la robustez ante ruidos y valores atipicos.

Para definir el grid se crea un conjunto de combinaciones posibles para los hiperparametros. Para cada combinacion en el grid, se entrena un modelo SVR. Se evalua el rendimiento del modelo utilizando el RMSE (Root Mean Squared Error). Se selecciona la combinacion de hiperparametros que produce el menor RMSE, indicando el modelo con el mejor rendimiento.

D. OPTIMIZACION CON CV

Se crea una variable para almacenar el modelo SVM que tiene el mejor rendimiento segun la metrica de RMSE (Root Mean Squared Error). Se inicializa una variable para almacenar el valor de RMSE del mejor modelo encontrado hasta el momento. Se crea una estructura para almacenar los hiperparametros que produjeron el mejor modelo en terminos de rendimiento RMSE.

Para entrenar el modelo se itera sobre cada combinacion de hiperparametros definida en el grid. Para cada combinacion, se entrena un nuevo modelo SVM utilizando los datos de entrenamiento. Una vez entrenado el modelo, se utilizan los datos de entrenamiento para hacer predicciones. Luego, se calcula el RMSE comparando las

predicciones del modelo con los valores reales conocidos del dataset train. Se evalua el rendimiento del modelo en funcion del RMSE obtenido. Si el rendimiento del modelo actual es mejor que el de los modelos anteriores (es decir, si el RMSE es menor), se actualiza el mejor modelo, el valor de RMSE y los hiperparametros correspondientes.

Hiperparametros otimizados para cada dataset:

Model	hp -> nu	hp -> gamma	hp -> cost	hp -> epsilon
SVR_Por_A_range01	0,20	0,5	100	0,1
SVR_Por_G_range01	0,20	2,0	100	0,1
SVR_Abs_A_cr	0,10	2,0	100	0,1
SVR_Abs_A_range01	0,10	2,0	100	0,1
SVRClust_Abs_A_range01	0,10	2,0	100	0,1
SVRClust_Por_A_range01	0,10	2,0	100	0,1
SVR_Abs_G_cr	0,20	2,0	100	0,1
SVR_Abs_G_range01	0,15	2,0	100	0,1
SVRClust_Abs_G_range01	0,15	2,0	100	0,1
SVRClust_Por_G_range01	0,15	2,0	100	0,1

E. ENTRENAMIENTO Y PREDICCION

Una vez optimizado el modelo se entrena con todos los datos de train y posteriormente se hacen las predicciones contra los datos de test. Para evaluarlo se utiliza una funcion de evaluacion (cargada desde un archivo externo: *EvaluacionResultados.R*) para evaluar las predicciones.

Se repite EL proceso para cada uno de los diferentes dataset.

F. EVALUACION

Metricas:

Model	RMSLE	Slope	Intercept	R2
SVR_Por_A_range01	0,659	0,459	38806,24	0,388
SVR_Por_G_range01	0,772	0,207	56997,66	0,326
SVR_Abs_A_cr	2,378	0,535	29078,62	0,489
SVR_Abs_A_range01	2,378	0,535	29088,79	0,489
SVRClust_Abs_A_range01	2,379	0,524	29580,77	0,480

SVRClust_Por_A_range01	2,379	0,524	29580,77	0,480
SVR_Abs_G_cr	2,650	0,541	28833,16	0,485
SVR_Abs_G_range01	2,650	0,541	28831,48	0,485
SVRClust_Abs_G_range01	2,650	0,530	29358,83	0,476
SVRClust_Por_G_range01	2,650	0,530	29358,83	0,476

Para una mejor **interpretacion** se resumen las principales caracterısticas:

RMSLE (Root Mean Squared Logarithmic Error): mide la diferencia entre los valores predichos y los valores reales utilizando logaritmos. Penaliza mas los errores relativos grandes y es util cuando los errores relativos son mas importantes. Un valor mas bajo de RMSLE indica un mejor rendimiento del modelo.

Slope: Es la pendiente de la lınea de regresion ajustada entre los valores reales y los valores predichos. Una pendiente cercana a 1 indica que los valores predichos aumentan en una proporcion similar a los valores reales.

Intercept: Es el valor de y para x = 0 en la lınea de regresion ajustada. Idealmente, deberıa ser cercano a cero si los valores estan bien centrados.

R2 (Coeficiente de Determinacion): Mide la proporcion de la varianza en la variable dependiente que es predecible a partir de las variables independientes. Un R2 cercano a 1 indica que el modelo explica bien la variabilidad de los datos.

Comparacion de Resultados

Mejores Modelos segun **RMSLE**:

SVR_Por_A_range01 (RMSLE = 0.659) y *SVR_Por_G_range01* (RMSLE = 0.772) son los modelos con mejor rendimiento. Ambos usan datos **porcentuales**.

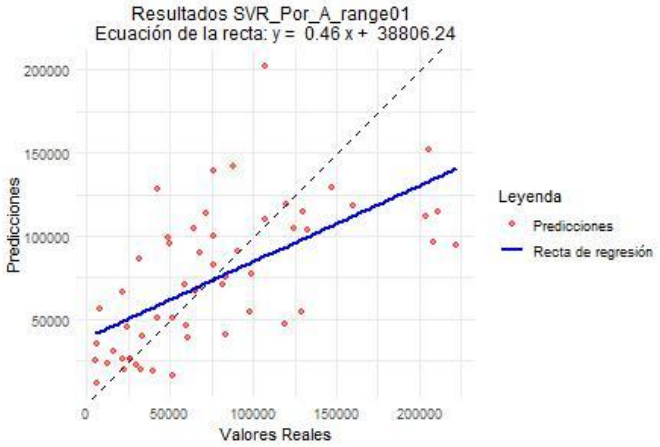
Modelos con **Peor** Rendimiento segun **RMSLE**:

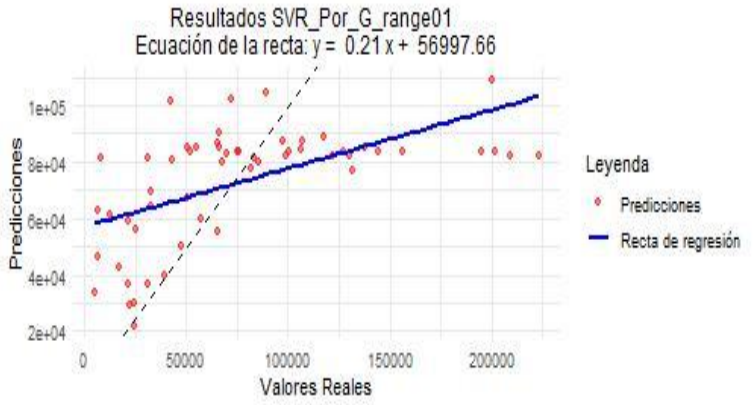
SVR_Abs_G_cr, *SVR_Abs_G_range01*, *SVRClust_Abs_G_range01* y *SVRClust_Por_G_range01* (todos con RMSLE = 2.650) tienen los peores rendimientos. Estos modelos usan datos **absolutos** y todos predicen **Generacion**.

El efecto de la Escala no parece ser relevante. Si, el hecho de que los datos esten en valores absolutos(escalados) o en valores porcentuales, siendo estos ultimos los que muestran el mejor comportamiento en terminos de rendimiento por minimizacion de error. Esto puede ser debido a que los datos porcentuales ya estan normalizados y reflejan mejor las relaciones relativas entre las variables. La clusterizacion parece no ayudar a mejorar el entrenamiento del modelo. Los modelos predictores de Atraccion tienden a tener un mejor rendimiento que los que predicen la Generacion de viajes.

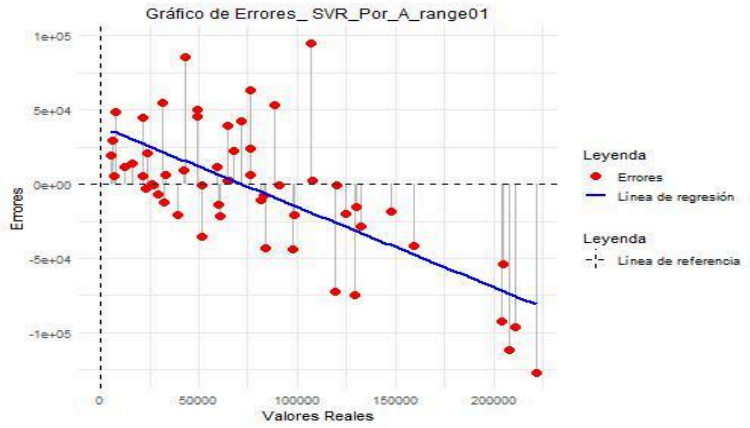
En general las predicciones de los modelos tienen una linea de regresion ajustada con un slope y un intercept que indican que se tiende a subestimar los valores superiores y sobreestimar los bajos. Es interesante observar como los mejores modelos segun la metrica de error, son los que peor Slope e Intercept presentan.

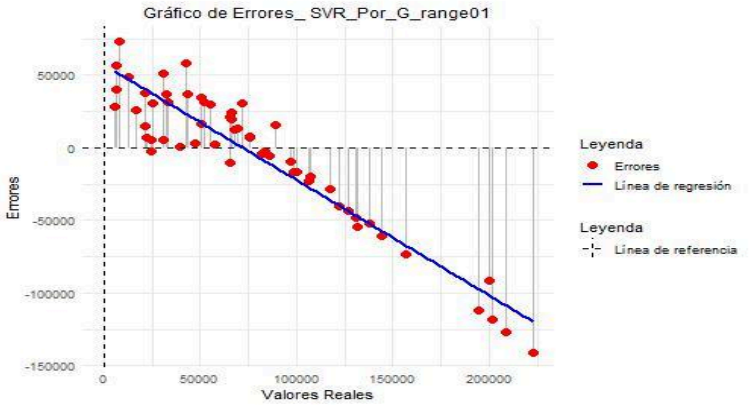
En terminos de R2 ocurre lo mismo que con la recta, los modelos entrenados con los datasets que tienen la mejor metrica muestran un R2 peor que el resto, lo que indica que explican menos variabilidad en los datos porcentuales.



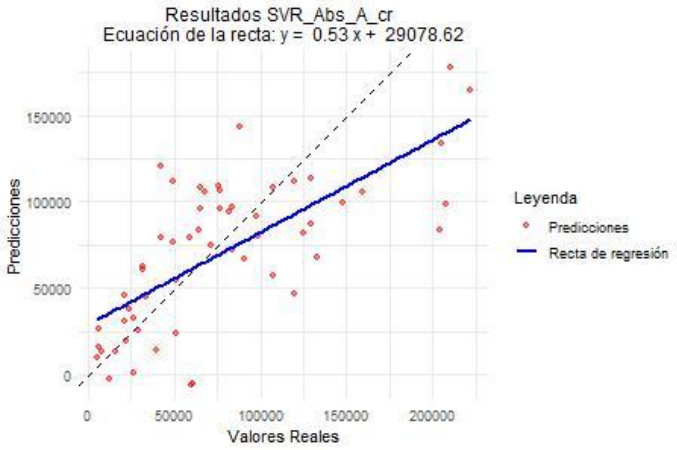


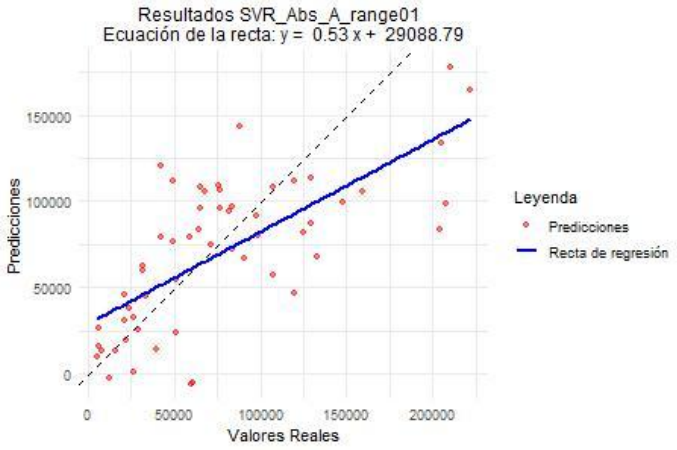
No obstante, seria muy osado afirmar que estos son buenos modelos, ni siquiera los mejores, atendiendo a la representacion grafica de las predicciones y de la recta de regresion. Sobre todo en el caso de la prediccion de Generacion.



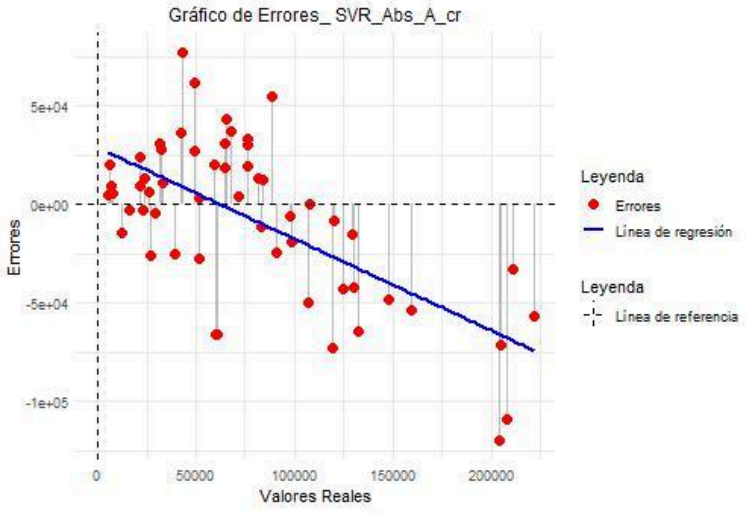


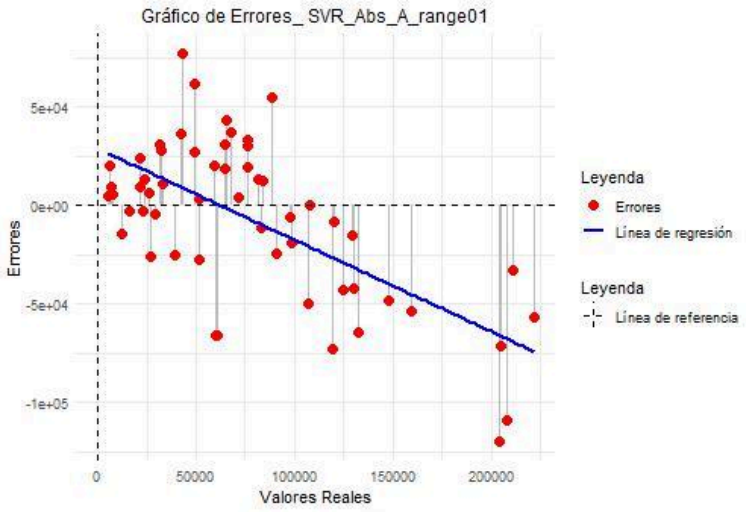
Estos graficos muestran el error absoluto de cada uno de las predicciones. Parece que en las zonas de valores bajos tiene mayor precision y acierto y un menor error, aunque tiende a sobreestimar las predicciones. Es en los valores altos donde falla mucho y subestima la prediccion.





En estos casos, aunque con mucha diferencia en la metrica de error, parece que la recta de regresion fuera mas acertada, a la vez que estos modelos tienen un R2 mayor.





Al atender a los errores absolutos se aprecia un comportamiento similar a las dos primeras opciones analizadas.

G. CONCLUSIONES

Los modelos de porcentaje atraccion y porcentaje generacion escalados en un rango de 0 a 1 son los mejores modelos en terminos de RMSLE, aunque muestran una menor capacidad explicativa segun el R^2. En este caso, a diferencia del modelo de redes neuronales, la prediccion de atraccion es ligeramente mas precisa en comparacion con la prediccion de generacion de viajes.

Como conclusiones del modelo se puede decir que la escala Min-Max parece ser mas efectiva en comparacion con la centrada y reducida (standard scaler). Ademas, la adicion de una columna de clusterizacion, con los datos trabajados, no mejora el rendimiento. Por ultimo, cabe destacar que los datos porcentuales tienden a producir modelos con mejor rendimiento en comparacion con los datos absolutos.

VIII. COMPARACION DE MODELOS

Para proporcionar una perspectiva mas completa, se compara el rendimiento de las redes neuronales con SVR (Support Vector Regression). Todos modelos se colocan en una tabla donde se compararan en la tabla inferior.

Model	RMSLE	Slope	Intercept	R2
NNClus_Por_G_range01	0,584	0,417	41.510	0,303
NN_Abs_A_range11	0,600	*0,600*	36.633	0,378
NN_Abs_G_range01	0,604	*0,449*	39.823	0,400
NNClus_Abs_G_range11	0,626	*0,433*	45.092	0,248
NNClust_Por_A_range01	0,647	*0,359*	43.727	0,214
SVR_Por_A_range01	0,659	0,459	38.806	0,388
NN_Abs_G_range11	0,761	*0,514*	35.325	0,340
SVR_Por_G_range01	0,772	0,207	56.998	0,326
NN_Abs_A_range01	0,868	*0,649*	26.322	0,347
NN_Por_A_range01	0,992	0,388	43.549	0,212
NN_Por_G_range01	1,234	0,450	38.653	0,369
NNClust_Abs_A_range11	1,482	0,605	29.511	0,281
NN_Abs_G_cr	2,027	0,485	30.643	0,298
NNClust_Abs_A_range01	2,185	0,392	33.533	0,223
SVR_Abs_A_cr	2,378	0,535	29.079	0,489
SVR_Abs_A_range01	2,378	0,535	29.089	0,489

SVRClust_Abs_A_range01	2,379	0,524	29.581	0,480
SVRClust_Por_A_range01	2,379	0,524	29.581	0,480
NN_Abs_A_cr	2,403	0,564	24.567	0,442
SVR_Abs_G_cr	2,650	0,541	28.833	0,485
SVR_Abs_G_range01	2,650	0,541	28.831	0,485
SVRClust_Abs_G_range01	2,650	0,530	29.359	0,476
SVRClust_Por_G_range01	2,650	0,530	29.359	0,476

En general, los modelos de redes neuronales (NN) tienden a tener un RMSLE mas bajo que los modelos SVR. Los modelos SVR parecen capturar mejor la tendencia de los datos, teniendo valores R2 ligeramente mas adecuados.

Se puede decir que el modelo *NN_Abs_A_range11*, tiene la segunda mejor metrica y las predicciones forman una de las rectas de regresion mas parecidas a y = x. Por lo tanto se podria decir que es el mejor modelo de los que se han entrenado.

IX. PUNTOS DE MEJORA

Parece claro que los modelos empeoran sus resultados a medida que aumenta el tamano del valor a predecir subestimando. Mas alla de tratar de ponderar esto en el entrenamiento de los modelos, seria interesante estudiar tambien las caracteristicas de las zonas en las que las predicciones fallan mas que el resto aun siendo valores pequenos.

Buscar datos reales ayudaria a tener un modelo que pudiera predecir con un poco mas de exactitud.

Probar mas arquitecturas de NN tambien seria de utilidad. Hay que tener en cuenta la carga computacional que esto puede conllevar.

X. DISTRIBUCION CARGA DE TRABAJO

	Joao	Juan Carlos	Julen	Lucas	Lucia
Limpieza			X		X
Univariante		X	X	X	X
Multivariante			X		X
Bivariante	X	X			
PCA		X	X	X	
Clustering		X	X		
Feature importance y selection		X			
Escalado y transformaciones		X	X		
Neural Networks		X	X		X
SVR		X	X		
Redaccion memoria	X	X	X		X
Presentacion	X	X	X		X