UNIVERSIDAD DE ANTIOQUIA
1803

Proyecto de Inteligencia artificial
Entrega final

Presentado por:
Andrea Sánchez Castrillón (1001420939)
Alejandro Vargas Ocampo (1088298091)

Profesor:
Raúl Ramos Pollán

Universidad de Antioquia
Facultad de Ingeniería
2023-2

Introducción

La Inteligencia Artificial (IA) se ha posicionado como una herramienta fundamental para el análisis de datos y la toma de decisiones, ofreciendo soluciones innovadoras que impactan positivamente en la sociedad. En línea con esta tendencia, se presenta nuestro proyecto de IA centrado en el Sistema de Identificación de Potenciales Beneficiarios de Programas Sociales (Sisben 2019), utilizando este último como dataset primario.

El Sisben, es un sistema ampliamente utilizado en diversos países para evaluar el nivel socioeconómico de la población y asignar recursos de manera eficiente, representa un componente esencial en la implementación de políticas públicas. Sin embargo, la complejidad y la dinámica de los factores que determinan la situación socioeconómica de los individuos demandan enfoques más avanzados y precisos. Aquí es donde entra en juego la Inteligencia Artificial.

El objetivo principal de nuestro proyecto es emplear algoritmos de aprendizaje automático y técnicas de procesamiento de lenguaje natural para mejorar la precisión y la eficacia del Sisben.

1. Exploración descriptiva del dataset

El dataset original se encuentra disponible en:
https://drive.google.com/uc?id=1AFaagbKMpY07iyvOsPpo8ZPmCMV8J0Pb

Podemos visualizar las estadísticas descriptivas del dataset, aunque estas deben analizarse con cuidado ya que algunas no nos ofrecen información relevante, por ejemplo DEPTO, MUNIC, ZONA, y COMUNA puede no tener un significado práctico o interpretativo directo. ya que estos están representados por código, que son generalmente identificadores únicos asignados a cada sitio y no tienen una relación numérica inherente que pueda ser utilizada para calcular un promedio significativo.

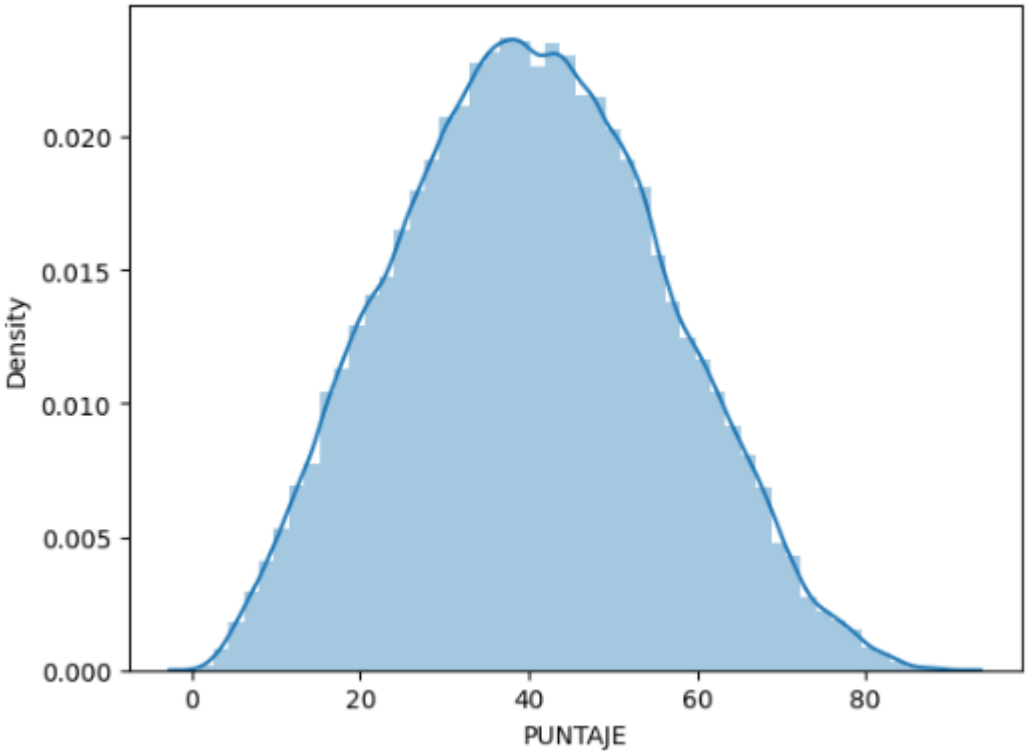
	FORMULARIO	DEPTO	MUNIC	ZONA	COMUNA
count	4.478720e+05	447872.0	447872.0	447872.000000	447872.000000
mean	2.105366e+06	5.0	1.0	1.144193	4.390866
std	6.139149e+04	0.0	0.0	0.351813	4.361268
min	2.960710e+05	5.0	1.0	1.000000	0.000000
25%	2.052189e+06	5.0	1.0	1.000000	1.000000
50%	2.104745e+06	5.0	1.0	1.000000	3.000000
75%	2.158755e+06	5.0	1.0	1.000000	8.000000
max	2.212035e+06	5.0	1.0	3.000000	80.000000

Pero para el caso de las variable puntaje, es útil analizarla ya que nos indica que el promedio de los colombianos se encontraba con un puntaje de 40 (pobreza).

	PAGAPOR	SERDOMES	PUNTAJE
count	447871.000000	447871.000000	447871.000000
mean	1.999167	1.999569	40.228908
std	0.028847	0.020754	15.714895
min	1.000000	1.000000	0.810000
25%	2.000000	2.000000	28.780000
50%	2.000000	2.000000	39.980000
75%	2.000000	2.000000	51.340000
max	2.000000	2.000000	90.230000

La variable objetivo que estamos tratando de predecir sigue una distribución gaussiana (normal), esto conlleva algunas implicaciones y consideraciones que pueden ser relevantes, por ejemplo, los modelos lineales, como la regresión lineal, asumen a menudo que los errores siguen una distribución normal. Si la variable objetivo sigue una distribución normal, esto puede ser beneficioso para estos modelos, ya que cumplen con las suposiciones subyacentes.

Una distribución normal es menos sensible a valores atípicos (outliers), en este caso la robustez de los modelos basados en distribuciones normales puede ser beneficiosa, aunque algunos modelos, como las regresiones lineales, asumen distribuciones normales, hay muchos modelos no paramétricos (como los árboles de decisión o las redes neuronales) que son menos sensibles a las suposiciones sobre la distribución.



Como se observa en la anterior figura, el comportamiento de la variable de interés (Puntaje del Sisben) presenta un comportamiento normal, por lo que no se requiere de una normalización o estandarización de la misma.

2. Preprocesado de datos

2.1 Reducción de variables

En los datos obtenidos, denominados sisben_2019.csv, se presentan alrededor de 79 variables, por cuestiones prácticas este número fue reducido a 30 de la siguiente manera: Inicialmente se estimó conveniente eliminar directamente algunas que se encuentran mejor representadas en otra variable ya existente. Dentro de las variables eliminadas se encuentran:

```
'FORMULARIO', 'SERDOMES', 'EXTRANJERO', 'TIPOESTA', 'EMBARAZA',
'CONYUVIVE', 'TRACTOR', 'ORDEN', 'CUANHORAS', 'CUANDI',  'FECHA',
           'DEPTO', 'BARRIO', 'VEREDA', 'PARED', 'PISO', 'USANITAR',
'USOSANI', 'AGUA', 'LLEGA', 'SUMINIS', 'CUANHORAS', 'PREPARAN'
```

Tras eliminar las variables mencionadas, se obtuvo un dataset con alrededor de 57 variables, incluida la variable de interés llamada PUNTAJE.

2.1.1 Correlación de Pearson

La determinación de la correlación de Pearson nos sirve para evidenciar qué variables tienen una mayor influencia sobre la variable de respuesta PUNTAJE. A continuación se presentan los resultados de dicha correlación, teniendo en cuenta que se eligieron las 30 variables con mayor influencia sobre la variable PUNTAJE.

PUNTAJE	1.000000	INGRESOS	0.195066
ESTRATO	0.352149	DUCHA	0.188893
TPERSONA	0.335194	ACUEDUC	0.186207
LAVADORA	0.329305	THOGAR	0.165095
ZONA	0.318070	TCUARTOSVI	0.162389
TCUARTOS	0.307183	TIPODOC	0.159893
VIVIENDA	0.299863	TSANITAR	0.158085
NIVEL	0.299811	MOTO	0.157553
ALCANTA	0.285275	TVCOLOR	0.151427
TVCABLE	0.282250	HIJOSDE	0.142096
COMPUTADOR	0.271075	PERCIBE	0.139172
SANITAR	0.270883	AUTO1	0.132105
NEVERA	0.261723	USOTELE	0.130389
GRADO	0.257657	ESTCIVIL	0.127800
GAS	0.256582	HOGAR	0.126723
TELEFONO	0.248966	COCINAN	0.120431
HORNO	0.229138	PARENTES	0.113926
EQUIPO	0.224590	ACTIVI	0.079633
TDORMIR	0.211229	DISCAPA	0.077668
COCINA	0.209659	BASURA	0.053970
CALENTA	0.199708	CARNET	0.051045
		ELIMBASURA	0.050493
		AIRE	0.042774
		TENEVIV	0.042205

Se filtraron los datos, formando un nuevo data frame que contuviera las 30 variables con la correlación más alta con la variable puntaje.

2.2 Cumplimiento de requisitos

El dataset original debió ser modificado para cumplir con las condiciones especificadas en el proyecto (al menos un 5% de datos faltantes en al menos 3 columnas), se realizó el reemplazo del 5% de los datos de 3 columnas por el valor de NaN, es decir, como datos nulos. En este sentido, cada columna contenía 447872 registros, lo que en un 5% equivale a 22393 datos a eliminar de cada columna. Se eliminaron datos de las variables denominadas BUSCANDO (Semanas que lleva buscando trabajo), TSANITAR (Cantidad de sanitarios del hogar) y TPERSONA (Cantidad de personas que habitan la vivienda). A continuación se presenta la tabla donde se visualizan los valores nulos para cada variable:

```
Columna 'TCUARTOSVI': 0 valores nulos
Columna 'THOGAR': 0 valores nulos
Columna 'HOGAR': 0 valores nulos
Columna 'TENEVIV': 0 valores nulos
Columna 'TCUARTOS': 0 valores nulos
Columna 'TDORMIR': 0 valores nulos
Columna 'SANITAR': 0 valores nulos
Columna 'TSANITAR': 23377 valores nulos
Columna 'DUCHA': 0 valores nulos
Columna 'COCINA': 0 valores nulos
Columna 'COCINAN': 0 valores nulos
Columna 'ALUMBRA': 0 valores nulos
Columna 'USOTELE': 0 valores nulos
Columna 'NEVERA': 0 valores nulos
Columna 'LAVADORA': 0 valores nulos
Columna 'TVCOLOR': 0 valores nulos
Columna 'TVCABLE': 0 valores nulos
Columna 'CALENTA': 0 valores nulos
Columna 'HORNO': 0 valores nulos
```

```
Columna 'EQUIPO': 0 valores nulos
Columna 'MOTO': 0 valores nulos
Columna 'AUTO1': 0 valores nulos
Columna 'BIERAICES': 0 valores nulos
Columna 'TPERSONA': 23369 valores nulos
Columna 'TIPODOC': 0 valores nulos
Columna 'PARENTES': 0 valores nulos
Columna 'ESTCIVIL': 0 valores nulos
Columna 'HIJOSDE': 0 valores nulos
Columna 'SEXO': 0 valores nulos
Columna 'FECHANTO': 0 valores nulos
Columna 'DISCAPA': 0 valores nulos
Columna 'CARNET': 0 valores nulos
Columna 'ASISTE': 0 valores nulos
Columna 'GRADO': 0 valores nulos
Columna 'NIVEL': 0 valores nulos
Columna 'ACTIVI': 0 valores nulos
Columna 'BUSCANDO': 23349 valores nulos
Columna 'PERCIBE': 0 valores nulos
Columna 'INGRESOS': 0 valores nulos
Columna 'PAGAPOR': 0 valores nulos
Columna 'PUNTAJE': 0 valores nulos
```

Como se puede observar, las variables TSANITAR, BUSCANDO y TPERSONA contienen valores nulos equivalentes a, al menos, el 5% de los datos de cada una de ellas. Posteriormente se procedió a calcular el promedio de los datos de las variables TSANITAR, BUSCANDO y TPERSONA y con ello se reemplazaron los valores nulos por dichos promedios. El valor de los promedios es usado para realizar la imputación de datos, sus valores se encuentran a continuación:

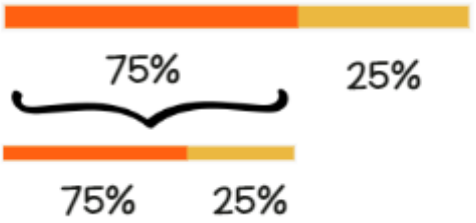
```
Promedio de BUSCANDO: 0.7932545309783712
Promedio de TSANITAR: 1.0431172172044838
Promedio de TPERSONA: 5.166114647280815
```

Tras el procedimiento observamos:

```
Cantidad de nulos: 0
```

3. Modelos de machine learning

En la primera iteración se emplearon los modelos de regresión lineal, árbol de decisión y bosque aleatorio. Para tal fin, se dividieron aleatoriamente los datos en dos conjuntos, un conjunto de entrenamiento equivalente al 75% de los datos y un conjunto de prueba equivalente al 25% (Test set) de los datos, este primer 75% se dividió nuevamente entre un 75% (Training set) y un 25% (Validation set) Tal y como se muestra en la siguiente imágen:



3.1 Modelos supervisados

Los resultados en la primera iteración para dichos modelos se presentan a continuación:

Modelo	Hiper Parámetros	Mean Squared Error (MSE)	R-squared (r^2)
Regresión lineal	-	113.59	0.53
Árbol de decisión	maxDepth= None	146.71	0.40
Bosque aleatorio	n_estimators= 100, maxDepth= None	98.58	0.59

En resumen, el modelo de Bosque Aleatorio parece tener el mejor rendimiento de los tres modelos evaluados en términos de MSE y R^2. Esto significa que es capaz de hacer predicciones más precisas y explicar una mayor proporción de la variabilidad en la variable PUNTAJE en comparación con los otros dos modelos, sin embargo, resulta conveniente realizar evaluaciones con otros modelos, pues los porcentajes de variabilidad explicada por los modelos abordados hasta ahora aún resultan ser relativamente bajos.

En vista de los bajos valores de las métricas elegidas con anterioridad (MSE y R^2), se decidió evaluar también como métrica de desempeño el error logarítmico cuadrático medio, con el fin de penalizar más fuertemente las predicciones que están lejos del valor real en comparación con las que están cerca. Esta métrica es una versión modificada de la métrica de error cuadrático medio (RMSE), pero en lugar de medir la diferencia absoluta entre las

predicciones y los valores reales, mide la diferencia relativa después de aplicar el logaritmo a ambos.

Se realiza una comparación de los tres modelos anteriormente evaluados (lineal, árbol de decisión y bosque aleatorio), utilizando la validación cruzada para evaluar su rendimiento. Luego, seleccionamos el modelo con el menor error logarítmico cuadrático medio en el conjunto de validación cruzada como métrica de desempeño. A continuación se presentan los resultados:

Regresión lineal

Árbol de decisión

Bosque aleatorio

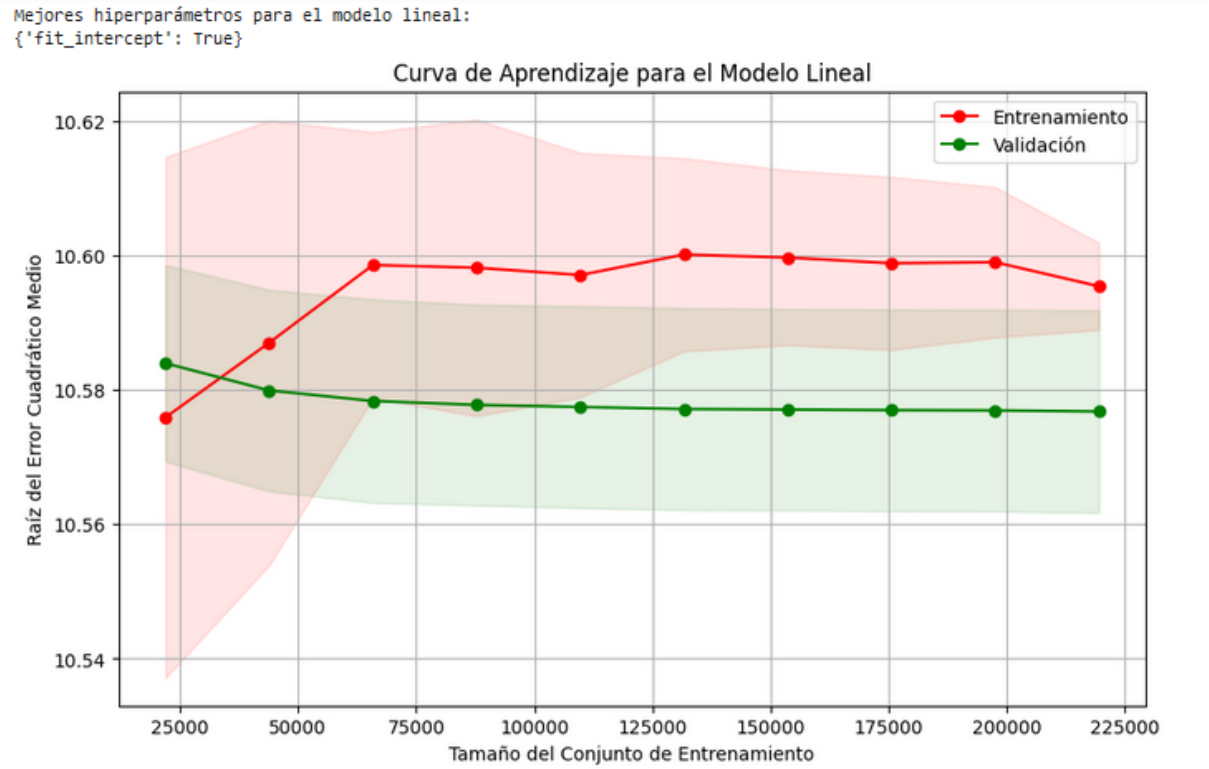
```
---------
RMSLE Test:  10.60306 (± 0.01722206 )
RMSLE Train:  10.58257 (± 0.00741121 )
---------
RMSLE Test:  11.89565 (± 0.01735391 )
RMSLE Train:  11.89229 (± 0.01212642 )
---------
RMSLE Test:  11.83503 (± 0.02647352 )
RMSLE Train:  11.84183 (± 0.02117346 )
Seleccionado:  0

 Mejor modelo:
LinearRegression()
```

En este caso, se aprecia que el modelo con menor valor de error logarítmico cuadrático medio corresponde a la regresión lineal, seguido por el modelo de bosque aleatorio. Con estos modelos se procede a obtener los mejores hiper parámetros y elaborar la respectiva curva de aprendizaje.

3.1.1 Modelo de regresión lineal

En la siguiente figura se presentan los mejores hiperparámetros y la curva de aprendizaje para el modelo de regresión lineal:

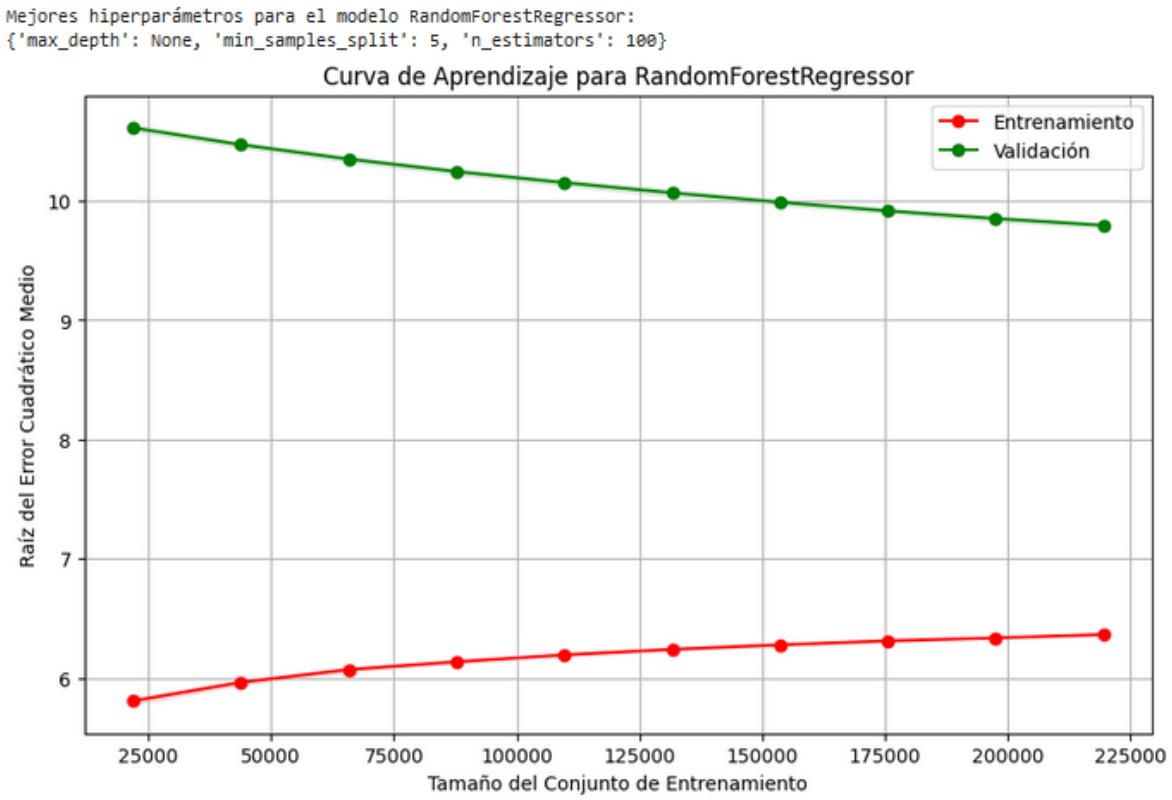


En este caso, se aprecia que durante la búsqueda de hiper parámetros, se determinó que incluir el término de intercepción (fit_intercept=True) produce un rendimiento óptimo para el modelo.

En la curva de aprendizaje, se observa que el error durante la validación resulta ser menor que el de prueba, lo que no suele ser frecuente, pues el modelo está construido específicamente para los datos de entrenamiento; en este sentido, el modelo está funcionando mejor con los datos de validación. Adicionalmente, debe existir cierta convergencia en el error entre ambos grupos de datos, lo que es relativamente apreciable si se tiene presente que la escala en la que se encuentran ambos está entre aproximadamente 10,57 y 10,60. Otro aspecto a resaltar es que, el error se estabiliza en ambos casos, lo que implica que agregar más datos no necesariamente incrementará significativamente el rendimiento del modelo.

3.1.2 Random forest regressor

En la siguiente figura se presentan los mejores hiper parámetros para el modelo Random forest regressor, así como su curva de aprendizaje:



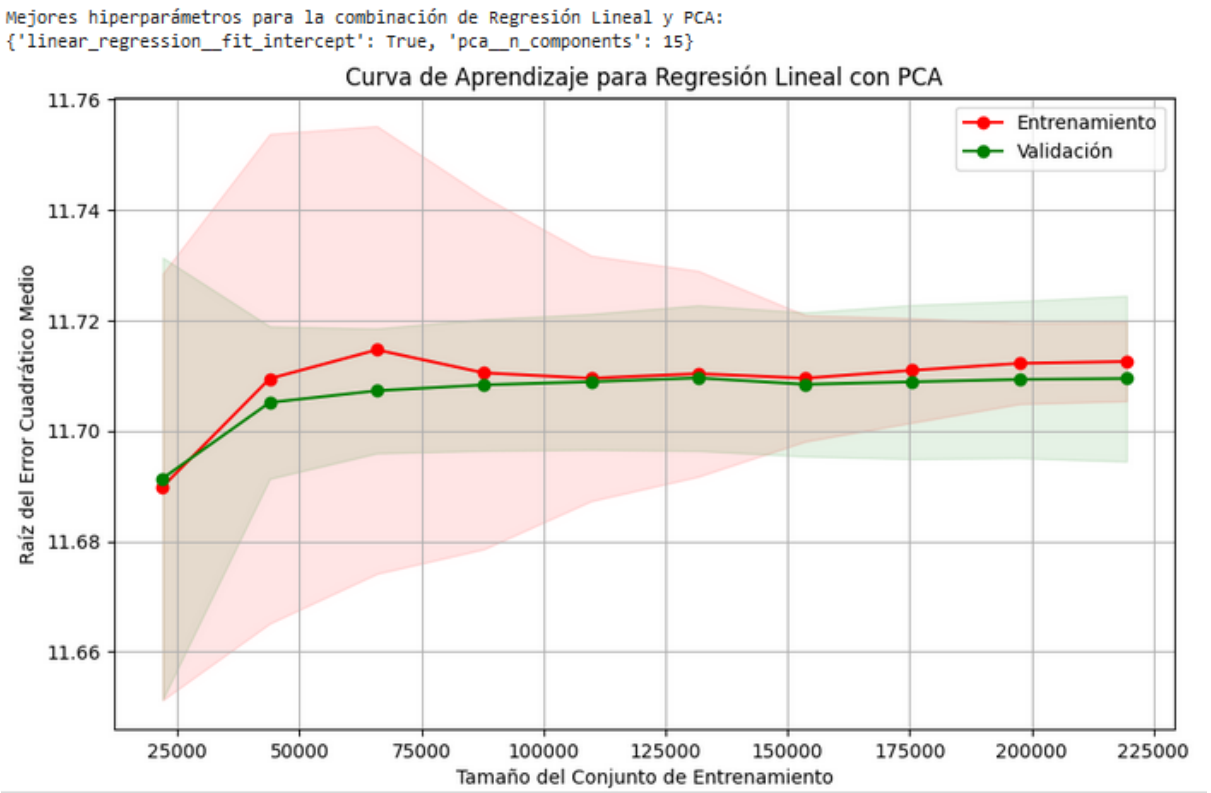
En cuánto a los mejores hiperparametros, inicialmente se evidencia que no hay restricción en la profundidad máxima de los árboles en el bosque. Los árboles pueden crecer hasta que contengan un número mínimo de muestras en cada hoja o hasta que se alcance otro criterio de detención.Se ha determinado también que un valor de 5 es óptimo para el número mínimo de muestras requeridas para dividir un nodo interno. Finalmente, se ha encontrado que un bosque con 100 árboles proporciona un rendimiento óptimo.

En cuánto a la curva de aprendizaje, se aprecia una amplia separación entre el valor del error para el conjunto de entrenamiento, respecto del conjunto de validación. Esta situación es problemática, pues es un síntoma de sobreajuste (overfitting). La anterior situación puede encontrarse debida, posiblemente, a la profundidad máxima no restringida, empleada en los árboles, pues ello puede conllevar a modelos más complejos, que pueden propiciar que el modelo se ajusta demasiado a los datos de entrenamiento y captura patrones específicos de esos datos que no generalizan bien a nuevos datos.

3.2 Combinación de modelos no supervisados con modelos supervisados

3.2.1 Combinación de regresión lineal con Análisis de Componentes Principales (PCA)

En la siguiente figura se aprecian los mejores hiperparámetros para el modelo combinado de regresión lineal con PCA, así como su curva de aprendizaje:



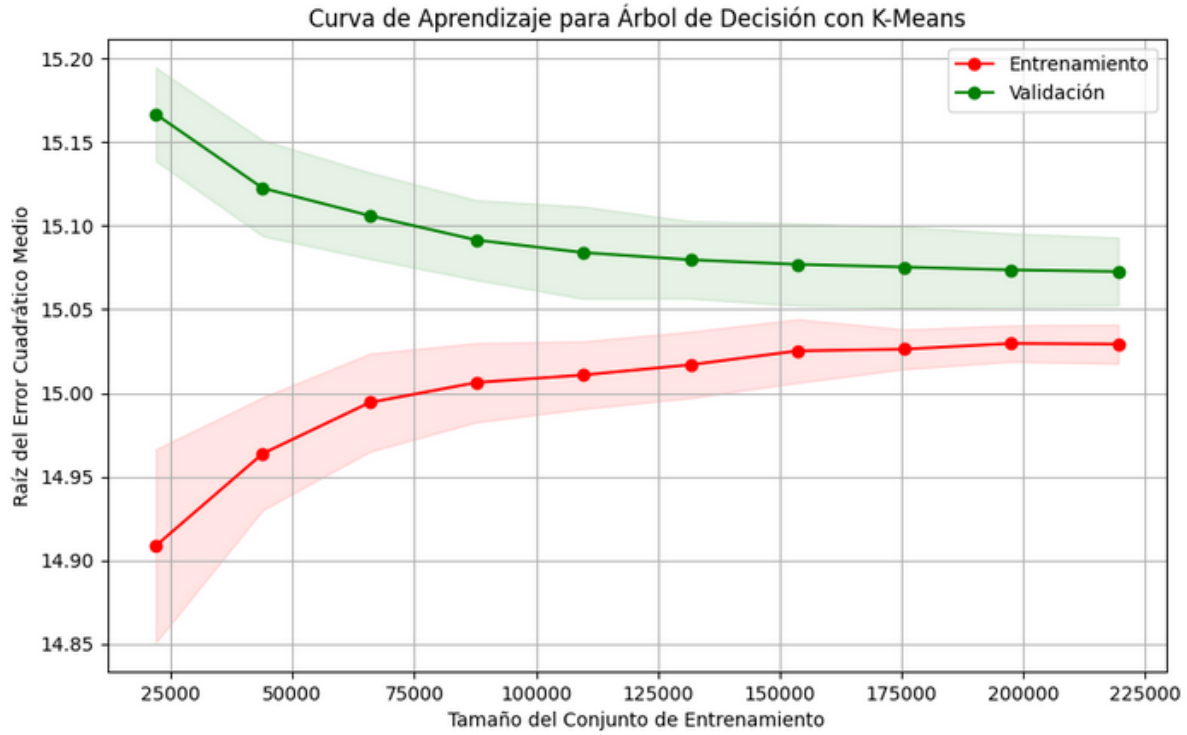
En cuánto a los hiperparámetros obtenidos, es de resaltar el hecho de que el número óptimo de componentes principales es de 15, lo que implica una significativa reducción de dimensionalidad, al pasar de 30 variables a 15 variables latentes.

En la curva de aprendizaje se puede observar que ambos errores, el de entrenamiento y el de prueba, convergen a medida que se incrementa el tamaño del conjunto de datos de entrenamiento. Esto sugiere que el modelo está aprendiendo de manera efectiva tanto de los datos de entrenamiento como de los datos no vistos durante la fase de entrenamiento. Adicionalmente, a medida que aumenta la cantidad de datos de entrenamiento, el error tiende a estabilizarse, lo que indica que agregar más datos no necesariamente mejorará significativamente el rendimiento del modelo. En cuanto al tamaño óptimo de entrenamiento, para este modelo, se observa que un tamaño bajo, de aproximadamente 25000 datos, genera el menor error posible.

3.2.2 Árbol de decisión con K-means

En la siguiente figura se presentan los mejores hiperparámetros para el modelo combinado de árbol de decisión con K-means. Adicionalmente, se presenta la curva de aprendizaje para dicho modelo:

```
Mejores hiperparámetros para la combinación de Árbol de Decisión y K-Means:
{'decision_tree__max_depth': 10, 'kmeans__n_clusters': 5}
```



En lo relacionado con los mejores hiperparámetros, se obtuvo una profundidad máxima óptima del árbol de decisión, de 10. Por otra parte, se ha determinado que el número de clústeres óptimo para K-Means es 5 para el rendimiento global del modelo.

En cuánto a la curva de aprendizaje, la separación inicial del error del conjunto de validación frente al conjunto de entrenamiento, puede ser síntoma de un sobreajuste, especialmente en tamaños bajos de conjunto de entrenamiento. La estabilización posterior, sugiere que el modelo ha alcanzado su límite en términos de capacidad de generalización. Es decir, después de cierto punto, el modelo deja de mejorar significativamente en términos de rendimiento en ambos conjuntos (entrenamiento y validación).

4. Retos y condiciones de despliegue del modelo

Los modelos evaluados deben permitir que la tasa de clasificación de potenciales usuarios del Sisbén en un periodo de tiempo determinado, sea superior en, al menos, un 20% de la tasa

de clasificación de usuarios con la anterior metodología, con el fin de que valga la pena emplear la propuesta aquí presentada, pues está encaminada hacia la reducción de trabajo y tiempo en la clasificación. En este sentido, para poder desplegar el modelo, se requiere que cada encuestador cuente con un software que le permita ingresar cada uno de los datos de las variables consideradas como relevantes en la presente propuesta y que, tras realizar la respectiva encuesta, el software le permita obtener la asignación de puntaje inmediata. Bajo esta perspectiva, el software de clasificación debería contar con las siguientes condiciones de despliegue:

1. Interfaz de usuario intuitiva
2. Utilizar técnicas de encriptación de los datos para su protección
3. Ser escalable
4. Ser integrable con sistemas ya existentes como bases de datos gubernamentales
5. Contar con documentación clara

Conclusiones

Se concluye que, durante la búsqueda de hiperparámetros para el modelo de regresión lineal, la inclusión del término de intercepción (fit_intercept=True) se determina como óptima. Además, se destaca la convergencia entre el error de validación y prueba, lo cual es un hallazgo valioso, indicando que el modelo generaliza bien a datos no vistos durante el entrenamiento.

Se identifica un posible problema de sobreajuste en el modelo Random Forest Regressor, evidenciado por la amplia separación entre el error del conjunto de entrenamiento y el de validación en la curva de aprendizaje. Esta situación puede deberse a la falta de restricción en la profundidad máxima de los árboles, lo que permite modelos más complejos que se ajustan demasiado a los datos de entrenamiento.

Para el modelo de regresión lineal con PCA, se destaca que el número óptimo de componentes principales es 15, implicando una significativa reducción de dimensionalidad. La curva de aprendizaje muestra convergencia de errores a medida que aumenta el tamaño del conjunto de datos, indicando eficaz generalización. En el modelo combinado de árbol de decisión con K-Means, la optimización de la profundidad del árbol y el número óptimo de clústeres es crucial. La curva de aprendizaje revela una fase inicial de sobreajuste, seguida de estabilización, indicando límites en la capacidad de generalización. Estas observaciones resaltan la importancia de ajustar hiperparámetros considerando la complejidad del modelo y su capacidad de generalización, proporcionando información valiosa para la eficiente implementación práctica en entornos del mundo real.