DEPARTAMENTO DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

Trabajo práctico 02

Verano 2025

Integrante	LU	Correo electrónico
Ballera, Alexander	668/24	alexballera@gmail.com

Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón 0+∞)

Intendente Güiraldes 2610 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (++54 +11) 4576-3300

http://www.exactas.uba.ar

Resumen

En el presente informe se realiza un análisis de clasificación y selección de modelos, utilizando el dataset MNIST-C. El dataset consta de columnas correspondiente a pixeles de imágenes de dígitos del 0 al 9 en escala de grises. La primera parte del informe consta de un análisis exploratorio de los datos, en la que se analiza la cantidad de datos, cantidad y tipos de atributos, cantidad de clases de la variable de interés (dígitos) y otras características. La segunda parte consta de clasificación binaria y por último análisis multiclase en la que se realiza un análisis de selección de modelo mediante validación cruzada con k-folding.

Introducción

El análisis exploratorio de datos (AED) es un proceso que utiliza estadísticas y gráficos para conocer los datos y explorar su naturaleza. Es un paso previo necesario antes de realizar cualquier análisis estadístico o predictivo. Con el AED buscamos: explorar, investigar y aprender de los datos; identificar posibles errores; revelar la presencia de valores atípicos; comprobar la relación entre variables y realizar un análisis descriptivo de los datos.

Con esta información exploratoria y descriptiva de los datos buscamos clasificar y escoger modelos que nos permitan predecir resultados esperados de acuerdo a los datos observados.

Análisis exploratorio

El dataset consta de **70.000 filas y 786 columnas**, del tipo número entero, sin campos vacíos ni nulos; el dataset presenta sus atributos con completitud del 100%.

Descripción y preparación de los datos

El dataset consta de 786 columnas (atributos) , la primera columna etiquetada **'Unnamed: 0'** es del tipo numérico incremental en 1, idéntica al index del conjunto, con lo cual para los efectos del análisis la excluyo del dataset al momento de preparar los datos.

Las siguientes 784 columnas, son del tipo numérico, etiquetadas de manera numérica, parten desde 0 y se extiende hasta la columna 783. El total de las columnas corresponde a una imagen de tamaño 28 x 28 = 784. Cada celda tiene valores que varían desde 0 (mínimo) hasta 255 (máximo), cada celda con estos valores representa un pixel de una imagen.

Cada fila o tupla del dataset corresponde a una imagen de tamaño 28 X 28, donde cada celda corresponde a un pixel en escala de grises cuyo valor varía desde 0 hasta 255 el cual corresponde al color de cada pixel. Los 784 pixeles, para construir la imagen se distribuyen de izquierda a derecha y de arriba hacia abajo, en 28 columnas y 28 filas, tal como lo podemos observar a continuación en la **Figura 1**:

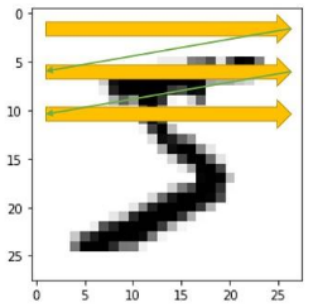

Fig. 1: Distribución de píxeles

La última columna, etiquetada como **'labels"** contiene la etiqueta de cada fila (el número representado en la imagen). Los valores de esta columna varían desde 0 hasta 9. Esta columna la excluyo del dataset y construyo otro subconjunto con la cual se realizaran los análisis descriptivos, de clasificación y predicciones.

Análisis descriptivo

En el anexo en la **Figura 2** se observa un cuadro con datos descriptivos de los datos por cada dígito. Cada dígito tiene un mínimo para un pixel de valor cero (0) y máximos que varían desde 253 correspondiente a pixeles del dígito de valor uno (1) y valores hasta 255 para pixeles de dígitos correspondiente a los dígitos 3 y 9. Cada grupo de datos por dígito presentan sus datos completos, no poseen celdas vacías o con valores nulos.

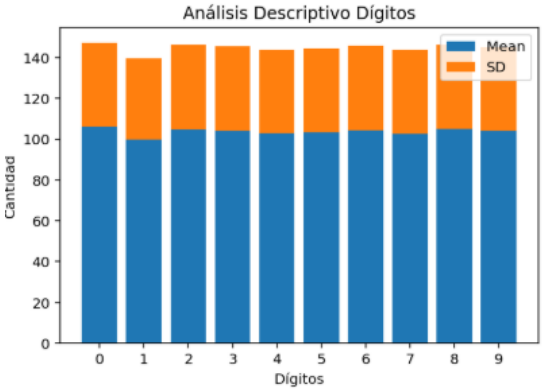

Fig 3: Análisis descriptivo de dígitos

En cuanto a las medidas de tendencia central y dispersión como la media, desviación estándar y coeficiente de variación de cada dígito se observa gran dispersión, y es de esperarse porque cada celda representa un pixel cuyo valor varía desde cero (mìnimo) hasta 255 (máximo).

La **Figura 3** muestra las medias y desviación estándar por dígito. Con estas métricas se pretende determinar cuán concentrado o dispersos están los valores (pixeles). Sin embargo, por la misma característica de lo que representa cada celda (un pixel), es irrelevante cuán centrados o dispersos se presentan los diferentes pixeles de la imagen.

En la **Figura 4** se pueden observar la cantidad de tuplas por cada dígito, se observa cierta similitud en cuanto a cantidad. Los dígitos uno (1) y siete (7) son los que presentan mayor cantidad de casos con 7.877 y 7.293 respectivamente y el dígito cinco (5) con la menor cantidad en 6.133.

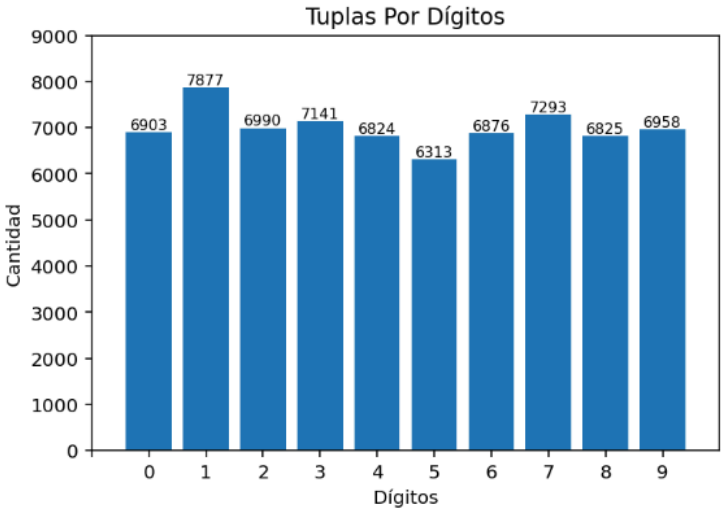

Figura 4: Cantidad de tuplas por dígitos

Algunos resultados de las imágenes

A continuación se muestran algunos dígitos, el primer grupo se muestran desde 0 hasta 9, el segundo grupo se muestran algunas imágenes aleatorias y finalmente se agrupan los datos por 'labels' y se muestran las imágenes sacando el promedio de cada subconjunto de dígitos.

Tomando las siguientes filas del dataset 0,1,2,3,4,5,7,13,15,17 se grafican cada uno de ellos. Se escogen estas filas ya que representan los dígitos etiquetados como 0,1,2,3,4,5,6,7,8,9 tal como se muestra a continuación en la **Figura 5**:

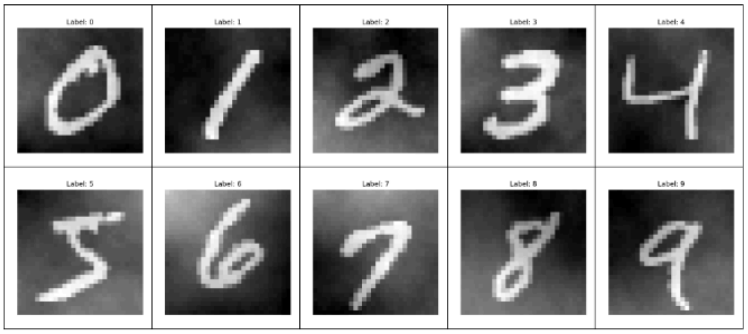

Fig. 5: Dígitos de 0 a 9

A continuación en la **Figura 6** se muestran algunos dígitos tomando filas de manera aleatoria.

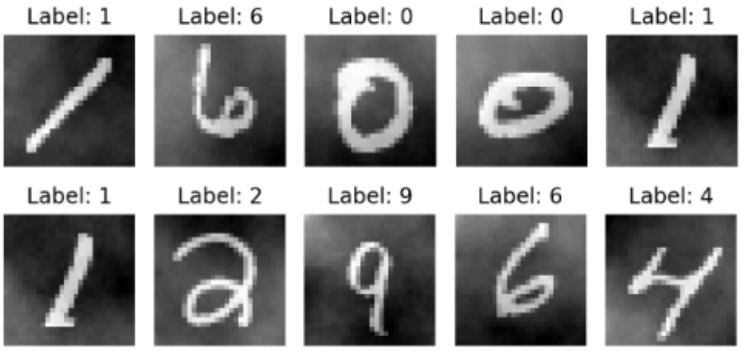

Fig. 6: Dígitos aleatorios

Finalmente se agrupan los datos por 'labels' y a cada subconjunto se grafican los promedios por bloques, el resultado se muestra a continuación en la **Figura 7**:

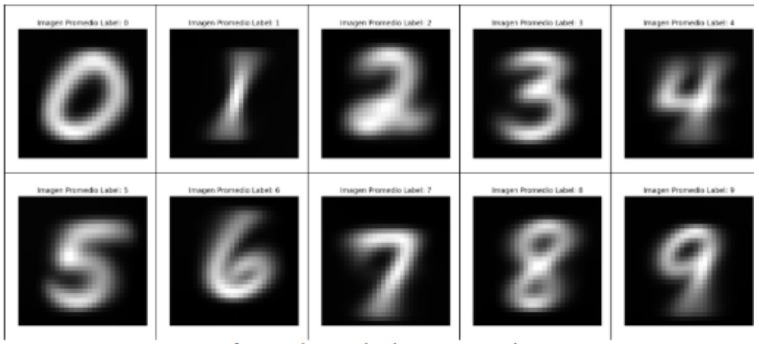

Fig. 7: Dígitos de datos promedio

a. Atributos

En la sección descripción y preparación de datos se describe detalladamente las columnas o atributos del dataset.

b. Números parecidos

El dataset muestra números parecidos que visualmente pueden presentar confusión, tales como 1 y 7. En las Figuras 9 y 11 se observan claramente los parecidos de estos dígitos.

c. Digitos similares

Además de lo visible, de los números parecidos, se evalúan los datos correspondiente a los pixeles y según la variación de cada línea o de cada imágen de cada dìgito se puede observar qué tan similar o diferente es cada un. En el **Cuadro 1** mostrado a continuación, se observa un resumen que incluye de filas y columnas Media (promedio de promedio), SD (promedio de la desviación standard), CV evalúo el coeficiente de variación que mide la proporción o variación de la SD respecto de la media de cada dígito, arrojando los siguientes valores:

	0	1	2	3	4	5	6	7	8	9
Media	106.11	99.71	104.71	104.13	102.90	103.28	104.53	102.79	104.96	103.88
SD	41.32	40.03	41.51	41.29	41.05	41.35	41.54	40.98	41.27	41.20
CV	0.39	0.40	0.40	0.40	0.40	0.40	0.40	0.40	0.39	0.40

Cuadro 1: Cuadro de análisis descriptivo del dataframe.

Según el **Coeficiente de Variación**, se observa variaciones similares para cada dataset por dígito, esto se debe a que las cantidades de líneas son similares como se observó anteriormente en la **Figura 4**, además en la Figura 3 también se observa similitud entre las medias y desviación estandard por subconjuntos de dìgitos. A continuación, en las **Figuras 8, 9, 10, 11 y 12** se muestran imágenes correspondiente a las clases 0, 1, 5, 7 y 8.

Algunos ceros (0):

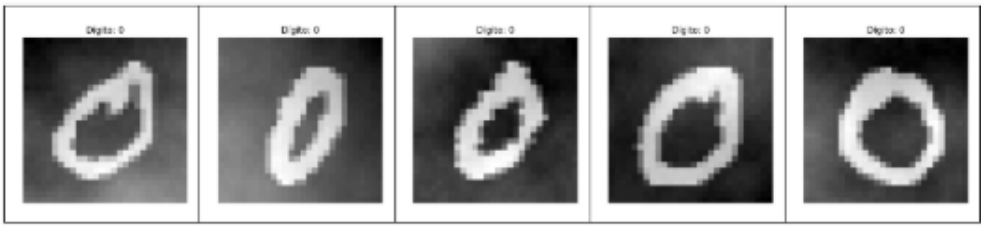

Fig. 8: Dígitos cero

Algunos unos (1):

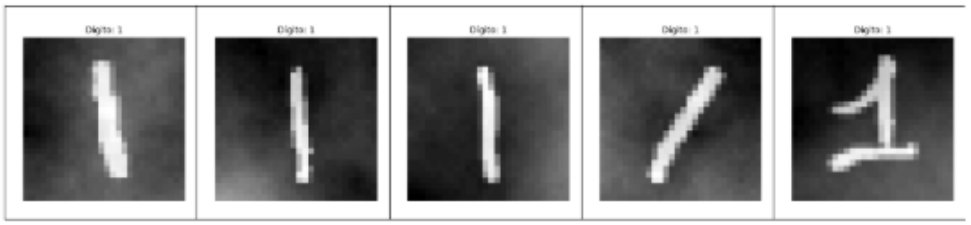

Fig. 9: Dígitos uno

Algunos cincos(5):

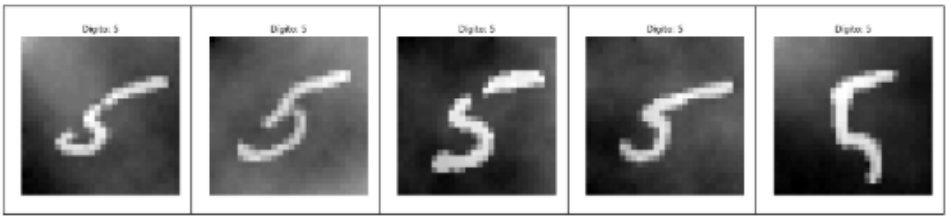

Fig. 10: Dígitos cinco

Algunos siete(7):

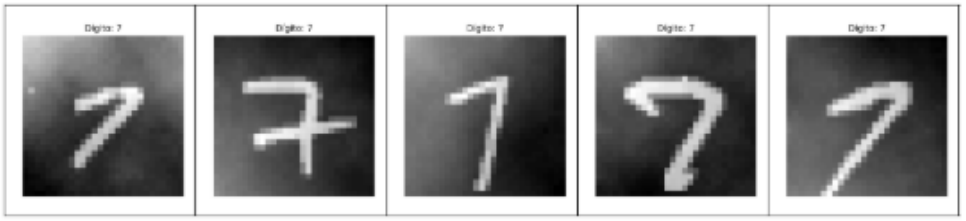

Fig. 11: Dígitos siete

Algunos ocho(8):

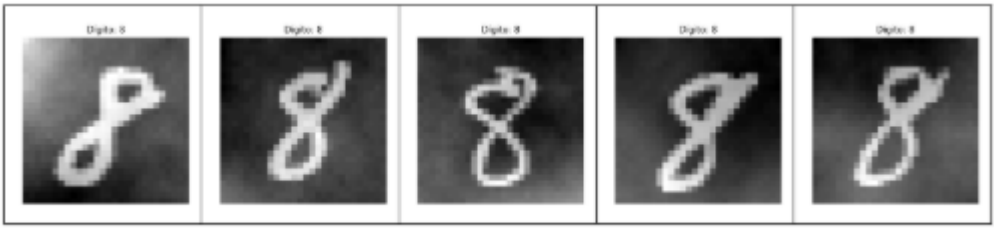

Fig. 12: Dígitos ocho

d. Exploración de datos

La manera de realizar un análisis exploratorio de este dataset es diferente a lo visto en clases. La principal diferencia es que cada columna representan pixeles de la imagen, con lo que todas las columnas están correlacionadas para mostrar un resultado final que es la imagen de un dígito.

A diferencia de otro dataset como el de Titanic en la que cada atributo representa una característica de la realidad y/o de la entidad. Cada campo representa el color del pixel con dígitos que varían de 0 a 255 implica una dispersión considerable de los datos cuando los comparamos con las medidas de tendencia central, esta dispersión se evidencia por la amplitud del rango de cada columna.

Clasificación binaria

a. Subconjuntos imágenes correspondiente a dígitos cero (0) y uno (1)

En las **Figuras 8 y 9** se muestran algunas imágenes de ceros y unos.

Ambos subconjuntos poseen los datos de sus campos completos y no nulos, además el coeficiente de variación (CV) de ambos está por encima del 30%, lo que indica que estadísticamente no son homogéneos, ya que presentan gran dispersión a la media.

Estos valores de CV indica que las imágenes se van a presentar con bastante variación en cada tupla que se grafique, la imagen cero (0) al presentar un CV menor va a presentar menos diferencias en sus imágenes, tal como se muestra en las en las **Figuras 13 y 14** de la sección Digitos similares del presente informe.

La **Figura 13**, muestra la cantidad de muestras para los dígitos cero (0) y uno (1) respectivamente, se observa una diferencia de 974 casos del dígito uno respecto de cero, corresponde a una diferencia del 14,11% superior y por último la cantidad del dígito cero corresponde en proporción el 87,63% de la cantidad del dígito uno.

De igual manera, en la **Figura 14,** se observan las proporciones de cantidad de filas de dígito Ceros y Uno con 46,7% y 53,3% respectivamente, en función del total del subconjunto. Estas diferencias entre las cantidades de una clase respecto de la otra no son significativos para el objetivo de este análisis.

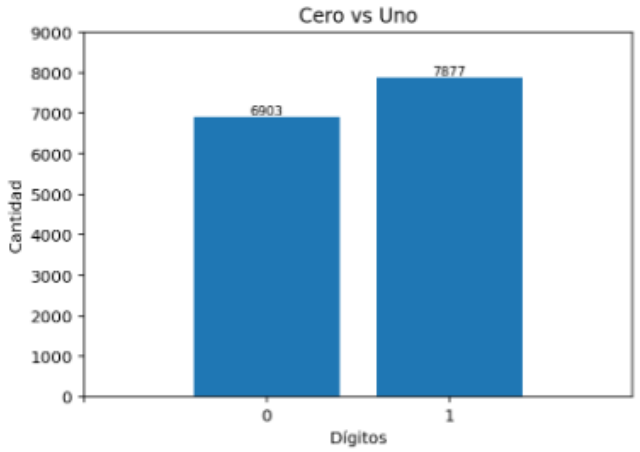

Fig 13: Balance ceros y uno

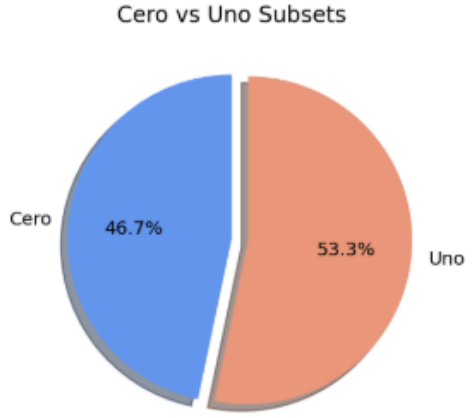

Fig. 14: Proporción ceros y uno

Con base en estos análisis: diferencia de casos, coeficiente de variación, porcentaje o proporción de un subconjunto respecto del otro se puede concluir que las dos clases están balanceadas.

b. Train & test subconjuntos

Se realiza un split del conjunto de datos X y se construyen los subconjuntos train y test, tanto para el subconjunto "X" como para el subconjunto "y", las **Figuras 15 y 16** muestra que los datos de cada subconjunto quedan distribuidos en 70% y 30% tanto para los sets train y test de los conjuntos "X" e "y" respectivamente.

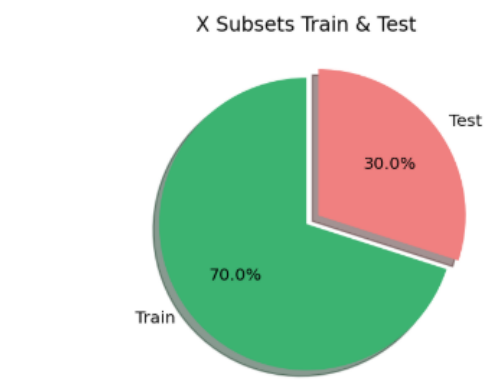

Fig 15: X Subsets Train & Test

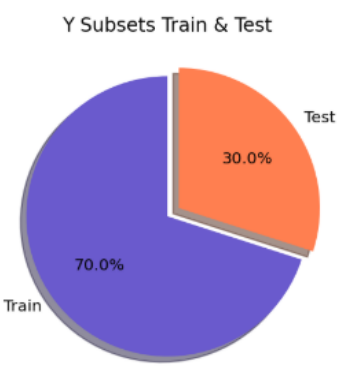

Fig 16: : Y Subsets Train & Test

c. Analizar modelo de KNN, con diferentes atributos: 3 y/o diferentes a 3

Conjuntos 0 y 1, con modelo KNN y k = 3: En primer lugar analizo el modelo **KNN** para las clases cero y uno analizadas en las secciones anteriores. Basado en este conjunto de datos y aplicando el modelo de KNN, se observan en las **Figuras 17 y 18**, Métricas y Matriz de confusión respectivamente, que el modelo predice de manera completa y precisa los valores. Se observa una **accuracy de 0.995264 y precision de 0.991014**, con lo cual el modelo KNN para k = 3 predice de manera completa y precisa el dígito.

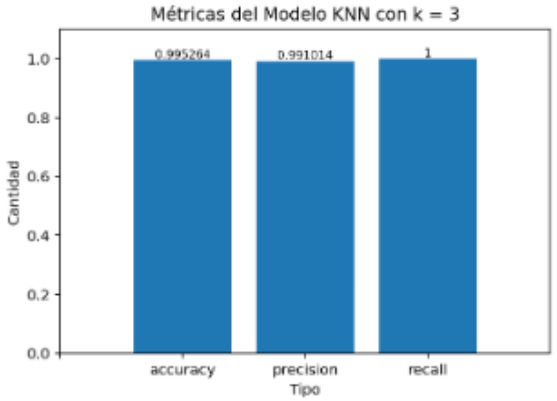

Fig 17: Métricas KNN Model

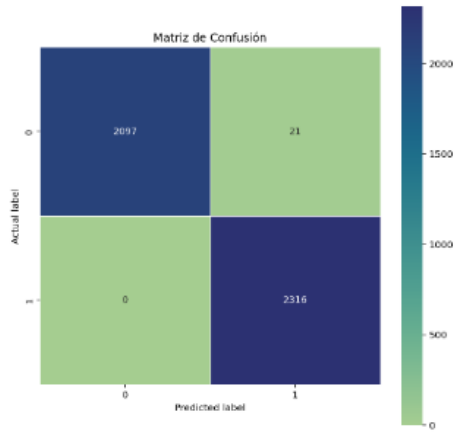

Fig. 18: Matriz de Confusión

Conjunto Cero y Uno, con modelo KNN y k = 3, y con 3 y 10 atributos: Evalúo el modelo del DF de cero y uno y varío la cantidad de atributos de manera random.

Las **Figuras 19 y 20** se muestran las métricas y la matriz de confusión correspondiente a un modelo de KNN con cantidad de vecinos cercanos K igual a 3 y con 3 atributos. Basado en estas mètricas y la matriz de confusión el modelo predice el dígito cero y uno, aún cuando se reduce la cantidad de columnas o pixeles a 3.

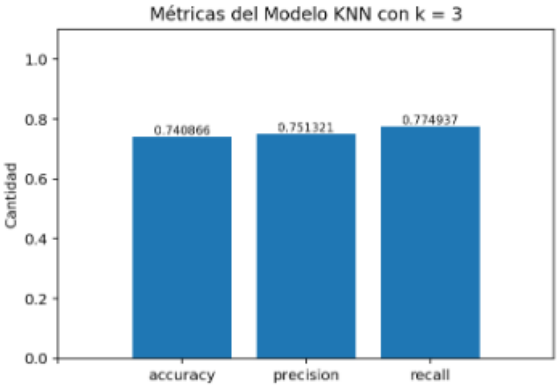

Fig 19: Métricas KNN Model k = 3 atributos = 3

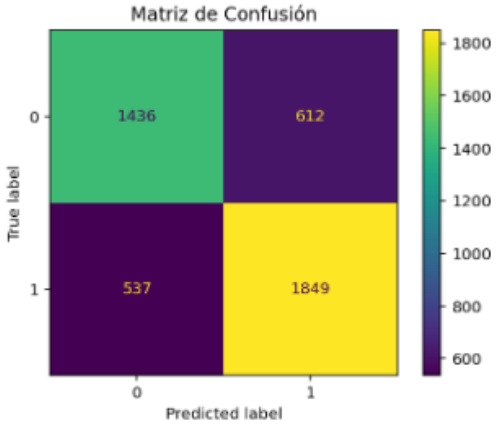

Fig 20: KNN Model k = 3 atributos = 3

Las **Figuras 21 y 22** se muestran las mètricas y la matriz de confusión correspondiente a un modelo de KNN con cantidad de vecinos cercanos K igual a 3 y con 10 atributos. Basado en estas métricas y la matriz de confusión el modelo predice el dígito cero y uno, aún cuando se reduce la cantidad de columnas o pixeles a 10.

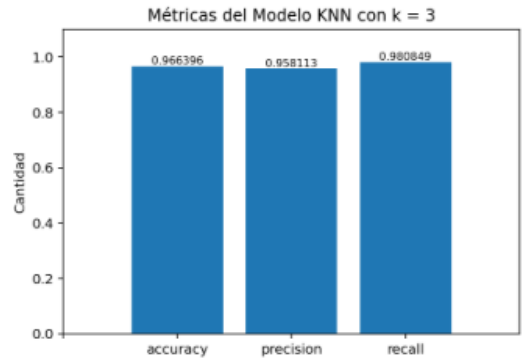

Fig 21: Métricas KNN Model k = 3 atributos = 10

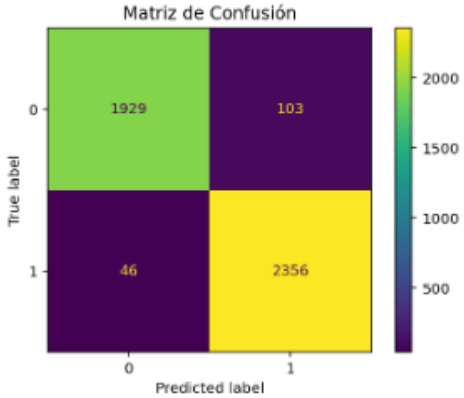

Fig 22: KNN Model k = 3 atributos = 10

Al analizar las métricas y las matrices de confusión variando los atributos o pixeles de 3 y 10 pixeles, para el modelo KNN con vecinos cercanos k igual a 3 se observa claramente que para ambos casos, el modelo con mayor atributos mejora sus métricas de predicción. Aún cuando no se utiliza el total de pixeles, con pocos atributos como lo son 3 y 10 pixeles el modelo puede responder si la imagen corresponde a cero o a uno.

Conjunto total de datos, con modelo KNN y k = 3 y con 3 atributos: Se construye un DF con 3 atributos escogidos de manera random, en este caso arrojó las columnas **79, 530 y 781**. Para estos atributos se analizó con el modelo KNN dando como resultado un **accuracy de 0.1353** lo que verifica que al usar 3 pixeles del total de 785 que constituye una imagen de 28 x 28, el modelo no predice correctamente el valor, se comprueba la incompletitud de los valores. La **Fig. 23**, muestra la matriz de confusión para este modelo y verifica esta observación.

Conjunto total de datos, con modelo KNN y k = 3 y con 10 atributos: Construyo un DF con 10 atributos escogidos de manera random, en este caso arrojó las columnas **41, 91, 81, 177, 206, 233, 252, 285, 287 y 617**. Para estos atributos se analizó con el modelo KNN dando como resultado un **accuracy de 0.1938** lo que verifica que al usar 10 pixeles del total de 785 que constituye una imagen de 28 x 28, el modelo tampoco predice correctamente el valor, se comprueba la incompletitud de los valores. A continuación en la **Fig. 24**, la matriz de confusión del model para 10 atributos confirma esta observación.

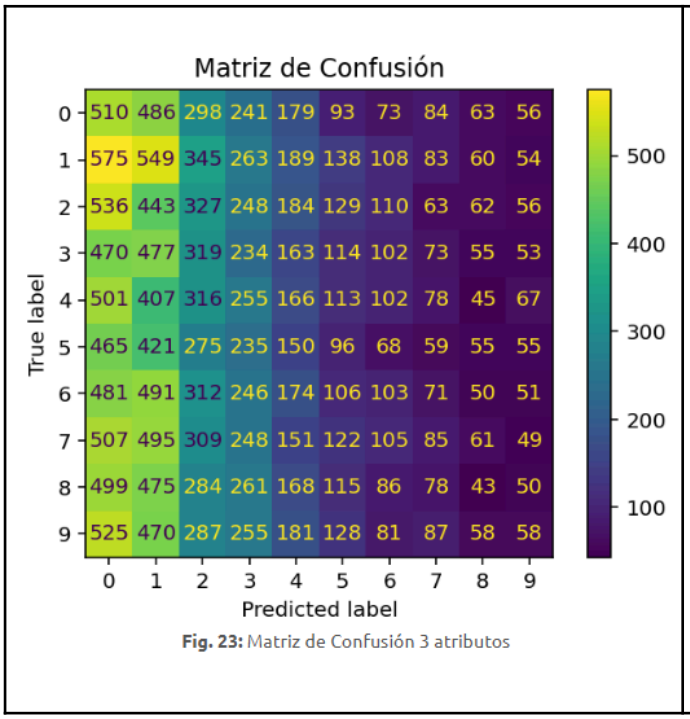

Fig. 23: Matriz de Confusión 3 atributos

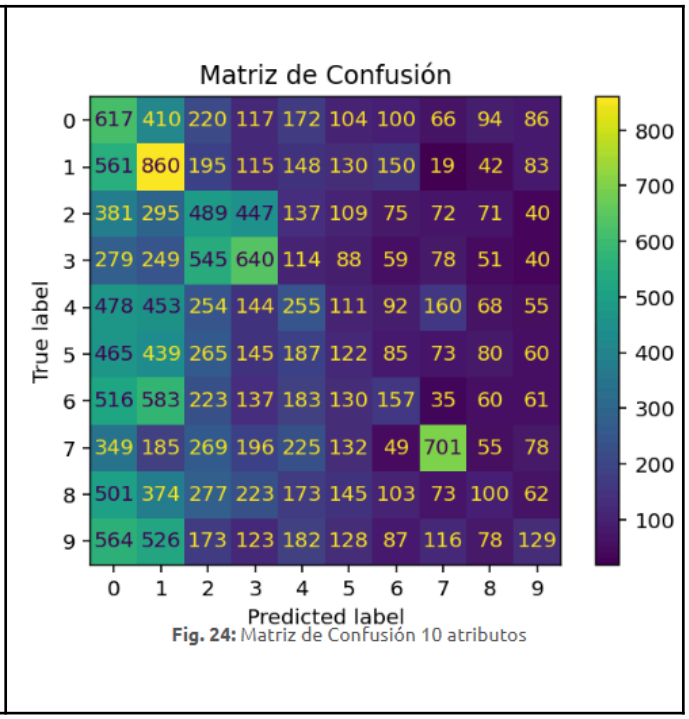

Fig. 24: Matriz de Confusión 10 atributos

d. Analizar modelo de KNN, diferentes atributos y diferentes k

Conjunto Cero y Uno con diferentes atributos y diferentes k: se construye un modelo con el subconjunto Ceros y Unos, se varían los atributos (pixeles) entre 3 y 10 y se varían las cantidades de vecinos cercanos k desde 3 hasta 10, la **Fig. 25** muestra este análisis de sensibilidad.

Para este conjunto de datos el modelo tiene un accuracy superior a 0,8 para aquellos casos en los atributos son superiores a 8. Para los valores k el accuracy es variado, **los valores mayores de accuracy es de 99 para el modelo de 4 atributos y k = 5 y de 98 para el modelo con 6 atributos y k = 5**. El modelo para valores k igual a 5 es la que presenta los mayores valores de accuracy.

Basado a esta matriz de exactitud se observa que para el conjunto de datos analizados (cero y uno), el modelo predice la imagen 0 y 1. Es decir, el modelo responde a la pregunta: ¿la imagen corresponde al dígito 0 o al dígito 1?

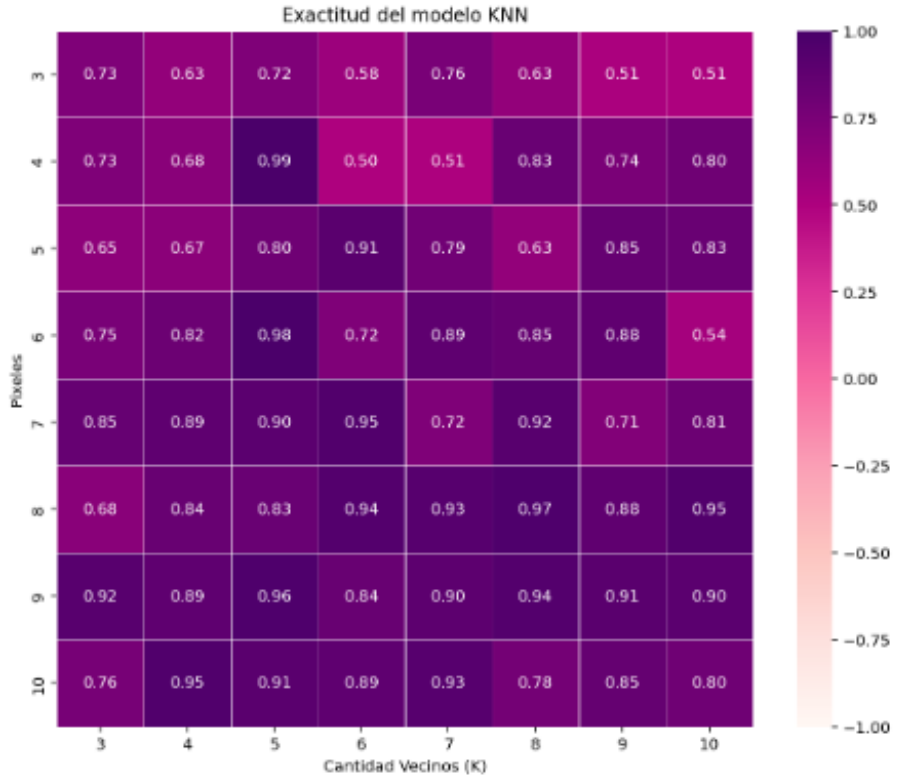

Fig. 25: Accuracy modelo subconjuntos ceros y unos

Conjunto total de datos con diferentes atributos y diferentes k : se construye un modelo con el total de los dígitos, se varían los atributos (pixeles) entre 3 y 10 y se varían las cantidades de vecinos cercanos k desde 3 hasta 10, la **Fig. 26** muestra este análisis de sensibilidad.

Los resultados de accuracy de cada subconjunto y modelo son los que se muestran a continuación, en el siguiente gráfico **Fig 26**, se observan valores accuracy máximos de 40 para el modelo de 8 pixeles con 6 k y de 41 para los modelos correspondientes a 8 y 10 pixeles con 3 y 7 k para el modelo KNN.

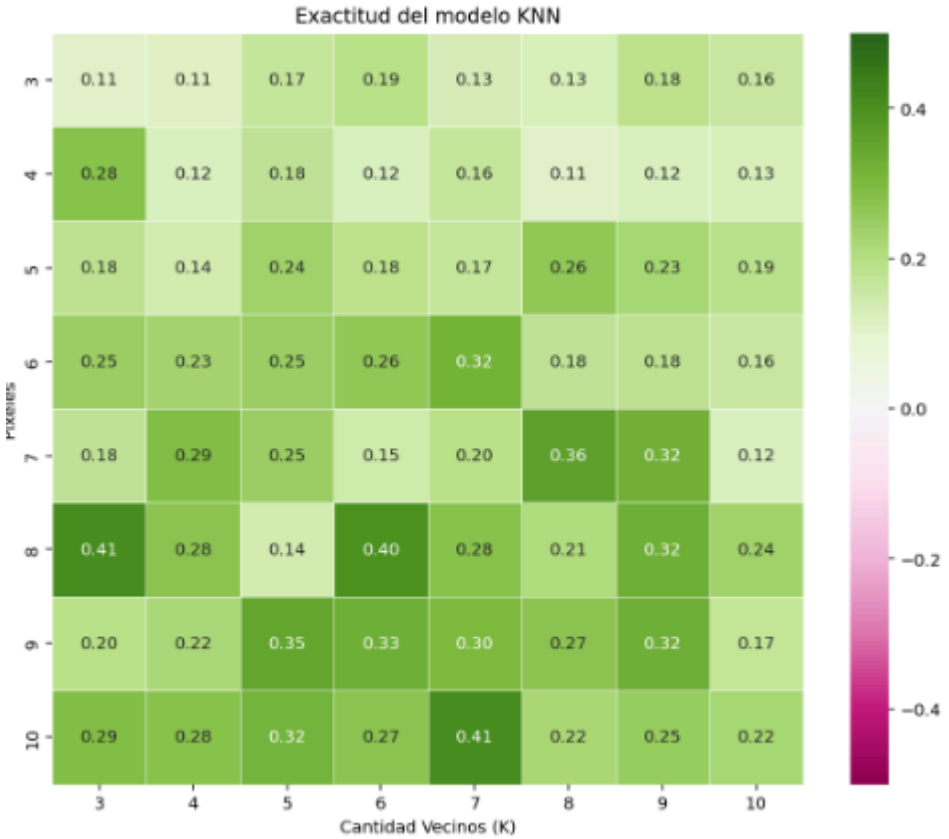

Fig. 26: Accuracy modelo todos los dígitos

En el modelo se observa que la completitud de los datos o la cantidad de atributos tiene relación positiva con la métrica accuracy, mientras más atributos mejora la accuracy, la métrica mejora a medida que se incrementan la cantidad de k o de vecinos cercanos a considerar. La cantidad reducida de pixeles impacta de manera negativa a esta métrica, se puede concluir que el modelo es sensible a la cantidad de pixeles para predecir el dígito.

Dado el análisis de sensibilidad del modelo KNN en la que se varían la cantidad de pixeles y la cantidad de vecinos cercanos podemos concluir en primer lugar que el modelo es sensible a la cantidad de atributos (pixeles) que se introducen para predecir el dígito. Sin embargo, cuando se evalúa el subconjunto de Ceros y Unos la accuracy del modelo es mucho más significativa.

Tal como se concluyó anteriormente, en el análisis de sensibilidad para el subconjunto de ceros y unos, variando la cantidad de pixeles y la cantidad de K en el modelo de KNN, el modelo responde a la pregunta ¿la imagen corresponde al dígito 0 o al dígito 1?.

Como conclusión, el modelo elegido para responder la pregunta "**¿la imagen corresponde al dígito 0 o al dígito 1?"** Es un modelo de clasificación con **KNN** que toma el hiperparámetro k=3 y de atributos se puede escoger 4 pixeles (atributos) o 6 pixeles (atributos), ya que con ambos presenta accuracy de 0,99 y 0,98 respectivamente.

Clasificación multiclase

El conjunto de datos se separó en desarrollo (dev) y validación (held-out), se ajustó el modelo de árbol de decisión y se realizó un experimento en la que se varía la cantidad de kfold en 5, 10 y 15 para distintas profundidades entre 1 a 10 y se escogió el modelo **DecisionTreeClassifier** de la librería **scikit-learn** en Python.

En las **Figuras 27, 28, 29, 30, 31 y 32** del **Apéndice** se muestran las métricas de exactitud para los modelos realizados con el método DecisionTreeClassifier**,** se realizaron modelos de 5, 10 y 15 Folds. Las figuras 27, 28 y 29 se muestran las métricas correspondiente al subconjunto de desarrollo (dev) y las figuras 30, 31 y 32 corresponden a los subconjuntos de validación (held-out).

Al observar las exactitudes del modelo para diferentes Folds, se observa que muestran promedios similares entre ellos, porco mas del 40%.

Y al compararlo con las exactitudes del subconjunto de validación, también se observa que tanto las métricas de exactitud como sus promedios son similares.

Lo que permite concluir que no existe sobreajuste.

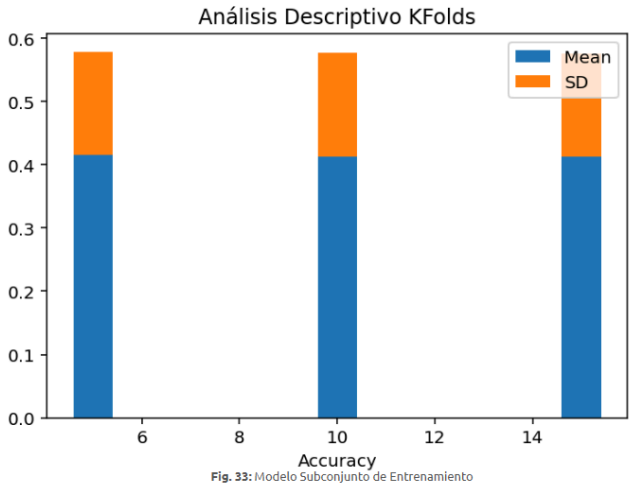

Fig. 33: Modelo Subconjunto de Entrenamiento

Adicionalmente a este experimento, también se experimentó usando el modelo **RandomForestClassifier**, y se realizó validación cruzada. Ene ste caso se realizaron validaciones para iteraciones de 5, 10, 15 y 20 particiones. En la **Figura 34** se pueden observar los resultados tanto del subconjunto de entrenamiento como del conjunto de validación, en este caso el modelo predice de manera completa los valores validadores.

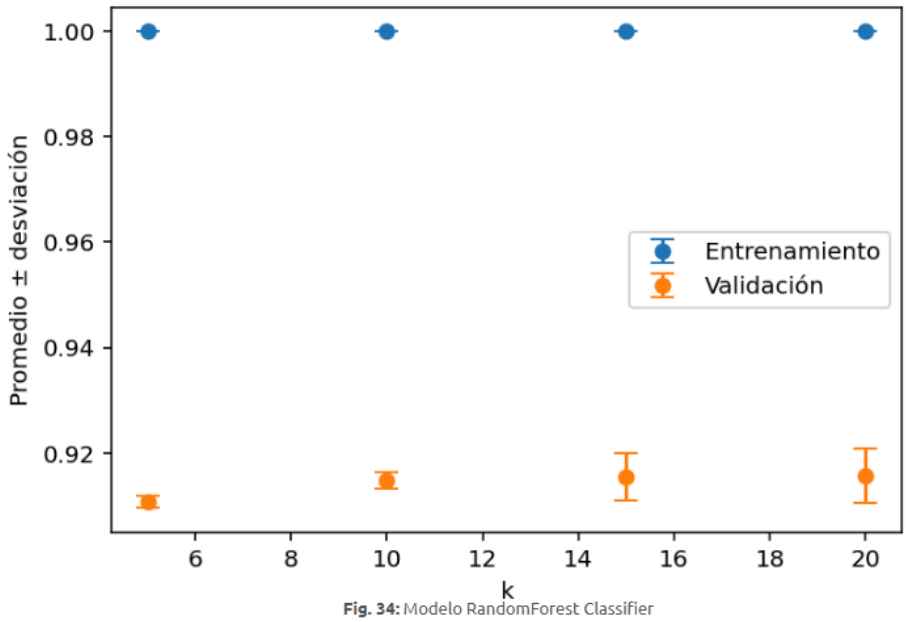

Fig. 34: Modelo RandomForest Classifier

En este caso particular el gráfico de desempeño del entrenamiento (en azul) no nos brinda información pues en todos los casos la media y la desviación tienen valores de 1.0 y 0.0 respectivamente.

Sin embargo el gráfico correspondiente a la validación (en naranja) sí contiene información relevante. A pesar de que no tiene muchas variaciones, se observa que a medida que "k" aumenta tenemos mayor variación en el desempeño, esta variación mide el error o la desviación estándar, son un poco más grandes.

Esto se debe a que a medida que "k" aumenta tendremos más particiones y por tanto cada partición contendrá menos datos. Esto incrementa la probabilidad de que haya mayor variabilidad entre las particiones y por tanto habrá mayor variabilidad en el desempeño, además que se requieren de mas recursos computacionales para realizar la corrida del análisis.

En este modelo en particular, dado sus métricas (media y desviación estandar de las accuracy) se puede concluir que dada una imagen se puede responder **¿A cuál de los 10 digitos corresponde la imagen?**.

Apéndice

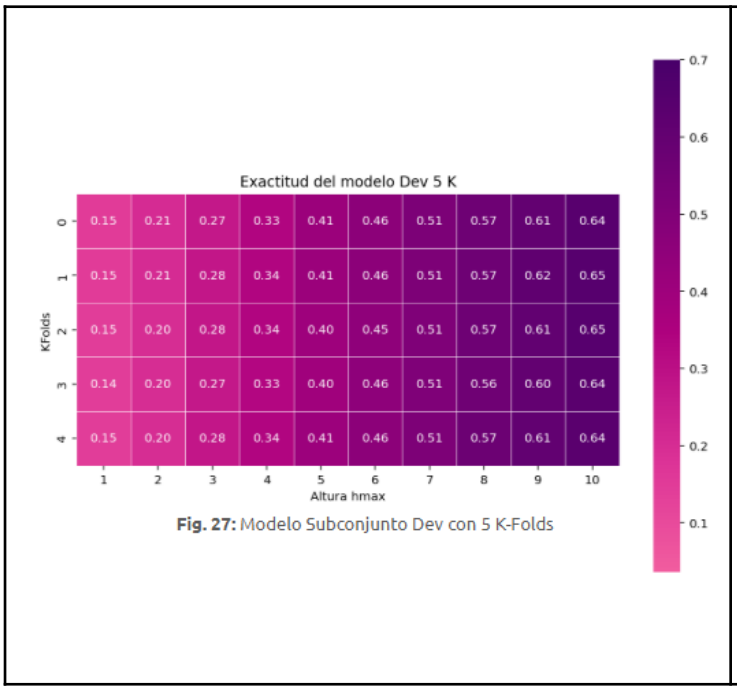

Fig. 27: Modelo Subconjunto Dev con 5 K-Folds

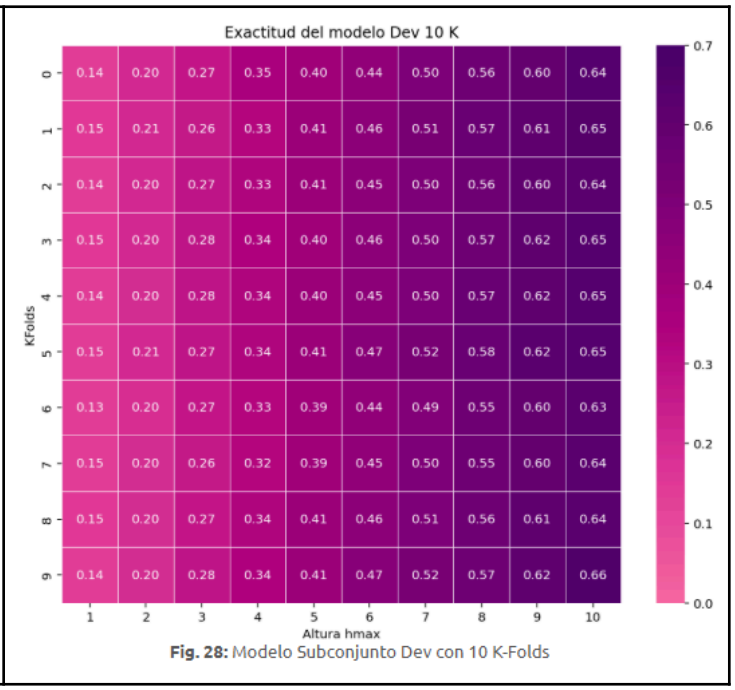

Fig. 28: Modelo Subconjunto Dev con 10 K-Folds

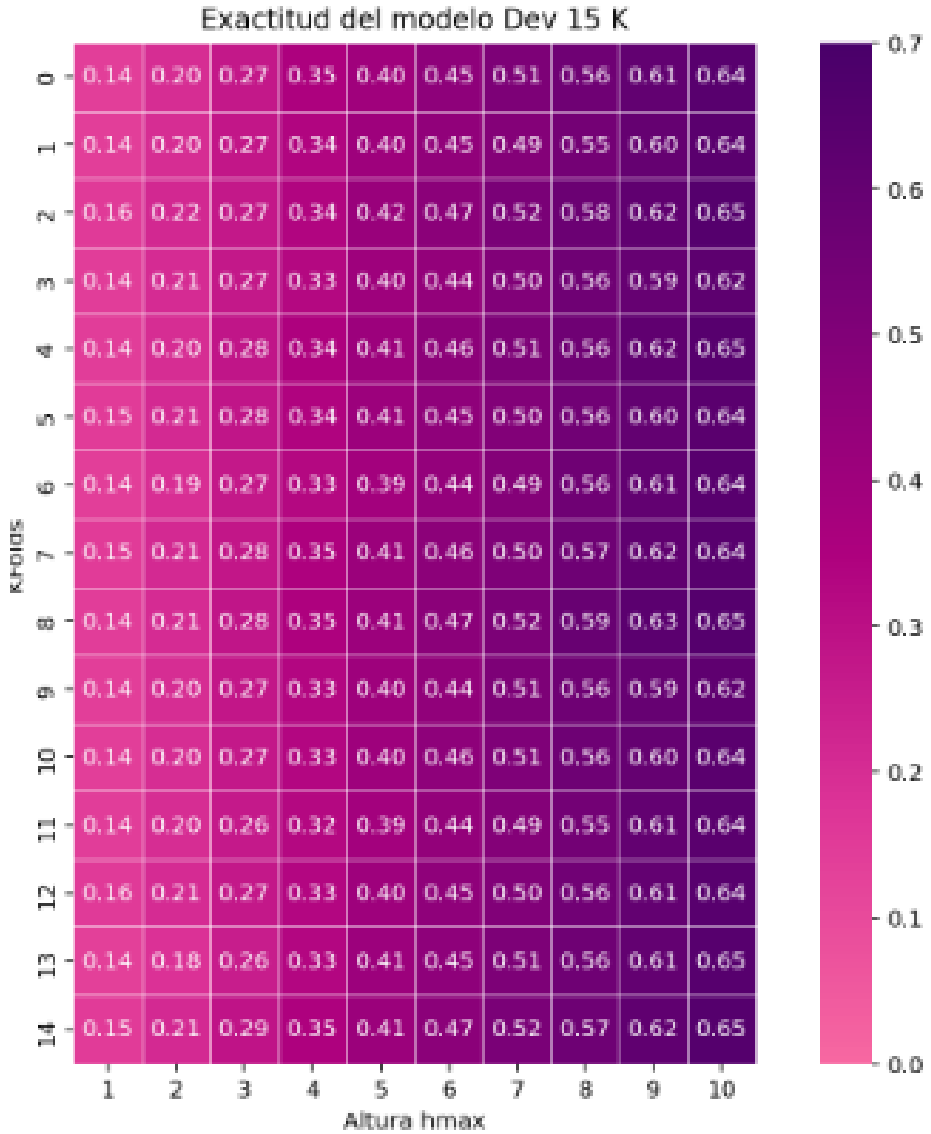

Fig. 29: Modelo Subconjunto Dev con 15 K-Folds

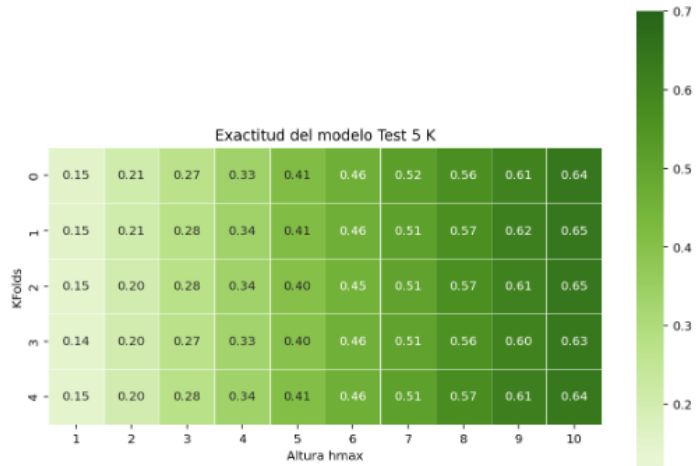

Fig. 30: Modelo Subconjunto Test con 5 K-Folds

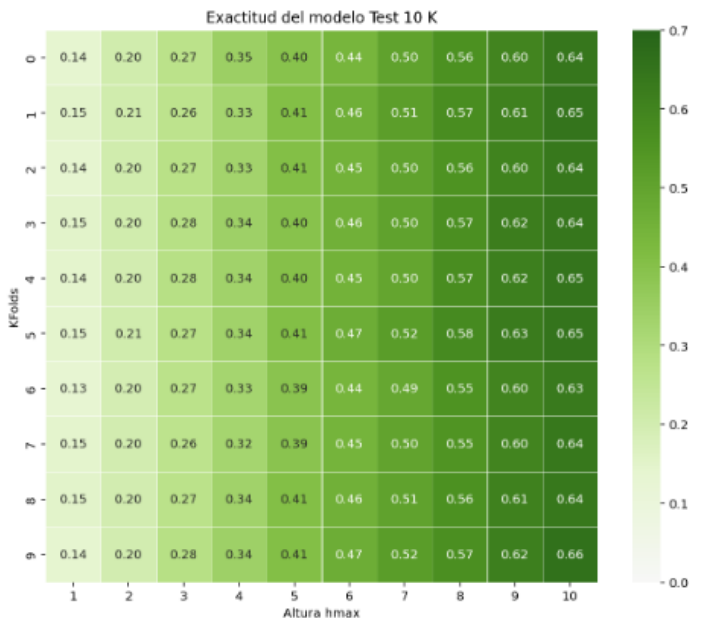

Fig. 31: Modelo Subconjunto Test con 10 K-Folds

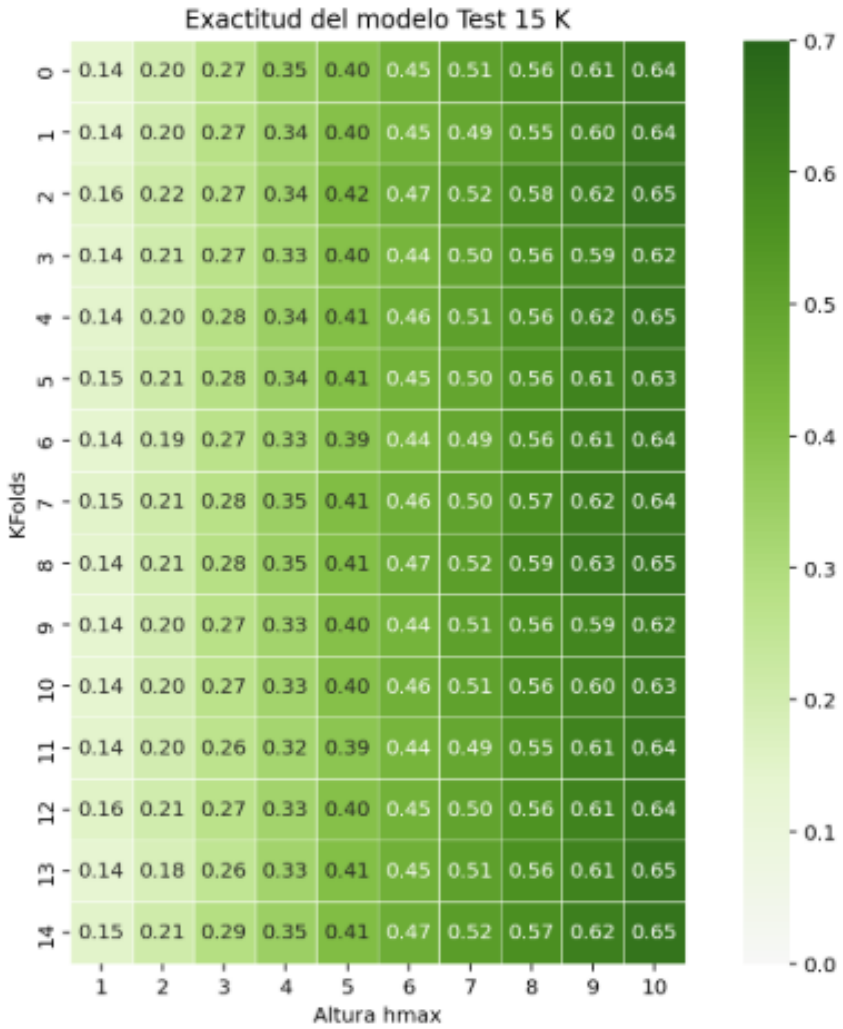

Fig. 32: Modelo Subconjunto Test con 10 K-Folds