DEPARTAMENTO DE COMPUTACION
Facultad de Ciencias Exactas y Naturales - UBA

Trabajo Práctico N°2

Clasificación y validación cruzada

12 de Noviembre 2023 **Laboratorio de Datos**

Grupo PeMaWa

Integrante	LU	Correo Electrónico
Dante Waisman	1163/22	waismandante@gmail.com
Alvaro Petriz Otaño	1080/22	alvarovpetriz@gmail.com
👻Matias Marzocca👻	621/22	matymarzocca@gmail.com

Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires
Ciudad Universitaria - (Pabellón I/Planta Baja)
Intendente Güiraldes 2610 - C1428EGA
Ciudad Autónoma de Buenos Aires - Rep. Argentina
Tel/Fax: (54 11) 4576-3300
https://exactas.uba.ar/

Introducción y Análisis Exploratorio de los Datos

En el siguiente trabajo, se utilizó el DataSet de Fashion MNIST con el objetivo de lograr confeccionar un modelo de clasificación que nos permita clasificar una imagen en base a 10 tipos de prendas de vestir. Para ello fue importante realizar un análisis de los datos antes de pensar el modelo a utilizar.

En primer lugar, nos pareció que hay atributos que no son igual de relevantes que otros. Dependiendo del objetivo que tengamos, será pertinente quedarse con ciertos atributos o no. Por ejemplo, si deseamos predecir si una prenda es una remera o un pantalón, en este caso, al ser prendas muy diferenciables, se pueden tomar los pixeles de las mangas y el espacio entre piernas para entrenar al modelo. Como se puede observar en las imágenes I y II, los pixeles de una manga de una remera de algún tono de gris mientras que el mismo pixel en un pantalón será negro.

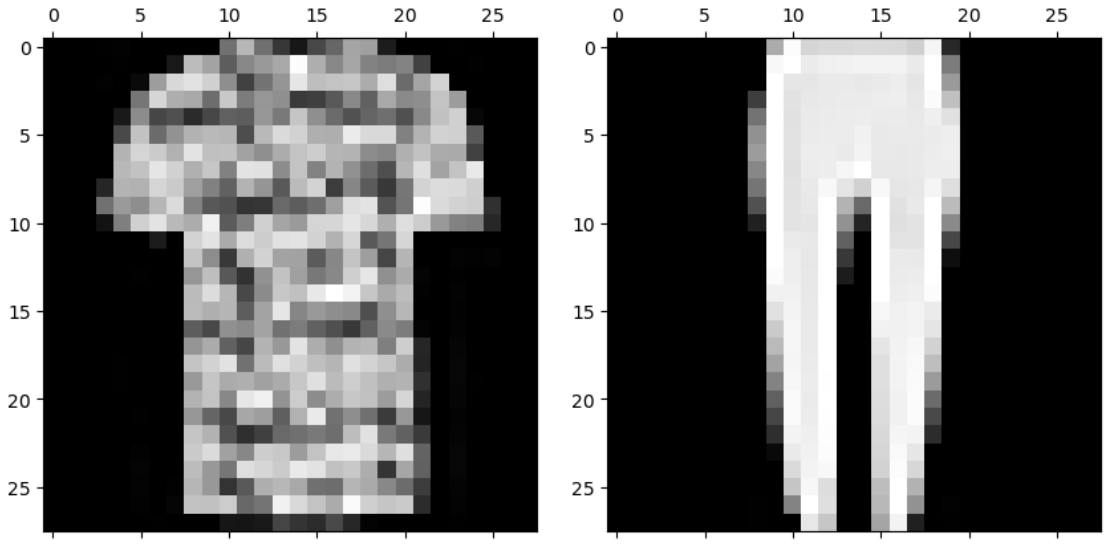


Imagen I (Izquierda): Gráfico de una remera perteneciente al Data Set.
Imagen II (Derecha): Idem Imagen I pero correspondiente a Pantalón.

Si quisiéramos cuantificar la relevancia/irrelevancia de algún atributo, deberíamos calcular el promedio en todo el DataSet del atributo y dependiendo que tan cerca se encuentre de 0, considerarlo más irrelevante. Por ejemplo el píxel 1, que dentro del Data Frame total (60.000

filas) tiene menos de 100 valores que no son 0, tendrá un promedio cercano a 0 y podríamos clasificarlo como irrelevante.

A diferencia del caso de la remera y el pantalón, se puede observar en el Data Frame prendas que se parecen entre sí. Por ejemplo, como se puede apreciar en la imagen III y IV, diferenciar entre un pullover y un abrigo es bastante difícil. Esto dificulta la posibilidad de seleccionar atributos relevantes para la diferenciación de estás prendas.

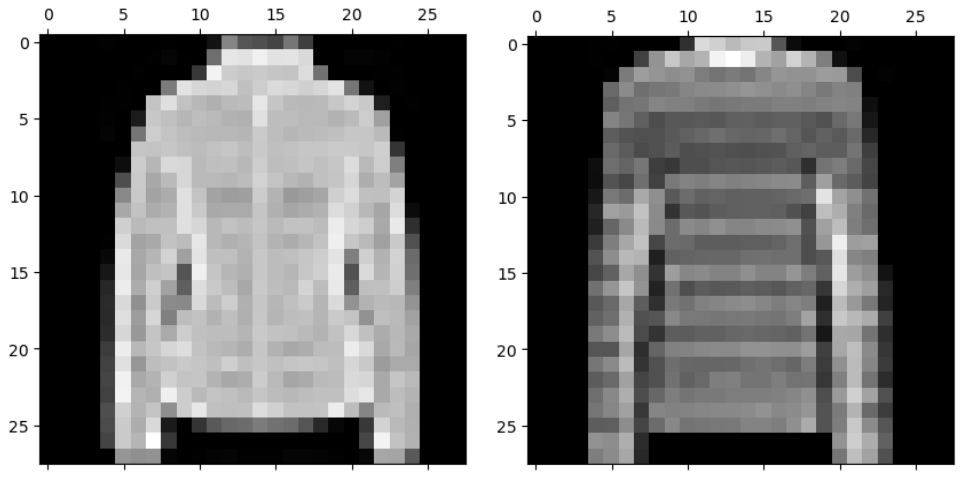


Imagen III (Izquierda): Gráfico de un abrigo correspondiente al Data Set.
Imagen IV (Derecha): Idem imagen III pero de un Pullover.

Es por eso que al tener en cuenta todas las prendas, por una cuestión de formato y pragmatismo consideramos preferible conservar todos los atributos dado que hay prendas más similares a otras y no amerita un análisis exhaustivo para determinar atributos más relevantes que otros.

Experimentos Realizados:

Gráficos por prenda respecto al promedio de cada una

En primer lugar y para analizar la variabilidad de las prendas, optamos por separar el data frame en cada una de las prendas y graficar el promedio de cada uno de sus atributos. De esta forma obtuvimos una imagen correspondiente al promedio de cada prenda. En base a los resultados, fue que observamos que ciertas prendas presentan una mayor variación respecto a otras. Para justificar esto tenemos un argumento visual. Mientras más desdibujado está el promedio de las prendas, menos similares son.

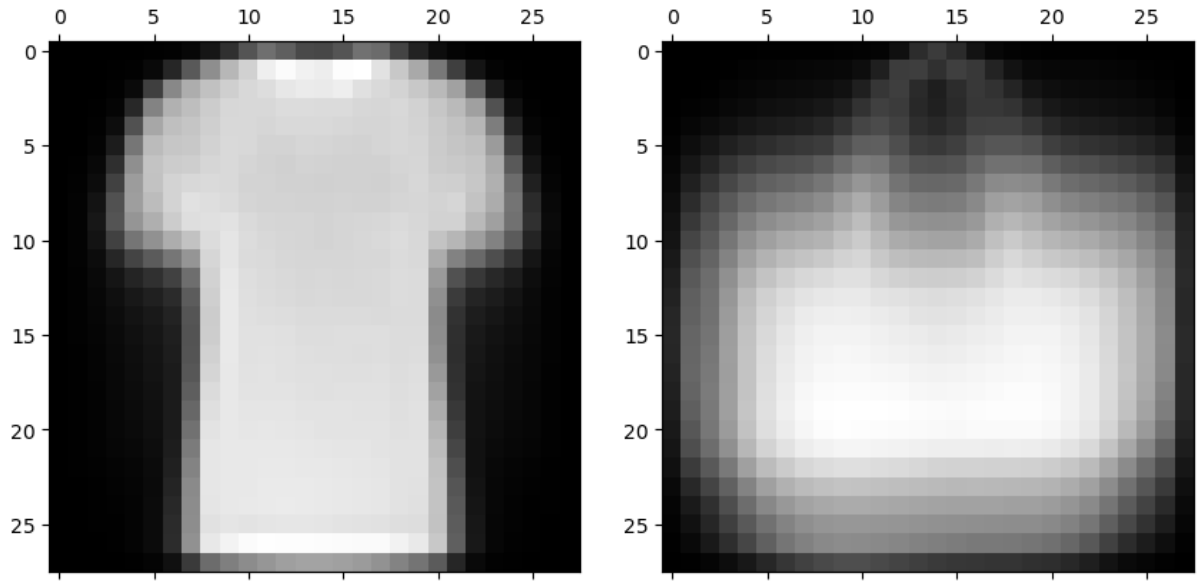


Imagen V: A la izquierda el gráfico del promedio para las remeras y a la derecha el de los bolsos.

Como se observa en la Imagen V, el resultado de graficar el promedio de las remeras nos muestra una imagen más nítida en relación al resultado del mismo experimento con los bolsos. En el primer caso es posible ver que se trata de una remera. Los bolsos por otro lado, son un rectángulo raro y para descifrar de qué se trata necesitamos apelar a la imaginación.

Modelo para predecir entre remeras y pantalones

Para diferenciar una remera de un pantalón usamos un modelo de K-neighbours. En primer lugar, acotamos el Data Frame original quedándonos únicamente con las filas correspondientes a remeras y pantalones. Luego separamos los datos en un conjunto de Train y otro de Test con una proporción de 70/30. A continuación, definimos tres conjuntos de tres variables independientes. El primer conjunto se eligió en base al análisis anterior, seleccionando atributos correspondientes a pixeles de las mangas de una remera y el espacio entre piernas de un pantalón. Luego se definió un conjunto tomando píxeles aleatorios y por último un conjunto con los tres primeros píxeles.

Nuestra hipótesis consistió en que, tomando 3 pixeles al azar, el modelo no tendría una exactitud alta, dado que estamos trabajando con muy pocos atributos para un Data Frame muy grande. Contrariamente, si utilizamos pocos atributos elegidos con un buen criterio, sería posible obtener una exactitud alta para nuestro modelo y con una gran eficiencia en tiempos y consumo de recursos computacionales. Por último, y como caso testigo, entrenamos al modelo con tres atributos deliberadamente malos, con el objetivo de obtener la exactitud más baja posible. Los resultados fueron los siguientes:

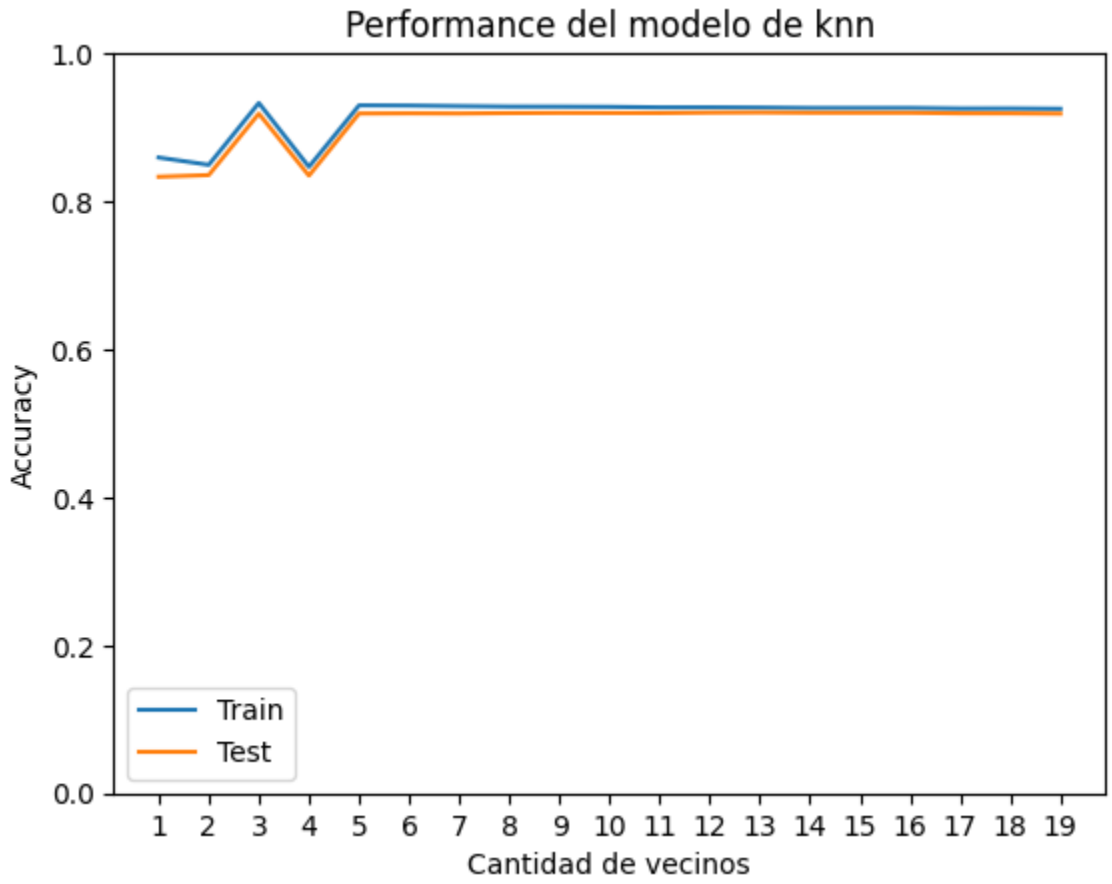
Variables independientes	**Accuracy Train**	**Accuracy Test**
Píxeles con criterio	93%	92%
Píxeles al azar	75%	73%
Píxeles malos	50%	51%

Tabla I: Score de modelos de K-neighbours para clasificador de remeras y pantalones.

Nos dimos cuenta que utilizando un modelo de k-vecinos más cercanos que toma tan solo 3 píxeles bien elegidos y un k=3 el modelo tenía un score muy alto al evaluarlo con el conjunto de Test, lo que valida nuestra hipótesis. Por otro lado, observamos que el score del modelo entrenado con píxeles aleatorios es de 73%, siendo considerablemente menor al primer modelo clasificador. Luego, el modelo entrenado con píxeles malos tuvo una exactitud del 51% al evaluarse con el conjunto de Test, lo que lo posiciona como el peor modelo tal y como esperábamos. Notamos que, si un modelo tiene que clasificar entre dos clases, una exactitud del

50% es el peor score que puede obtener ya que sería equivalente a elegir al azar entre dos opciones. De esta manera confirmamos todos los puntos planteados en la hipótesis.

Aun así, quisimos encontrar con que cantidad de k-neighbours nuestro modelo obtenía una mejor exactitud con los 3 píxeles que habíamos elegido. Para ello, iteramos 5 veces el experimento, separando en conjuntos de train y test por cada iteración y evaluando la exactitud para distintos modelos entrenados con un k de 1 hasta 20. Luego graficamos el promedio del score correspondiente al conjunto de train y test de las 5 iteraciones para cada k-neighbour de 1 a 20. De esta forma obtuvimos un gráfico de la performance del modelo.



Acá podemos ver cómo a partir de k=3 el score no mejora. Por ende, elegimos k=3 ya que tiene menor costo computacional en comparación a un k más alto.

Modelo para predecir entre las 10 prendas

Para predecir entre las 10 prendas confeccionamos un modelo de árbol de decisión. Para ello primero separamos los datos en un conjunto de Train y otro de Test en una proporción 70/30.

Para saber cuál era el mejor árbol posible, utilizamos Grid Search CV. Evaluamos alturas de 2 a 16 y los criterios de Gini y Entropía.

El Grid Search que utilizamos hace una búsqueda exhaustiva a través de K-folds y Cross-validation con los hiper parámetros definidos. Con esto, reducimos el riesgo de separar "incorrectamente" los datos en train y test, y aumentamos las probabilidades de elegir un modelo robusto. En nuestro caso, decidimos usar 5 folds.

El output de este método, arroja como mejor árbol uno de altura 12 y de criterio entropía con un score de 80,7%.

Finalmente, para evaluar la performance del modelo obtenido, lo evaluamos con el conjunto de Test que separamos al principio. Esto nos da una exactitud de 81%.

Conclusión

En base al análisis del Data Frame y los distintos experimentos realizados, podemos concluir que un data frame compuesto por imágenes dificulta la exploración de datos en una primera instancia. Esto se debe a que los datos son mucho menos “tangibles”. Los datasets comunes tienen una interpretación más directa. Por ejemplo, en la primera fila del dataset ejemplo correspondiente a la Tabla II, observamos que hay un Stock de 25 autos modelos Toyota Camry y no requirió ninguna complejidad llegar a esa interpretación más que mirar los datos. En cambio, para interpretar los datos del data frame de Fashion-MNIST, es necesario leer la documentación y aún así no es suficiente para comprenderla. Para ello debemos recurrir a herramientas de visualización. Antes de sumergirnos en el trabajo y de leer la documentación, el dato de que el pixel 417 es 180 no nos dice absolutamente nada.

Vehículo	Marca	Modelo	Año	Stock
Auto	Toyota	Camry	2022	25
Moto	Honda	CBR600RR	2021	15
Auto	Ford	Mustang	2022	30

Tabla II: Data Frame vehículos fácilmente interpretable.

También podemos concluir que fue posible adaptar distintos tipos de modelos de clasificación para determinados escenarios, cumpliendo así los objetivos propuestos.

Para diferenciar entre una remera y un pantalón, usamos un modelo K-neighbours con tres valores para k y 3 píxeles que elegimos estratégicamente como variables independientes obteniendo gran precisión.

Luego, para el problema de predecir de qué prenda estábamos hablando, utilizamos Grid Search para obtener el mejor árbol posible obteniendo uno de 12 niveles de profundidad y con criterio de entropía. Con dicho modelo obtuvimos una precisión de 81% de exactitud para nuestro conjunto de Test. Siendo que el data frame poseía 60.000 filas y 784 columnas y que buscábamos clasificar entre 10 tipos de prendas, consideramos que el resultado fue satisfactorio.

En ambos casos, pudimos lograr un rendimiento aceptable con modelos relativamente simples.