Trabajo Práctico 2

Laboratorio de Datos
Grupo Pierre Menard

Gantes, Augusto
Belmes, Martin
D'Andrea, Matias

Índice

1. Análisis exploratorio

En primera instancia, nos encontramos con un *dataset* que codifica imágenes cuadradas con lados de 28 píxeles en escala de grises por píxel, es decir, hay una columna por posición de píxel con algún valor entre 0 y 255. A estas posiciones se le suma otra columna que contiene la información referida a qué tipo de prenda corresponde cada imagen. Hay diez posibles clasificaciones, cada una con un número y una prenda correspondiente: *0-T-shirt/top, 1-Trouser, 2-Pullover, 3-Dress, 4-Coat, 5-Sandal, 6-Shirt, 7-Sneaker, 8-Bag, 9-Ankle boot.* El gran problema que se presenta es que esta data suponemos que tiene un formato que favorece su almacenamiento pero dificulta de manera significativa su análisis, no nos parece la mejor idea para visualizar y analizar imágenes.

Lo que nos parece interesante al ver medidas resumen de los datos es que la posible información que posee cada píxel varía de manera drástica. Esto nos puede ayudar a especular en qué información es útil para estimar la clasificación de prenda. Desde ya, es indispensable la etiqueta de la prenda pero especulamos que hay píxeles que generalmente no aportan para ningún caso, i.e., creemos que hay píxeles que se mantienen invariantes o no varían demasiado para todas las imágenes. Nos parece que los bordes podrían ser candidatos bastante claros para los cuales esto sucede en varios casos. A partir de estas intuiciones, generamos un gráfico que vincule la desviación estándar de cada píxel, sin discriminar por prenda, con la posición de dicho píxel junto a otro gráfico que permita ver el promedio de la escala de grises para todos. Se recurrió a una paleta diferente a escala de grises para lograr mayor contraste entre valores relativamente menores al resto.

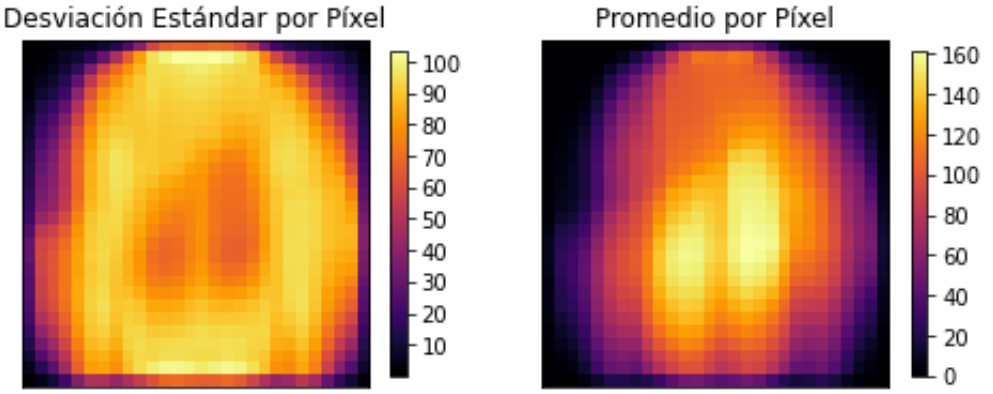

Figura 1: Desviación estándar por píxel junto al promedio de cada píxel para todas las clases.

Se utilizó un gráfico que muestra que tanto varían los píxeles para el conjunto total de las clases (lado izquierdo) y también se le adjunto el gráfico que representa el promedio de los píxeles para el conjunto total (lado derecho). Por lo que se puede apreciar, de alguna manera estamos en lo correcto, ciertas partes del borde varían relativamente menos que el resto de la imagen. Se denota sobre todo en la imagen del promedio que hay muchos valores por debajo de la cota de 40. Todo esto nos indica que sería buena idea utilizar los píxeles que más varían.

Podemos también analizar como es la comparación entre clases. Tomamos el promedio de cada clase y generamos un gráfico que ilustra que tan diferentes son las "plantillas" de cada clase. Lo que se observa en el gráfico es que básicamente hay grupos de clases que va a ser fácil de diferenciar del resto pero, posiblemente, muy difíciles de diferenciar entre los miembros del grupo. En principio, se distingue entre el grupo que llamaremos *calzado*, que contiene las etiquetas 5, 7 y 9; luego, se puede ver que hay un grupo muy marcado que son las *prendas superiores* que contiene a las etiquetas 2, 4 y 6, excluimos la etiqueta 0 ya que se puede ver que el promedio indica que es distinguible por las mangas. El resto de clases que no están contenidas en los grupos mencionados consideramos que son significativamente diferentes del resto.

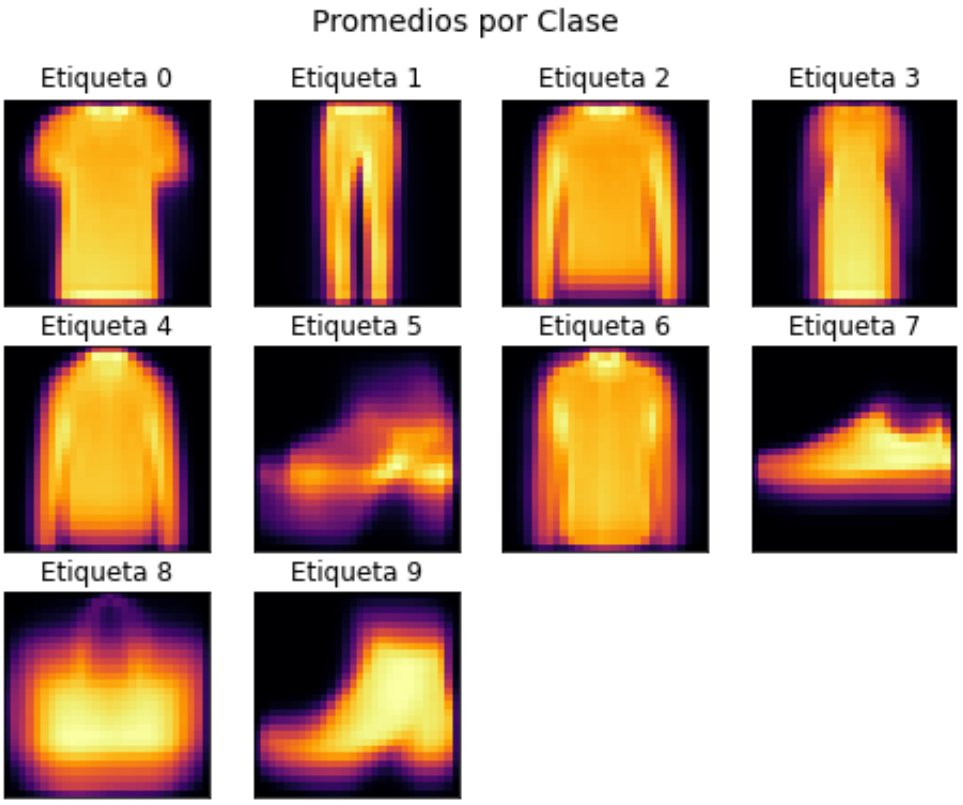

Figura 2: Promedio de cada píxel por etiqueta en paralelo.

En cuanto a que tan parecidas son las instancias de cada clase entre ellas depende de la clase. Por ejemplo, se nota en la Fig. 2 hay una clara visualización de que para la clase 5 no hay un modelo estándar muy marcado para la clase, mientras que si vemos el promedio de la clase 1 parece ser bastante claro que todas las instancias siguen una forma repetitiva. Otro ejemplo interesante es el de la clase 3 en la que se ve cierto grado de irregularidad en la forma que varía la clase. Podemos ver en la Fig. 3 el desvío estándar de la clase junto al promedio para evidenciar lo que acabamos de comentar.

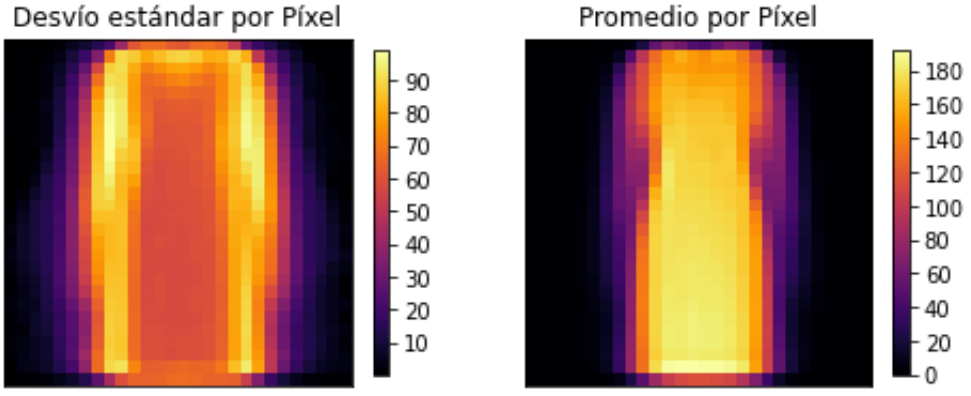

Figura 3: Promedio y desvío estándar de la etiqueta 3 en paralelo.

2. Clasificación binaría

Se nos pide crear un modelo que diferencie entre un pantalón y una remera utilizando solo tres píxeles de la imagen. Siguiendo los pasos vistos en clase, primero modificamos el *dataframe* para quedarnos sólo con los pantalones y las remeras. Ahora, teniendo en cuenta las imágenes promedio de las clases 0 y 1, nos centramos en seleccionar una región de interés, manualmente, de tal manera que se contengan los píxeles que creemos que pueden generar la mayor capacidad de predicción en el modelo. La región de interés seleccionada vendría a ser la parte de la imagen en la cual el pantalón suele tener poca variación mientras que en la remera todo lo contrario. Esto último queda evidenciado en la Fig. 4, que muestra como es la desviación estándar de la región de interés para ambas clases. La región de interés tiene una dimensión de 3 píxeles de ancho y 14 píxeles de alto. Luego, iteramos por cada grupo de tres píxeles que se distribuyen de manera horizontal y los separamos para entrenar al modelo de KNN que tenemos en cuenta para este caso.

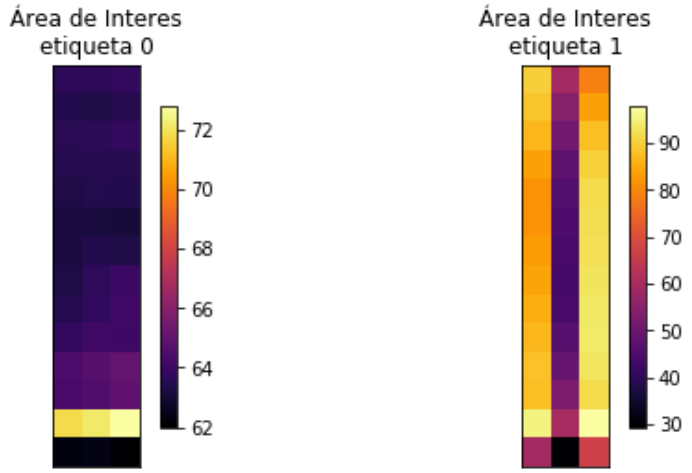

Figura 4: Desviación Estándar de cada una de las etiquetas en la región de interés.

En cuanto al modelo y su entrenamiento, separamos el 20 % de la data para usar como *holdout*. Con el otro 80 % creamos nuestro modelo KNN y lo alimentamos con nada más tres píxeles. Buscamos qué parámetros nos conviene usar con la función *GridSearchCV* de *sklearn*. En la utilización de la función *GridSearchCV* se varío el k del modelo entre tres y veinte, incluidos. La búsqueda de parámetros se realizo para cada combinación de tres píxeles que estuvieran contiguos horizontalmente. Se escogió al modelo cuyo puntaje de entrenamiento fuera el máximo, obtuvimos que $k = 6$, para los píxeles $(630, 631, 632)$. Este resultado fue obtenido en la última ejecución del código, esto ha de ser aclarado por la falta de establecimiento de semillas dentro de todo el proceso. El modelo resulto tener un puntaje de entrenamiento de 97,07 % y un puntaje de validación con el *holdout* de 98,41 %. Cabe aclarar que el puntaje de validación es aquel proveniente de tomar el promedió a el puntaje devuelto por *cross_val_score* con cinco *folds*.

3. Clasificación multiclase

Se nos pide seguir una serie de pasos que ayuden a determinar, dada una imagen, a qué clase corresponde, es decir, qué tipo de prenda es, y todo esto con un modelo de clasificación de árbol de decisión. El carácter que tomamos para ver con que datos alimentar a nuestro modelo fue elegir aquellos píxeles que contengan la desviación estándar por arriba de cierta cota que se establecería manualmente ya que creemos que los mejores candidatos son aquellos píxeles que más varían. Luego, se entrenaría al árbol de decisión con los datos que superen dicha cota. Se decidió que la cota a utilizar sería de noventa, i.e., los píxeles que tengan desvío estándar por encima de esta cota serían los utilizados para entrenar el modelo. Esta cota deja al modelo con 215 píxeles a utilizar.

Para el entrenamiento y validación, separamos el 20 % de la data para ser usada en validación y con el otro 80 % creamos nuestro modelo de árbol de decisión. Buscamos qué parámetros nos convenía usar con la función *GridSearchCV* de *sklearn*. Los hiper parametros establecidos fueron los criterios *gini* y *entropy* junto con la exploración de profundidades desde 10 a 15. Una vez entrenado obtuvimos que el criterio que maximizaría el puntaje es *entropy* y la profundidad del árbol debía ser de once, de nuevo, estos hiperparametros se obtuvieron en la última ejecución del código. Luego, con *cross_val_score* evaluamos qué tan preciso fue nuestro árbol en predecir el tipo de prenda de una imagen y devolvió un *mean* de *0.7645*, con cinco *folds*. Para analizar en dónde estaba fallando el árbol, recurrimos a una matriz de confusión.

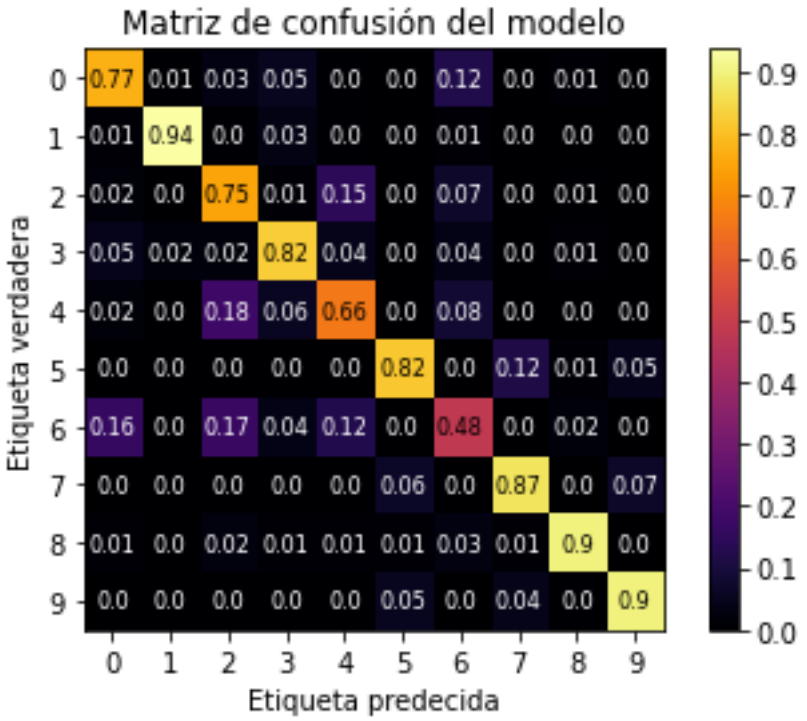

Figura 5: Matriz de confusión.

Gracias a esta herramienta, pudimos observar que el sistema tiene una muy alta eficacia para reconocer a la etiqueta 1, 8, y 9. También alta para la 3, 5 y 7. Pero que tuvo problemas para identificar la 0, 2, 4, y especialmente, la 6, que era lo que habíamos llamado *prendas superiores*. Se ve en la matriz que el sistema encuentra similares a las clases de este grupo, ya que se las confunde casi exclusivamente entre sí. Casi la mitad de las veces que se le mostró una clase 6, predijo clase 0, 2, o 4. Esto tiene sentido, porque las imágenes muchas veces son similares, y los atributos que las distinguen parece cambiar entre imágenes.

4. Conclusiones

Luego de tomar los datos, analizarlos, y trabajar sobre ellos debidamente, llegamos a las siguientes conclusiones. Primero, los datos, que estaban en formato que pareciera ser ideal para el trabajo que hicimos, por el volumen, la distribución de clases y la regularidad de las dimensiones de las imágenes, contaban con ciertas desventajas. Luego de calcular la desviación estándar, y el valor promedio de cada píxel por cada clase, pudimos descartar los píxeles que menos variaban, y que por lo tanto, menos información nos aportaban, y al analizar los promedios, encontramos un posible problema, que luego confirmaríamos al ver la imágenes individualmente. Específicamente en las clases 0, 2, 4, y 6, a las que llamamos *prendas superiores* (inicialmente no incluimos al 0 dentro de este grupo porque asumimos que sería fácilmente diferenciable del resto), sus imágenes eran demasiado parecidas a imágenes de otras clases de este grupo, y no lo suficientemente parecidas a las otras imágenes de la misma clase.

Esto tomó un poco más de fuerza cuando creamos un modelo que diferencie un pantalón de una remera. Con nuestro análisis preliminar, dedujimos que esto sería fácil, porque los pantalones cuentan con una franja de un valor distinto en el medio, y la remera no. Tomamos 3 píxeles y creamos un modelo KNN altamente efectivo, con un valor máximo de 0,97 % en el entrenamiento , y un promedio de 0.94. Lo que tomamos de esto es que el modelo KNN, o incluso puede que otro modelo, es altamente efectivo para diferenciar *prendas superiores* de otras prendas, y también intuimos que podría diferenciar otras prendas entre sí, pero sospechábamos que tendría problemas para diferenciar *prendas superiores* entre si.

Finalmente hicimos un modelo de árbol de decisión que confirmó lo que esperábamos, al tener mayor problema para predecir correctamente una imagen de la clase 2, 4, y 6, y para nuestra sorpresa también de la clase 0, e hicimos una matriz de confusión donde confirmamos que la dificultad de hallaba en diferenciar *prendas superiores* entre si. Igualmente, a excepción de la clase 6, y quizás la 4 también, los resultados no son malos, y serían una herramienta útil para, dada una imagen, determinar qué tipo de prenda es. También notamos que se podría optimizar el árbol, para lograr un valor más alto, pero viendo las imágenes, es lógico el resultado que devuelve, y no queremos indebidamente ocasionar *over-fitting* en nuestro modelo, es decir que nuestro sistema, que aunque efectivo, no está completamente optimizado para trabajar con las imágenes dadas, si sería el mejor para identificar la clase de una imagen nueva de un tipo de ropa en el mismo formato.