Pontificia Universidad Javeriana.

Inteligencia Artificial.

José Daniel Arango López

Santiago Franco Alfonso

Miriam Zareth Osorio Mendoza

Proyecto Final: Clasificador de frutas

1. Introducción:

En el presente documento se presenta el desarrollo de un clasificador de frutas con el cual se buscará distinguir 3 tipos de frutas diferentes, las cuales son manzanas, bananos y kiwis. Para el proceso se hará uso de los clasificadores vistos, entre los cuales se tiene: un clasificador por regresión logística, KNN, SVM y una red neuronal artificial. Cada uno de los clasificadores posee diferentes grados de dificultad y complejidad que permiten obtener una mejor o peor clasificación de los datos.

Otro reto que supone realizar este proyecto es realizar un procesamiento de las imágenes que nos permita obtener un conjunto de datos apropiado o manejable para poder aplicar los clasificadores. En este proyecto se utilizó un *dataset* de la página de *Kaggle* de donde se obtuvo las imágenes a las cuales, posteriormente se les aplico un histograma, mediante el cual se pudo convertir los datos de pixeles a valores numéricos manejables; y por último se les asigno manualmente las etiquetas y así tener un problema de clasificación supervisado.

2. Desarrollo:

Para el desarrollo del proyecto se tomó un *dataset* identificado como "*Fruit recognition*"[1] en la página de *Kaggle*, el cual está conformado en su totalidad de imágenes, cuenta con un total de 44406 fotos divididas en 15 tipos de frutas. Teniendo en cuenta la información anterior y los objetivos planteados inicialmente, se tomó la decisión de seleccionar 3 tipos de frutas (manzana, banano y Kiwi) y se redujo el número de imágenes de cada una de estas carpetas a 740, constituyendo de esta forma los datos que más adelante tomaran la forma del *dataset* que será usado en la implementación.

- Obtención de datos

Los datos tenían que ser procesados para poder introducirlos a los diferentes clasificadores, para hacer esto se realizó un proceso que se puede dividir en varias etapas.

En primer lugar, fue necesario hacer un filtrado del *dataset*, es decir, seleccionar las frutas y las imágenes de cada fruta que serían procesadas para obtener el archivo final con todos los datos que se utilizarían. Esto fue necesario porque el *dataset* original contenía una gran cantidad de frutas y una gran cantidad de imágenes por cada fruta, por eso fue necesario escoger específicamente que frutas e imágenes se utilizarían para demostrar con más facilidad el funcionamiento del proyecto.

1. https://www.kaggle.com/chrisfilo/fruit-recognition

A continuación, fue necesario investigar cómo obtener datos útiles a partir de una imagen, ya que nuestro *dataset* está compuesto en su totalidad de imágenes. Este proceso nos ayudaría a identificar cada fruta y así diferenciar una de otra, a pesar de tener elementos similares cada una de las imágenes independientemente de la fruta, por ejemplo, la bandeja metálica y el fondo de cada fotografía.

Al realizar esta investigación se halló que existen varias herramientas para hacer análisis y procesamiento de imágenes, pero una de las herramientas más apropiadas dadas las circunstancias era la función de histograma 2D de opencv. Esta función usa una imagen para extraer de ella datos en forma matricial, este análisis lo hace a partir de los colores que se aprecian en la imagen, lo cual resulta ser muy útil en este contexto.

Después de comprobar el funcionamiento de los histogramas, se hizo una función que leyera automáticamente todas las imágenes de una carpeta y al mismo tiempo fuera extrayendo los histogramas. Todos estos datos obtenidos se iban guardando en un archivo de texto para luego unirlo a los otros archivos, correspondientes a las otras dos frutas. Para simplificar un poco los datos obtenidos se hizo uso de otra función que convierte una matriz en un vector, de esta manera se convertía la matriz de histograma en un vector único para cada una de las imágenes, y de esta manera se logró simplificar el *dataset* que se iba obteniendo.

Finalmente, se juntaron todos los datos obtenidos después de realizar el proceso anterior con cada una de las carpetas con imágenes de las frutas. Para consolidar todos los datos en un mismo *dataset* se hizo uso de Excel, allí se separaron los valores en celdas independientes y se colocaron los conjuntos de datos uno debajo del otro, adicionalmente, se agregaron las etiquetas para cada vector, las cuales indican que tipo de fruta corresponde a esos datos, utilizando valores numéricos, en donde el 1 corresponde a los kiwis, 2 para las manzanas y 3 para los bananos. Al finalizar todo este proceso se generó el archivo final, un archivo "*csv*" que sería el *dataset* utilizado en el resto del código.

Al momento de implementar los clasificadores, se utilizaron los cuatro siguientes, los cuales van de menor a mayor nivel de dificultad.

- Regresión logística

Para la implementación del clasificador por regresión logística en primer lugar se importó el modelo con ciertas características como la penalidad, la cual es un factor de regularización; el valor de C que va a ser inversamente proporcional al peso de la regularización; y por último el valor de random_state que determina la generación de números pseudo aleatorios para la "mezcla" o "shuffle" de los datos para estimados de probabilidad. Posteriormente se ingresan los datos de entrenamiento y luego se evalúa el clasificador con los datos de prueba, para finalmente obtener las métricas de desempeño, que en este caso es el coeficiente de correlación de Matthews. Adicionalmente, se muestra una matriz de valores predichos por el clasificador del conjunto de prueba, y se muestra también los valores reales para poder contrastar ambos conjuntos de datos.

- KNN

Para implementar este clasificador solo se utilizan las características x, luego se selecciona como 15 el número de vecinos contra los cuales se aplicará el algoritmo por cada muestra evaluada, la distancia entre la muestra y los vecinos será tomada euclidiana. Luego para poder

1. https://www.kaggle.com/chrisfilo/fruit-recognition

realizar una evaluación del desempeño del clasificador se calculan e imprimen en pantalla las métricas del accuracy sobre el conjunto de entrenamiento y sobre el conjunto de validación. Adicionalmente se obtuvo la matriz de confusión sobre el conjunto de prueba y la predicción y las métricas que se pueden calcular a partir de la matriz de confusión.

Luego de obtener los datos anteriores, se procede a graficar diferentes valores de K hallados por el clasificador, estos valores de k se grafican con respecto a su accuracy para de manera visual poder seleccionar un valor de k optimo, y comparar si el algoritmo pudo llegar a predecir que eran 3 clases.

- SVM

Para implementar el clasificador SVM se probaron varios tipos de kernel, los cuales son lineal, polinomial cuadrático, polinomial cubico, rbf y sigmoide, tras las pruebas del clasificador haciendo uso de todos los kernel anteriores se calcularon las métricas a partir de la matriz de confusión. Y con esos resultados de las métricas se pudo decidir que el kernel polinomial cuadrático presenta mejores resultados que el resto.

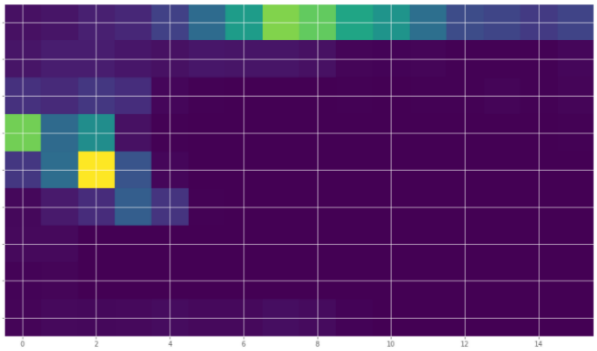
- Red neuronal

Finalizando, se decidió implementar una red neuronal de la librería de Sklearn, a la cual se le establecieron los parámetros de 3 capas ocultas donde las primeras dos están conformadas por 8 neuronas y la tercera de dos neuronas, cada una con una función de activación “relu” y un máximo número de iteraciones de 500, para que el algoritmo no sea tan lento, pero alcance a tener buenos resultados.

Luego se calculan e imprimen las métricas de probabilidad de predicción, la predicción realizada sobre el conjunto de validación y por último la métrica de Score.

3. Resultados:

Como primer resultado se tiene la imagen luego de su procesamiento aplicado para traducirla de una imagen a color a un histograma de dos dimensiones que permite ubicar y obtener un valor específico para cada tipo de fruta.



A continuación, se muestra la matriz que construye la imagen anterior; así mismo es el conjunto de datos que vamos a utilizar por cada imagen.

1. https://www.kaggle.com/chrisfilo/fruit-recognition

```
[6.4000e+02 8.0700e+02 1.2160e+03 1.5740e+03 2.7320e+03 4.9000e+03
 7.7720e+03 1.1412e+04 1.0697e+04 8.2810e+03 7.2800e+03 5.1510e+03
 3.2810e+03 2.9190e+03 2.3740e+03 2.8660e+03 7.6900e+02 1.1430e+03
 1.1350e+03 8.3500e+02 6.2100e+02 8.2500e+02 8.0000e+02 8.0400e+02
 6.1600e+02 2.0300e+02 1.1000e+02 1.8500e+02 9.2000e+01 9.5000e+01
 7.5000e+01 2.3400e+02 1.9830e+03 1.6590e+03 2.2910e+03 1.8230e+03
 2.5500e+02 6.7000e+01 3.5000e+01 2.1000e+01 1.8000e+01 5.7000e+01
 4.6000e+01 8.5000e+01 1.0500e+02 1.8000e+02 5.9000e+01 2.0700e+02
 1.1040e+04 4.8370e+03 6.9000e+03 6.5500e+02 8.2000e+01 5.4000e+01
 3.9000e+01 2.0000e+01 9.0000e+00 1.1000e+01 1.1000e+01 2.5000e+01
 2.5000e+01 2.9000e+01 9.0000e+00 7.6000e+01 2.2840e+03 5.0380e+03
 1.4055e+04 3.6400e+03 2.5600e+02 3.9000e+01 2.0000e+01 8.0000e+00
 7.0000e+00 3.0000e+00 4.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00
 0.0000e+00 0.0000e+00 2.9700e+02 1.0380e+03 1.7610e+03 4.1990e+03
 2.0930e+03 1.0300e+02 3.7000e+01 2.3000e+01 1.1000e+01 5.0000e+00
 8.0000e+00 3.0000e+00 1.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00
 3.6100e+02 3.3100e+02 4.0000e+01 5.0000e+00 0.0000e+00 0.0000e+00
 1.0000e+00 0.0000e+00 1.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00
 0.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00 1.4200e+02 1.7900e+02
 2.3000e+01 3.0000e+00 1.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00
 0.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00
 0.0000e+00 0.0000e+00 2.0900e+02 3.3800e+02 3.2800e+02 3.3500e+02
 4.4800e+02 3.5800e+02 3.3700e+02 4.9500e+02 4.0000e+02 1.1900e+02
 5.0000e+00 8.0000e+00 1.0000e+00 0.0000e+00 0.0000e+00 2.0000e+00]
```

Por cada imagen del conjunto de datos se creó una matriz como la anterior se aplanaron los daos para volverlos un solo vector con todos los datos y se exportaron a un archivo .txt; luego se crea un archivo .txt por cada conjunto de frutas por lo cual se crean 3 archivos, uno por cada fruta. Luego se unifican los archivos en uno solo .csv, al cual se le introducen manualmente las etiquetas. Y finalmente se importa ese archivo al programa y se le asigna una redistribución de filas aleatoriamente y el resultado se muestra a continuación.

	0	1	2	3	4	5	6	7	8	9	...
1367	2	791.0	1610.0	2870.0	4130.0	5310.0	7610.0	9440.0	9560.0	12300.0	...
747	2	2240.0	2670.0	2610.0	3340.0	3810.0	4720.0	6080.0	8820.0	9570.0	...
309	1	219.0	37.0	15.0	16.0	12.0	17.0	2.0	5.0	7.0	...
1485	3	173.0	172.0	246.0	527.0	1150.0	400.0	338.0	117.0	16.0	...
2165	3	85.0	114.0	163.0	132.0	42.0	26.0	4.0	4.0	4.0	...

5 rows × 145 columns

Luego de esto se procede a utilizar el primer clasificador que es KNN y su desempeño se evalúa de acuerdo con las siguientes métricas.

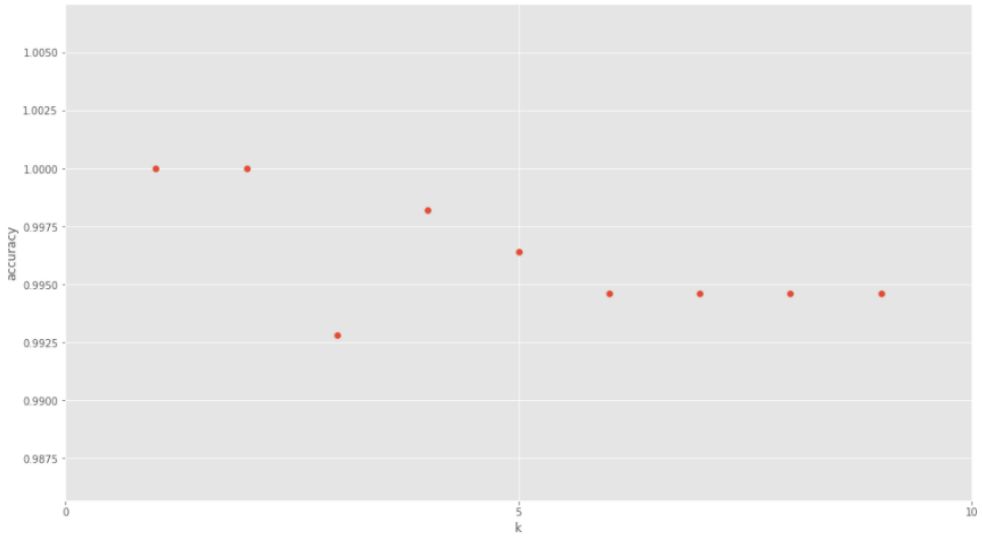
```
Accuracy of K-NN classifier on training set: 0.98
Accuracy of K-NN classifier on test set: 0.99
[[197   0   0]
 [  0 171   0]
 [  5   0 182]]
              precision    recall  f1-score   support

           1       0.98      1.00      0.99       197
           2       1.00      1.00      1.00       171
           3       1.00      0.97      0.99       187

    accuracy                           0.99       555
   macro avg       0.99      0.99      0.99       555
weighted avg       0.99      0.99      0.99       555
```

Luego para comparar diferentes valores de k y seleccionar uno, estos se grafican de acuerdo con su medida del acurracy como se muestra en la siguiente grafica.

1. https://www.kaggle.com/chrisfilo/fruit-recognition



El segundo clasificador que se probo fue SVM, y las métricas que miden el desempeño se encuentran a continuación, y luego de varias pruebas se observó que se presentan mejores resultados utilizando un kernel polinomial cuadrático.

```
[[197   0   0]
 [  0 171   0]
 [  2   0 185]]
              precision    recall  f1-score   support

           1       0.99      1.00      0.99       197
           2       1.00      1.00      1.00       171
           3       1.00      0.99      0.99       187

    accuracy                           1.00       555
   macro avg       1.00      1.00      1.00       555
weighted avg       1.00      1.00      1.00       555
```

El tercer algoritmo implementado es el de regresión logística, mediante el cual se obtienen los siguientes resultados de la métrica de mathews y el conjunto de predicciones del clasificador, adicionalmente se muestra el conjunto de los datos reales para comparar con los resultados del clasificador y ver los errores que puedan existir.

1. https://www.kaggle.com/chrisfilo/fruit-recognition

```
matthews_corrcoef 0.975799805998435
Accuracy 0.9837837837837838
[3 2 3 3 1 2 1 1 1 2 1 3 1 1 2 2 1 3 1 3 3 3 1 3 2 1 3 3 3 3 2 3 1 2 3 2 3
 1 1 2 1 1 2 1 1 3 2 1 3 1 2 1 3 1 2 2 1 1 2 3 2 2 2 3 1 3 1 1 2 2 2 1 1 1
 1 3 2 1 3 1 1 3 2 3 1 1 2 1 1 3 1 2 2 2 1 3 2 1 2 2 2 3 1 1 1 1 2 3 3 1 2
 3 1 3 2 2 3 1 3 1 3 2 1 1 1 1 3 1 3 1 3 3 3 2 2 3 2 1 2 1 2 2 3 2 2 2 1 1
 2 1 3 1 2 3 3 3 3 1 3 1 2 3 1 1 1 2 2 2 1 3 2 2 2 3 3 2 3 3 2 2 3 1 3 3 3
 3 2 2 1 1 3 3 1 2 3 3 2 3 3 2 3 2 3 3 2 2 1 2 3 1 2 3 1 2 3 3 1 1 1 2 3 1
 2 3 2 3 1 2 2 3 3 1 3 3 1 1 2 1 3 2 3 1 3 3 2 2 3 3 2 1 1 3 2 1 1 2 3 3 3
 1 1 2 1 2 3 3 3 3 2 3 3 1 1 2 2 2 3 2 2 3 3 3 2 1 2 1 3 1 3 3 3 3 2 1 3 3
 3 2 2 2 2 2 2 1 2 2 3 2 3 2 3 2 2 1 3 1 2 3 3 1 2 3 3 1 1 1 1 3 2 2 2 1 1
 2 1 3 2 2 1 1 3 1 3 3 2 2 3 2 2 1 1 3 1 2 3 3 3 1 2 1 3 3 2 2 3 1 3 1 1 1
 3 1 2 2 3 1 2 3 3 3 1 2 2 1 2 3 3 3 3 2 2 2 3 1 2 1 1 3 3 2 2 2 1 1 2 3 2
 3 3 2 1 3 3 3 1 1 3 2 2 1 3 1 3 3 2 2 3 2 3 1 3 3 3 3 1 3 3 2 2 2 1 3 1 1
 1 3 2 3 1 3 2 2 1 2 3 1 3 3 1 1 2 3 3 3 2 3 2 1 1 3 2 2 1 2 3 3 1 3 3 3 1
 2 2 3 2 1 1 3 3 3 3 1 2 3 2 1 3 1 3 3 2 3 1 3 3 3 3 1 3 1 2 2 2 3 3 2 2 3
 1 3 1 2 2 3 2 1 3 1 3 1 2 3 1 1 2 3 2 2 1 3 1 1 2 1 2 2 3 3 3 2 2 2 1 1 3]
Valores reales
1984    3
1064    2
1590    3
1568    3
2001    3
       ..
1162    2
1082    2
408     1
212     1
1779    3
```

Por último, se implementa una red neuronal artificial a la cual se le calculan las métricas mostradas a continuación.

```
La probabilidad de prediccion es:  [[8.10345228e-08 9.99999547e-01 3.72019538e-07]]
La predicion sobre el conjunto de test es:  [2 1 1 3 2]
El score de la red es:  0.990990990990991
```

4. Conclusiones:

- En primer lugar, se tiene que es de gran importancia la obtención de los datos, es decir, el procesamiento del dataset original para obtener así el dataset que se utilizaría posteriormente en el código. Para esto es de gran importancia saber qué se quiere obtener del dataset escogido.
- Por otro lado, la división de los datos en el conjunto de entrenamiento y el conjunto de prueba también resulta ser de gran importancia, ya que una mala división puede resultar en sobre-entrenamiento o sub-entrenamiento de los clasificadores.
- También se pudo observar que la presencia de tres clases únicamente en los datos de entrada ayuda a obtener mejores resultados en la clasificación, es decir, los resultados podrían variar negativamente al tener una mayor cantidad de clases dependiendo del tipo de clasificador utilizado, por ejemplo, si se hubieran utilizado todas las frutas del dataset original.
- Finalmente, se concluye que el desempeño de los clasificadores es muy bueno en general, esto puede deberse a que el procesamiento de las imágenes se realizó de una manera óptima, obteniendo así un buen dataset. Las variaciones entre los clasificadores en cuanto a las métricas de desempeño son pocas, por lo tanto, solo se aprecia una variación en la complejidad de aplicación de cada uno de ellos, buscando generalmente los métodos que son menos complejos si arrojan buenos resultados.

5. Referencias:

- Creative Commons — Atribución 4.0 Internacional — CC BY 4.0 *(Licencia dataset original)*

1. https://www.kaggle.com/chrisfilo/fruit-recognition

- *Fruit Recognition*. (2020, 4 febrero). Kaggle. https://www.kaggle.com/chrisfilo/fruit-recognition

Link del video youtube: https://youtu.be/-FlVS1eFevI

Link del repositorio:

1. https://www.kaggle.com/chrisfilo/fruit-recognition