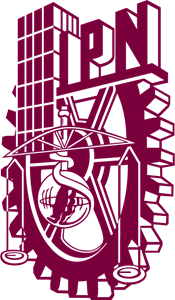
Regiones: Cielo, Bosque y Suelo

Objetivo

Diseñar un clasificador paramétrico bayesiano que identifique tres regiones o áreas en una imagen digital: “Cielo, Bosque y Suelo”

Instituto Politécnico Nacional

*Escuela Superior de Cómputo*

**Unidad de Aprendizaje:** Pattern Recognition

**Práctica 1:** Reporte

**Unidad Temática:** Diseño de un Clasificador Paramétrico Bayesiano para una distribución normal (Distancia Euclidiana)

**Grupo:** 3CM11

**Profesora:** Cruz Meza María Elena

**Fecha de entrega:** 5 de mayo de 2021

**Integrantes:**

Arévalo Andrade Miguel Ángel

Medero Luján Alejandro

Castro Cruces Jorge Eduardo

Franco Ake Alan

[**Introducción**](#_heading=h.8uobgdnf7o0) **4**

[**Marco teórico**](#_heading=h.l8mpbbf5opzl) **4**

[**Diseño de un Clasificador Paramétrico Bayesiano para una distribución normal (Distancia Euclidiana)**](#_heading=h.kb8emtf3ox30) **5**

[Preprocesamiento](#_heading=h.ahu2mgr25ub1) 6

[Segmentación](#_heading=h.7a2lsmt4ddhu) 7

[Multi Umbralización](#_heading=h.jkadyy40b2ng) 9

[Operación Lógica “OR” ….](#_heading=h.16v4dmjk3ihr) 11

[**DESARROLLO**](#_heading=h.v5yvzbhwijh3) **12**

[Clasificador Paramétrico Bayesiano para una distribución normal (Distancia Euclidiana)](#_heading=h.juw7jgj3tyih) 12

[Fase de aprendizaje](#_heading=h.5ra747qv1bmd) 12

[FASE DE RECUPERACIÓN](#_heading=h.8vyryxk2tclc) 15

[**Implementación**](#_heading=h.hff6x8exjwd6) **17**

[Pruebas y resultados](#_heading=h.ev5a0ujmorqs) **18**

[Conclusiones](#_heading=h.o75ph1vp3f69) **19**

[**Fuentes consultadas**](#_heading=h.29whc1iy7n78) **20**

# **Introducción**

En el caso de estudio, que se trata en este documento, se desea resolver un problema diseñando un clasificador basado en métricas que permita identificar o clasificar tres tipos de texturas naturales, es decir, tres regiones en una imagen digital (ver fig. 1) son: Cielo Azul, Zonas Boscosas. En la imagen se distinguen el predominio de áreas verdes y Zonas sin Cultivos, en color ocres, a partir de una imagen de entrada, basadas en tres componentes de color (R, G, B)



# **Marco teórico**

La clasificación consiste en asignar clases o etiquetas a objetos. Hay dos tipos básicos de de clasificación:

**No supervisada:** en este caso las clases son desconocidas, por lo que el problema consiste en dividir un conjunto de objetos en ***n*** grupos o clusters, de modo que se asigne una clase a cada grupo diferente. También se le conoce como ***clustering***.

**Supervisado:** las posibles clases o etiquetas se conocen a priori, y el problema consiste en encontrar un función o regla que asigna cada objeto a uno de las clases.

**Clasificación probabilística**

La clasificación supervisada consiste en asignar a un objeto particular descrito por sus atributos, *A1, A2, ..., An,* una de las m clases*, C = {c1, c2, ..., cm}*, tal que la probabilidad de la clase dados los atributos se maximiza:

**Evaluación del clasificador**

**Precisión:** se refiere a qué tan bien un clasificador predice la clase correcta para ejemplos invisibles (es decir, aquellos no considerados para aprender el clasificador).

**Tiempo de clasificación:** cuánto tarda la clasificación. proceso para predecir la clase, una vez que el clasificador ha sido entrenado.

**Tiempo de formación:** cuánto tiempo se necesita para aprender clasificador de datos.

**Requisitos de memoria:** cuánto espacio en términos de memoria se requiere para almacenar los parámetros del clasificador.

**Claridad:** si el clasificador es fácilmente comprensible para una persona.

**Desequilibrio de clase**

En general queremos maximizar la clasificación precisión; sin embargo, esto solo es óptimo si el costo de un la clasificación incorrecta es la misma para todas las clases

Cuando existe un desequilibrio en los costos de clasificación errónea, debemos minimizar el esperado costo (EC). Para dos clases, esto viene dado por:

**Donde:** FN es la tasa de falsos negativos, FP es el falso tasa positiva, P (+) es la probabilidad de positivo, P (-) es la probabilidad de negativo, C (- | +) es el costo de clasificando un positivo como negativo, y C (+ | -) es el costo de clasificar un negativo como positivo

**Clasificador Bayesiano**

Aplicación de la regla de Bayes:

Que se puede escribir de forma más compacta como:

El problema de clasificación se puede formular como:

Equivalente:

•

•

•

Tenga en cuenta que la probabilidad de los atributos, **P(A)**, no puede variar con respecto a la clase, por lo que puede ser considerada como una constante para la maximización.

**Complejidad**

La aplicación directa de la regla de Bayes da como resultado una problema computacionalmente costoso

El número de parámetros en el término de probabilidad, P (A1, A2, ..., An | C), aumenta exponencialmente con el número de atributos.

Una alternativa es considerar ciertas propiedades como en los modelos gráficos, en particular que todos los atributos son independientes dada la clase, lo que resulta en el clasificador bayesiano ingenuo.

El clasificador bayesiano simple o ingenuo (NBC) se basa en el supuesto de que todos los atributos son independiente dada la variable de clase:

**Naive Bayes**

El clasificador bayesiano simple o ingenuo (NBC) se basa en el supuesto de que todos los atributos son independiente dada la variable de clase:

La fórmula ingenua de Bayes reduce drásticamente la complejidad del clasificador bayesiano, como en este caso solo requieren la probabilidad previa (unidimensional vector) de la clase, y las n probabilidades condicionales de cada atributo dada la clase (bidimensional matrices)

El requisito de espacio se reduce de exponencial a lineal en el número de atributos

El cálculo de la parte posterior se simplifica enormemente, ya que para estimarlo (no normalizado) solo se realizan n multiplicaciones requerido

**Modelo gráfico**

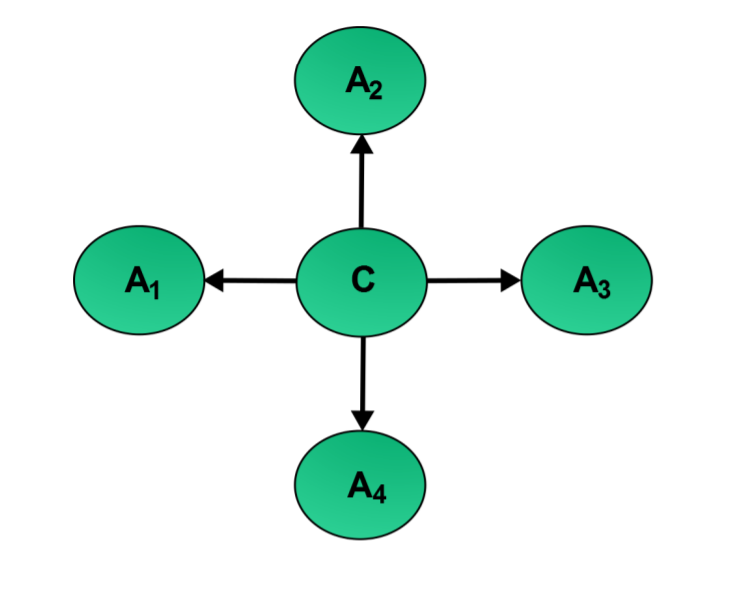
****

Fig. 2 Modelo gŕafico del clasificador bayesiano simple

**Aprendizaje de parámetros**

Las probabilidades se pueden estimar a partir de datos utilizando, por ejemplo, estimación de máxima verosimilitud.

Se dan las probabilidades previas de la variable de clase C por:

Las probabilidades condicionales de cada atributo, Aj, pueden ser estimadas como:

**Inferencia**

La probabilidad posterior se puede obtener simplemente por multiplicar el anterior por la probabilidad de cada atributo.

Dados los valores de ***m*** atributos, a1, ... am, para cada clase ci, el posterior es proporcional a:

La clase ck que maximiza la ecuación anterior debe ser seleccionada.

# Diseño de un Clasificador Paramétrico Bayesiano para una distribución normal (Distancia Euclidiana)

Dada una imagen (fig. 1), en ella podemos distinguir tres tipos de regiones o áreas de interés, las cuales pueden describirse por los colores de los píxeles que conforman tales regiones a las que denominaremos clases Ci:

* Área Cielo, C1
* Área Boscosa, C2
* Área Suelo, C3

De las tres clases o regiones notemos que se tiene datos a priori, de antemano sabemos que cada píxel de f(x,y) corresponderá a un patrón xi. Cada patrón xi corresponderá tiene una pertenencia a una clase, es decir xi ϵ Ci, donde Ci = {C1,C2,C3}. Además, dada la naturaleza de cada píxel, podemos considerar la textura como el rasgo principal de cada región o área a identificar. Con estos datos es posible diseñar un clasificador (fig. 2) que permita identificar a cada una de las clases.

Los elementos que intervienen para este proceso se listan a continuación:

* Preprocesamiento de la imagen
  + Abrir imagen
  + Convertirla a ng
  + (Extracción del canal verde
  + Extraer canales en RGB)
* Segmentación
  + Multi Umbralización
  + Operación puntal OR
  + Generación del descriptor de texturas para cada clase Ci
  + Asociación de cada píxel f(x,y)  Ci o Definición del CMA
* Aprendizaje
  + Vector de entrada (r,g,b)
  + Cálculo del Centroide o vector media Zi para cada clase Ci
  + Algoritmo BM
  + Pruebas
* Implementación del Clasificador
  + Interfaz gráfica
  + Extracción de características
  + Clasificador



Fig. 3 Esquema general del diseño del Clasificador

## Preprocesamiento

Dada la imagen de estudio para resolver el problema, se requiere abrir la imagen para poder extraer los valores de los píxeles, es decir, extraer la textura de cada región o clase ya que se llevarán a cabo la pruebas necesarias para definir si este primer rasgo es el que puede describir adecuadamente a los patrones de cada clase, posteriormente se someterán a la regla de aprendizaje para finalmente llevar a cabo la implementación del clasificador que resuelva el problema de la identificación de las tres regiones o clases.

Especificaciones de la imagen de entrada:

o Imagen en color (R, G, B), a la que denominaremos f(x,y), donde el rango de valores para cada canal va de [0-255] niveles de color. o Píxel (r,g,b), que llamaremos patrón xi  Ci o Tamaño de la imagen: MxN = 334 x 218

Definición del vector de entrada (r,g,b),

La descripción de cada región o clase será mediante el análisis de su textura, por lo que se necesita extraer cada píxel, estos conformarán nuestro vector de entrada para el clasificador.

𝒓

Píxel (x,y) para cada canal (r,g,b)  𝒙𝒊 = (𝒓, 𝒈, 𝒃)  𝒙𝒊 = 𝒈

𝒃

Para clase Ci, se requiere almacenar los píxeles que conforman ese tipo de textura, por ejemplo, tomemos un píxel cualquiera para cada clase, la representación del vector de rasgos que describe a cualquier patrón para cada una de las clases tendrá la siguiente forma:

𝒓 𝟐𝟎𝟎

- Un patrón de la clase boscosa: 𝒙𝒊 ⇒ (𝒓, 𝒈, 𝒃)  𝒙𝒊 = 𝒈 = 𝟏𝟔𝟎 ∈ 𝑪𝟏

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | 𝒃 | 𝟏𝟐𝟎 |
| - | Para la clase cielo: 𝒙𝒊 ⇒ (𝒓, 𝒈, 𝒃)  𝒙𝒊 = | 𝒓  𝒈  𝒃 | = | 𝟗𝟎  𝟏𝟑𝟎  𝟔𝟎 | ∈ 𝑪𝟐 |
| - | Para la clase suelo: 𝒙𝒊 ⇒ (𝒓, 𝒈, 𝒃)  𝒙𝒊 = | 𝒓  𝒈  𝒃 | = | 𝟑𝟎  𝟒𝟒  𝟏𝟕𝟖 | ∈ 𝑪𝟑 |

Demás especificaciones (en caso de que haya…)

## Segmentación

La segmentación subdivide una imagen en sus partes constituyentes u objetos, con el fin de separar las partes de interés del resto de la imagen, por lo tanto, el nivel al que se lleva a cabo esta subdivisión depende del problema a resolver [2]. La segmentación automática es una de las tareas más difíciles del procesamiento de imágenes, esta etapa determina el eventual éxito o fracaso del análisis, de hecho, rara vez llega a alcanzar una solución satisfactoria, se debe buscar un método alternativo de comprobación para la verificación de los resultados. Usualmente hay dos formas de segmentación de la información, mediante el análisis de dos conceptos aplicados a los píxeles: la similaridad o bien mediante la discontinuidad.

Los algoritmos de segmentación de imágenes monocromáticas generalmente se basan en una de las dos propiedades básicas de los valores del nivel de gris: discontinuidad y similaridad [3]. En la discontinuidad el método consiste en dividir una imagen basándose en los cambios bruscos del nivel de gris [4]. Los temas más importantes en la discontinuidad son:

1. Detección de puntos aislados,
2. Detección de líneas
3. Detección de bordes de una imagen.

En la similaridad, se presenta la regularidad en los valores del nivel de gris, los principales métodos están basados en:

1. Técnicas basadas en la umbralización,
2. Crecimiento de región,
3. División y fusión de regiones.

La segmentación de una imagen basada en la discontinuidad o en la similaridad de los valores del nivel de gris de sus píxeles es aplicable tanto a las imágenes estáticas como a las dinámicas (variantes en el tiempo). En la Figura N.º 2 se muestra un ejemplo de segmentación de imagen.

Hemos dicho que la segmentación subdivide una imagen en sus partes constituyentes u objetos, con el fin de separar las partes de interés del resto de la imagen, por lo tanto el nivel al que se lleva a cabo esta subdivisión depende del problema a resolver. En el proceso de detectar las partes en una imagen se identifican bordes de la imagen, o se segmenta está en regiones, líneas o curvas, etc. Otra definición considera a la segmentación como la clasificación de los puntos de la imagen (píxeles), indicando las clases a la que pertenecen los diferentes pixeles. Los atributos básicos de segmentación de una imagen son: la luminancia en imágenes monocromáticas.

Actualmente hay una gran variedad de técnicas que abordan este tema de interés, por ejemplo, las que se pueden agrupar en tres: técnicas orientadas al píxel, a los bordes y a las regiones. Dentro de ellas se pueden destacar los siguientes métodos: línea divisoria de aguas (watershed), una técnica de morfología matemática que a partir de los mínimos en la imagen se aumenta gradualmente el nivel de gris, como si fuera agua que se vierte en un valle, hasta encontrar sus valles vecinos; detección de bordes de las regiones mediante la búsqueda de máximos en el gradiente de la imagen o cruces por cero en la segunda derivada de la imagen; filtros en los que se optimiza una función de costo que considera la exactitud en la posición del borde y la cantidad de bordes detectados; y detección de regiones mediante agrupación de píxeles vecinos con características similares [2,3,5]. Otra técnica interesante es por medio de filtros de mediana adaptados al objeto de inspección, la estimación de una imagen libre de fallas a partir de la imagen tomada del objeto mismo. Mediante simple comparación entre la imagen real y la imagen libre de fallas estimada se segmentan las fallas. Si bien es cierto que con este último método se obtienen excelentes

resultados, es necesario invertir mucho tiempo en el diseño del filtro para conseguir una adaptación al objeto. Típicamente, para el análisis de una imagen es necesario dividirla a priori en cientos de sub imágenes. Cada subimagen posee un filtro morfológico distinto, configurado a partir de las características de la porción del objeto presente en ella

* Extracción de los niveles de gris para cada canal (RGB), especificando el tamaño de cada clase (cuantos pixeles contiene cada una)

Convertir la imagen en escala de grises. imagen tomando el máximo R, G o Nivel B y usarlo como escala de grises nivel de píxel. Esto creará una imagen en escala de grises brillante y puntos brillantes, como las áreas naranja o violeta en él las imágenes de abajo se convierten en blanco.

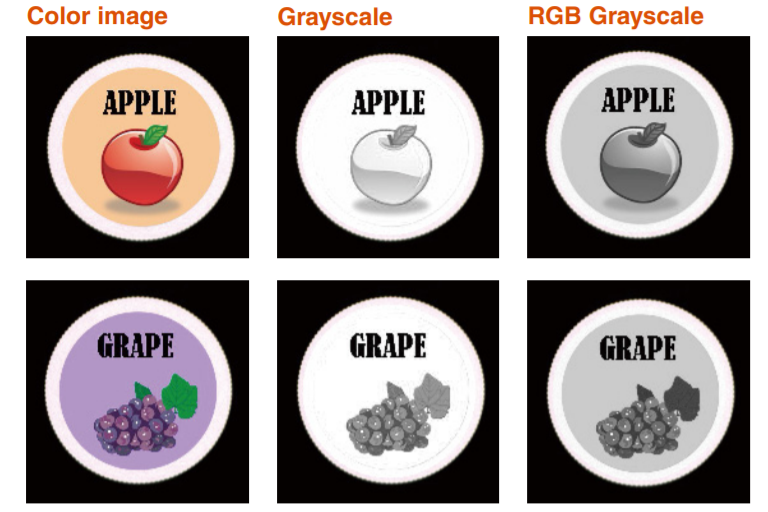


Fig. 4 Imagen ejemplo de la conversión a escala de grises

Convierte la imagen en escala de grises, imagine tomando el promedio de la R,Nivel G y B y usar eso como el nivel de píxeles en escala de grises. Esto creará una imagen en escala de grises cercana a la capturada con una cámara monocromática.

El procesamiento de color a gris convierte una imagen en color en una imagen en escala de grises mediante el uso de un elegido gama de colores como brillo máximo. La el resto de los colores se convierten en una gama de valores de escala de grises basados ​​en los colores seleccionados. Dado que tanto la información de brillo como de color es utilizado, distinción entre colores similares como el oro y la plata se distingue fácilmente como una escala de grises imagen.

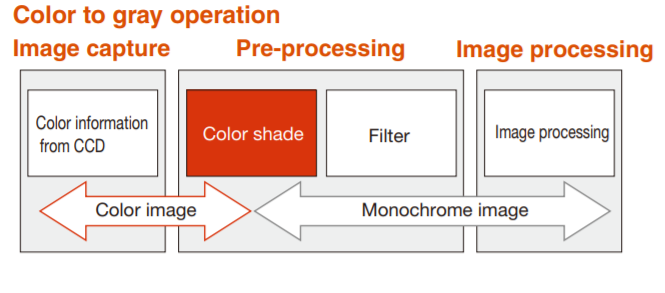


Fig. 5 Diagrama de captura de imagen, preprocesamiento y procesamiento

La etapa de la descripción o extracción de características consiste en extraer características con alguna información cuantitativa de interés o que sean fundamentales para diferenciar una clase de objetos de otra y posteriormente estos datos alimentarán a la fase de reconocimiento, proceso que asignará una etiqueta a un objeto basándose en la información proporcionada por sus descriptores. [1,3] La interpretación implica asignar significado a un conjunto de objetos reconocidos.

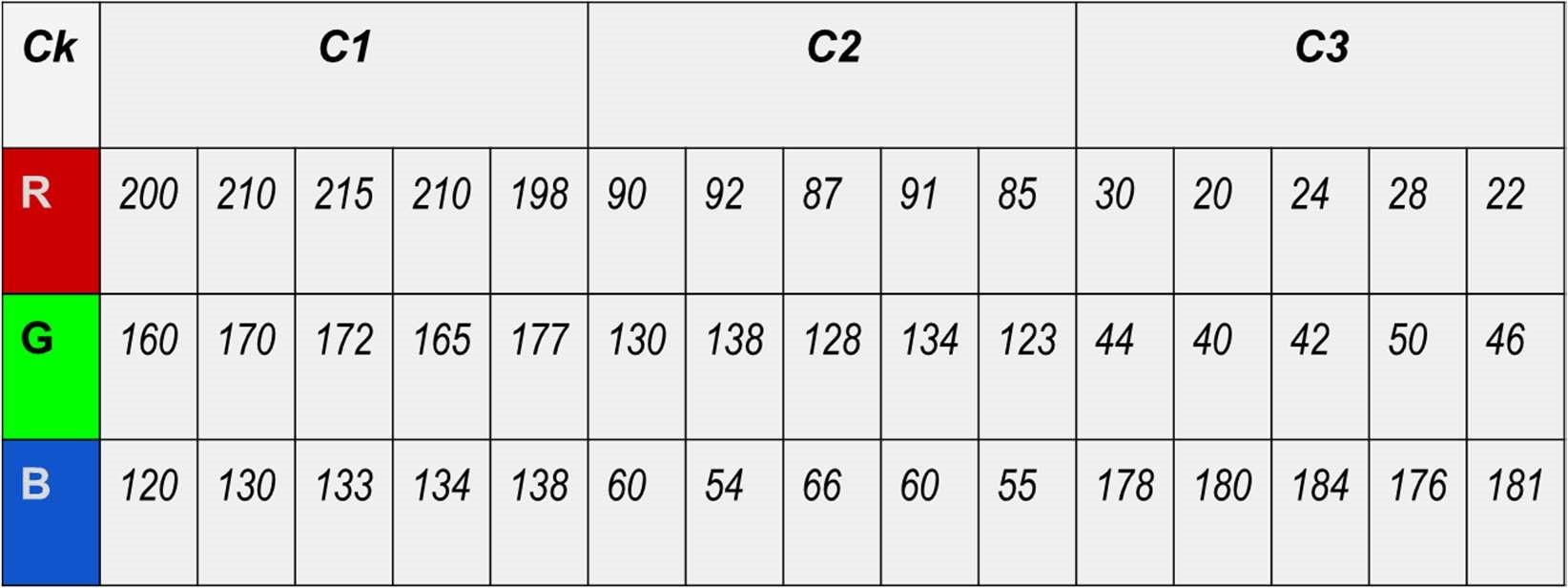


Fig. 6 Matriz que muestra el CMA

### Multi Umbralización

La umbralización es uno de los métodos más importantes de la segmentación de imágenes. Se define el umbral como una función que convierte una imagen con diferentes tonalidades en una imagen en blanco y negro. Si la imagen original es f(x, y), la imagen umbralizada g(x, y) y se fija un umbral U (0 < U < 255), la operación de umbralizado se define como:

g(x, y) = 255 si f(x, y) > Umbral g(x, y) = 0 si f(x, y) ≤ Umbral

Se selecciona un umbral que permita agrupar los píxeles de una imagen pertenecientes a los diversos objetos de la misma imagen diferenciándolos del fondo [1,2,4]. De esta manera la segmentación basada en el histograma se basa en la elección de uno o varios umbrales que permiten agrupar los puntos de la imagen en regiones de características similares en función de sus niveles de gris. El umbral de una imagen se define como:

T = T [x, y, p(x, y), f(x, y)]

Donde:

f(x, y) es la intensidad o nivel de gris del punto (x, y) y p(x, y) representa alguna propiedad local medida en un entorno de vecindad de este punto.

La imagen que se obtiene al llevar a cabo un procesamiento de umbral se define como:

(x, y) = 1 si f(x, y) > T o si f(x, y) ≤ T

De este modo los píxeles marcados con 1 corresponden a objetos, mientras que los píxeles marcados con 0 corresponden al fondo. En la fig, 3 se ha supuesto que la intensidad de los objetos es mayor que la intensidad del fondo. Este proceso es conocido como binarización, por lo que los píxeles por encima del umbral T= 78 corresponden al objeto y los que no serán el fondo [1].

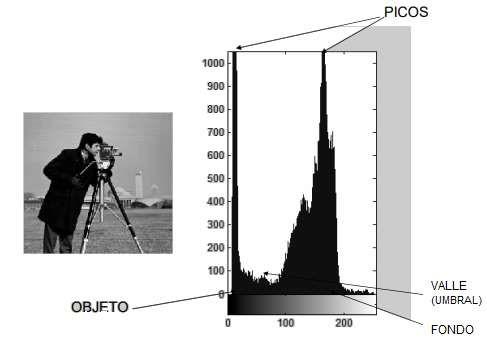


Fig. 7 Imagen “cameraman” en grises y su histograma. Fuente: Gonzalez & Woods



a) b) c)

Fig. 8 Imagen con las distintas transformaciones, a) Imagen original, b) Segmentación de región boscosa y c) Extracción de la región boscosa en b)

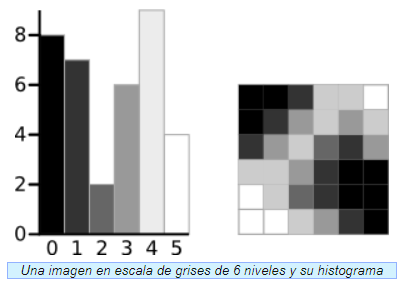
Hay varios tipos de umbrales, los más utilizados:

1. Umbral global, cuando T depende solamente de f(x, y),
2. Umbral local, si T depende de f(x, y) y de p(x, y),
3. Umbral dinámico, si T depende de las coordenadas x e y, además de f(x, y) y de p(x, y).
4. Multi Umbralización

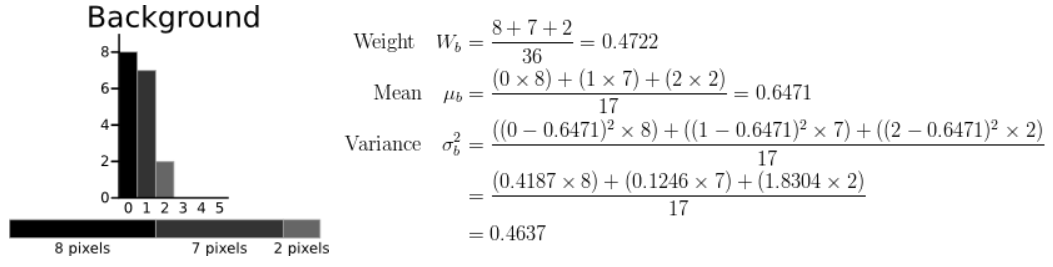
**Umbral de Otsu**

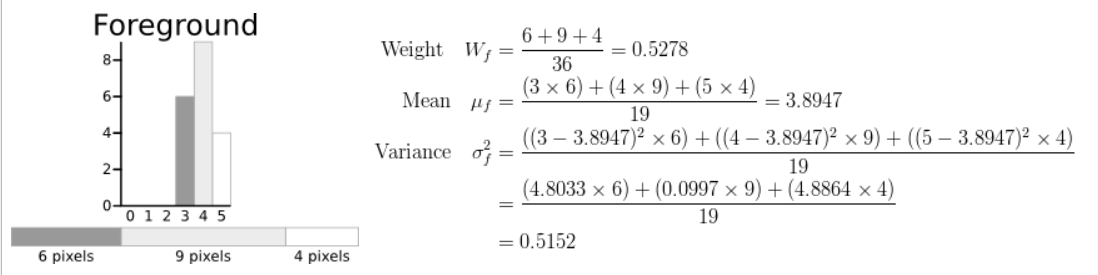
El método de umbral de Otsu implica iterar a través de todos los posibles valores de umbral y calcular una medida de dispersión para los niveles de píxeles a cada lado del umbral, es decir, los píxeles que caen en primer plano o en segundo plano. El objetivo es encontrar el valor de umbral en el que la suma de los diferenciales de primer plano y de fondo es mínima.

El algoritmo se demostrará utilizando la imagen simple de 6x6 que se muestra a continuación. Junto a él se muestra el histograma de la imagen. Para simplificar la explicación, solo se utilizan 6 niveles de escala de grises.

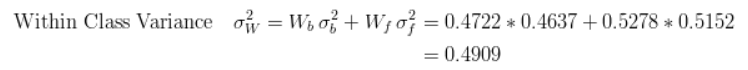


Ahora se muestran los cálculos para encontrar las variaciones de primer plano y de fondo (la medida de la dispersión) para un único umbral. En este caso, el valor umbral es 3.





El siguiente paso es calcular la 'Varianza dentro de la clase'. Esta es simplemente la suma de las dos variaciones multiplicadas por sus pesos asociados.



Este valor final es la 'suma de las variaciones ponderadas' para el valor de umbral 3. Este mismo cálculo debe realizarse para todos los posibles valores de umbral de 0 a 5. La siguiente tabla muestra los resultados de estos cálculos. La columna resaltada muestra los valores del umbral calculado anteriormente.

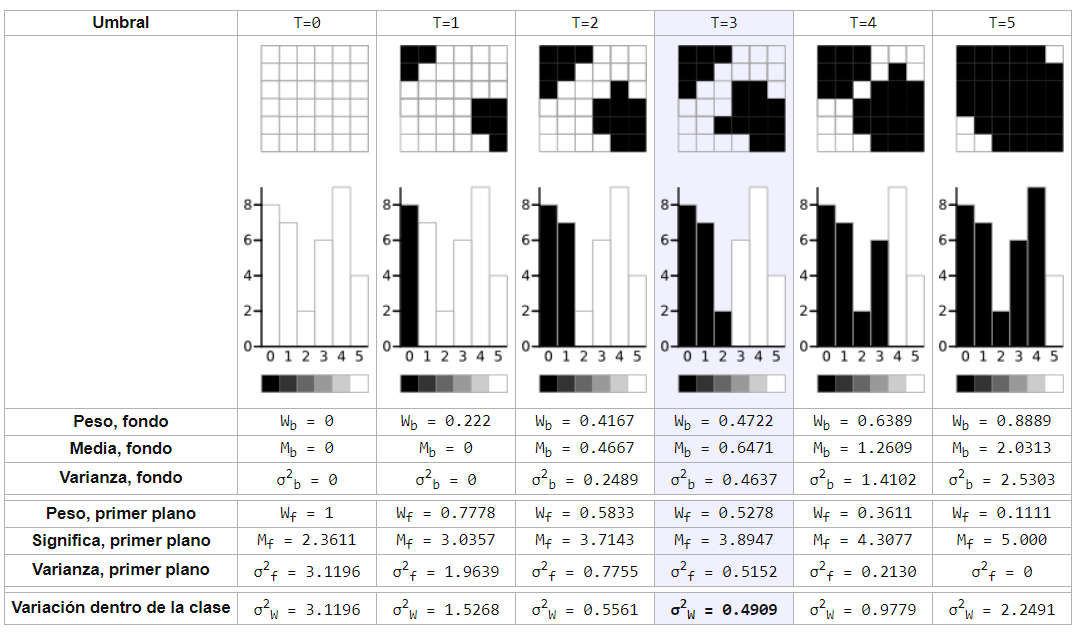


Fig 9. Tabla de resultados

Se puede observar que para el umbral igual a 3, además de ser utilizado para el ejemplo, también tiene la menor suma de varianzas ponderadas. Por lo tanto, este es el umbral seleccionado final. Todos los píxeles con un nivel inferior a 3 son de fondo, todos aquellos con un nivel igual o superior a 3 son el primer plano. Como muestran las imágenes de la tabla, este umbral funciona bien.



### Operación Lógica “OR” ….

# DESARROLLO

Propuesta de solución

## Clasificador Paramétrico Bayesiano para una distribución normal (Distancia Euclidiana)

## Fase de aprendizaje

Recordemos que:

𝑥

𝑥(𝑘) = 𝑥    ó 𝑏𝑖𝑒𝑛, 𝜖 𝑅

Además de que la forma de la función discriminante es:

𝑥

𝑓𝑑(𝑥) = 𝑤 𝑋 ∗ 𝑚 = [𝑤 𝑤 𝑤 ] 𝑥

1

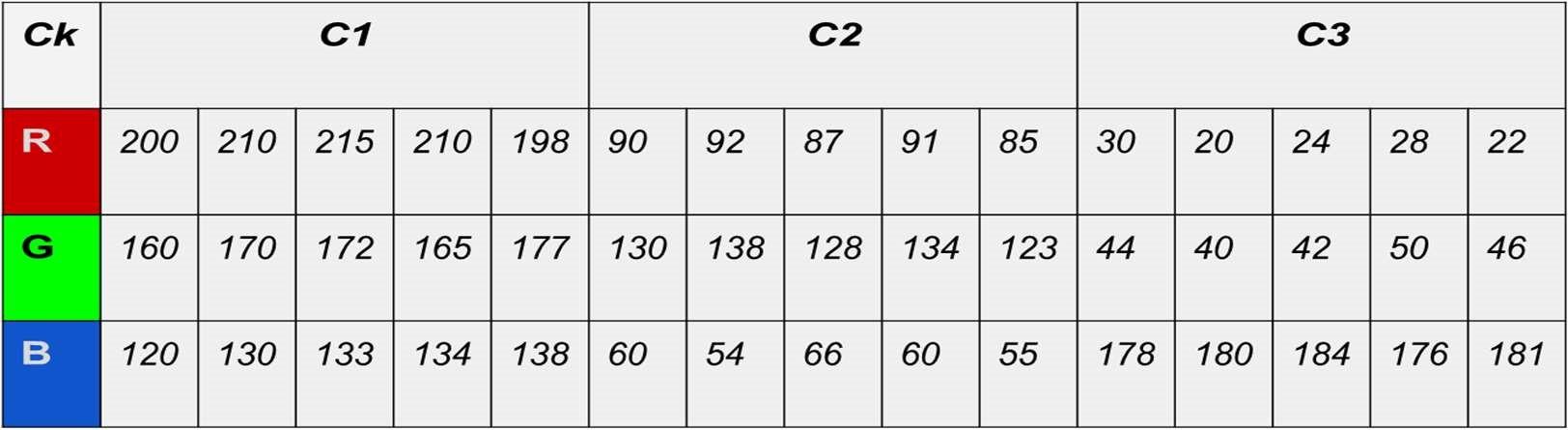
La construcción de la FD lineal debe ser:

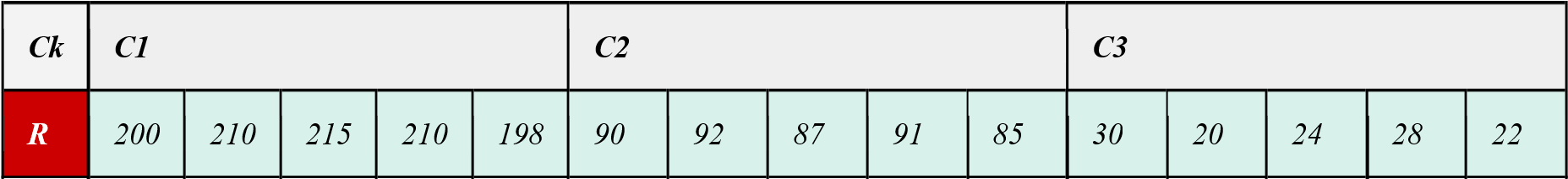
𝑓𝑑(𝑥) = 𝑤 𝑥 + 𝑤 𝑥 + 𝑤

Donde: w=vector de coeficiente de discriminación o ponderación (el vector medio), en este caso se desconocen la media.

La problemática en este tipo de clasificador radica en la importancia y la necesidad de seleccionar los rasgos que mejor representen a las clases, por lo que debe elegirse el par de rasgos que son útiles en la discriminación. Se proponen los rasgos que tengan mayor representación en un rango de color entre las clases.

Tabla de patrones asociados a las clases Cielo, Bosque y Suelo

Fig 9. Tabla de resultados 





Diseño del algoritmo con la 1er. propuesta: Elección de los canales r= rojo y g= verde

1. Se elige una muestra de patrones clasificada de antemano con n clases {c1 , c2 ,.., cn } y la métrica dE (distancia Euclidiana).

 El CMA consta de un total de n muestras para cada clase:

C1 ={X1, … , Xp} con i=1 hasta p

C2 ={X1, … , Xp} con j=1 hasta p

C3 ={X1, … , Xp} con k=1 hasta p

Consideremos el CMA indicado, entonces el cálculo del vector medio:

𝑚 = 1 𝑥 1 200 + 210 + 215 + 210 + 198 = 206

5 2 160 170 172 165 177 168.8

206 89.0 24.8

𝑚 = , 𝑚 = , 𝑚 =

168.8 130.6 44.4

1. Generar funciones discriminantes dij(x) para cada par de clases Ci,Cj, de forma que:

𝑑 (𝑥) = 𝑧 − 𝑧 𝑥 −  𝑧 − 𝑧 𝑧 + 𝑧

𝑑 (𝑥) = ( 𝑧 − 𝑧 ) 𝑥 − [( 𝑧 − 𝑧 ) (𝑧 + 𝑧 )]

2

1

𝑑 (𝑥) = 𝑧 − 𝑧 𝑥 − 𝑧 − 𝑧 𝑧 + 𝑧

2

𝑥 ⎞

⎛

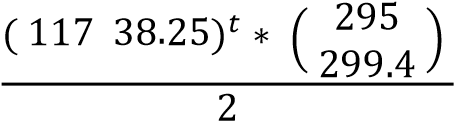
𝑑𝑜𝑛𝑑𝑒 𝑥 = ⎜ 𝑥 ⎟

1

⎝ ⎠

Calculando la primer: Fd1, con𝒎𝟏 =168.8 206 , 𝒎𝟐 = 130.689.0 ,

𝑑 (𝑥) = 168.8 206 − 130.689.0 𝑥 − 12 168.8 206 − 130.689.0 168.8 206 + 130.689.0

𝑑 (𝑥) = ( 117 38.2) . 𝑋 − 

= ( 117 38.2) .𝑋 −  ( 34515 + 11526.9 ) = ( 117 38.2) . 𝑋 − 22983.525

𝑋1

𝑑𝑎𝑑𝑜 𝑞𝑢𝑒 𝑥 = 𝑋2 , 𝑒𝑛𝑡𝑜𝑛𝑐𝑒𝑠:

𝑑 (𝑥) = ( 117 38.2) . 𝑋𝑋12 − 22983.525

𝒅𝟏𝟐(𝒙) = 𝟏𝟏𝟕𝑿𝟏 + 𝟑𝟖.𝟐 𝑿𝟐 − 𝟐𝟐𝟗𝟖𝟑.𝟓𝟐𝟓

206 24.8

Calculando la segunda Fd2 con 𝒎𝟏 = 168.8 , 𝒎𝟑 = 44.4

𝑑 (𝑥) = 168.8 206 − 24.844.4 𝑥 − 12 168.8 206 − 24.844.4 168.8 206 − 44.424.8

𝑑 (𝑥) =

𝑋1

𝑐𝑜𝑚𝑜 𝑥 = , 𝑒𝑛𝑡𝑜𝑛𝑐𝑒𝑠:

|  |  |  |  |
| --- | --- | --- | --- |
| 𝑋2 | |  |  |
|  | 𝑑 (𝑥) = |
| Calculando la tercer Fd3 𝒄𝒐𝒏 𝒎𝟐 = | 89.0  , 𝒎  130.6 𝟑 = | 24.8    44.4 |  |
| 206  𝑚 = , 𝑚  168.8 | 89.0  = , 𝑚  130.6 | 24.8  =  44.4 |  |
| 89.0 24.8 1  𝑑 (𝑥) = − 𝑥 −  130.6 44.4 2 | 89.0 − 24.8 130.6 44.4 | 89.0 | − 24.8 |
| 130.6 44.4 | |

𝑑 (𝑥) =

𝑋1

𝑐𝑜𝑚𝑜 𝑥 = 𝑋2 , 𝑒𝑛𝑡𝑜𝑛𝑐𝑒𝑠:

𝑑 (𝑥) =

Probando la función discriminante Fd1 encontrada. Si se desea conocer por donde pasa la recta solo debemos despejar alguna de las incógnitas (X1 o X2).

También, una vez encontradas las ffdd´s restantes, se pueden obtener las mediatrices:

𝑓𝑑 (𝑥) = 𝑓𝑑 − 𝑓𝑑

𝑓𝑑 (𝑥) = 𝑓𝑑 − 𝑓𝑑

𝑓𝑑 (𝑥) = 𝑓𝑑 − 𝑓𝑑

## FASE DE RECUPERACIÓN

3. En el momento de clasificar (recuperación), el patrón x será clasificado en la clase i si cumple lo siguiente:

 j, k, j i , k  i , x  Ci, si dij(x)  0 y si dik(x)  0

 Probando el clasificador con el patrón desconocido 𝑥? = (208 170 135)

Tomando la fd1

𝑑 (𝑥) = 𝟏𝟏𝟕𝑿𝟏 + 𝟑𝟖. 𝟐 𝑿𝟐 − 𝟐𝟐𝟗𝟖𝟑.𝟓𝟐𝟓

Queda:

𝑑 (𝑥? ) = 117 (208) + 38.2 (170) − 22983.525

𝑑 (𝑥? )= 24336 + 6494 -22983.525 =7846.475

Verificar que se cumple:

 j, k, j i , k  i , x  Ci, si dij(x)  0 y si dik(x)  0

Para conocer a qué clase pertenece x?, x?  C?

2da. Propuesta: elección de los canales g= verde y b= azul



# Implementación

## SEGMENTACIÓN

Antes de poder segmentar la imagen en zonas de interés, se convierte en escala de grises con la función rgb2gray de OpenCV. Esto con la finalidad de no tener que trabajar con valores de colores RGB en escala de 0 a 255.

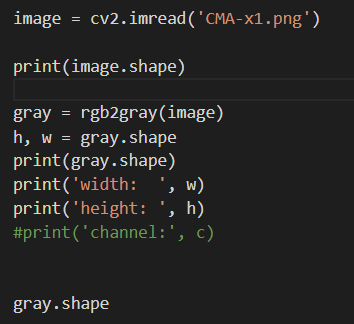


Fig 10. Segmento de código

Una vez obtenida la imagen en escala de grises, se toma la media de los valores de los píxeles y la usaremos como umbral. Si el valor del píxel es mayor que nuestro umbral, podemos decir que pertenece a una zona de interés. Si el valor del píxel es menor que el umbral, se tratará como fondo.

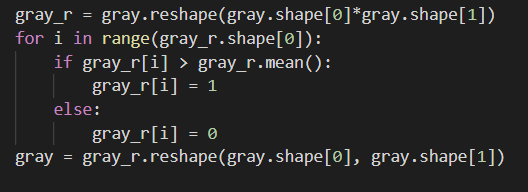
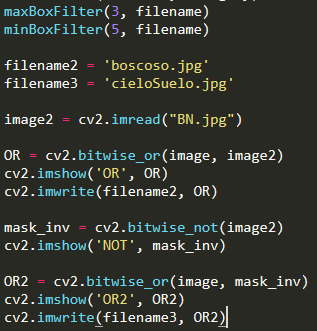


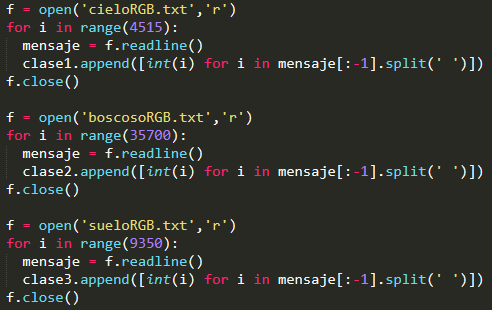
Fig 11. Segmento de código

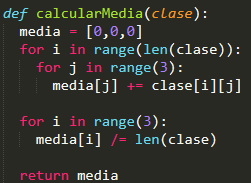
Se le aplicó 3 filtros máximos y 3 mínimos a la imagen para obtener la zona boscosa en blanco y negro, se hizo un ‘or’ con la imagen original para obtenerla con sus colores originales, se le aplicó un ‘not’ a la imagen de la zona boscosa en B/N para obtener las otras 2 clases y aplicar un ‘or’ para obtenerla en color original.

Fig 12. Segmento de código

## ALGORITMO

Se insertan los pixeles del CMA a su respectiva clase.

Fig 13. Segmento de código



# 

# 

Fig 14. Segmento de código

Mediante la media se sacan los centroides

de cada clase.

# 

Fig 15. Segmento de código

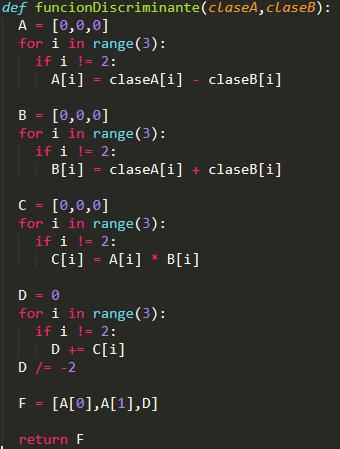
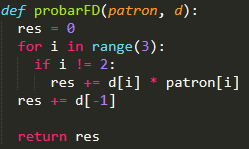


Fig 16. Segmento de código

Se utilizaron los canales rojo y verde como las

características, se calcula la función discriminante.

Fig 17. Segmento de código



# 

# 

Fig 18. Segmento de código

Se prueba la función discriminante con el patrón

desconocido

Fig 19. Segmento de código

La regla de aprendizaje utilizada fue la siguiente

# 

Fig 20 . Segmento de código

Se usó la regla aprendizaje propuesta en clase y además se tuvo que poner 2 condiciones adicionales, nos dimos cuenta que al hacer clic en el suelo los valores que a veces nos arrojaba eran de d1 y d2 negativos y d3 positivo, y en caso del bosque, a veces nos daba d2 y d3 negativos y d1 positivo, así que se agregaron esas condiciones

# 

# Pruebas y resultados

A continuación se presentan las pruebas y los resultados de la implementación del Clasificador Paramétrico Bayesiano para una distribución normal (Distancia Euclidiana):

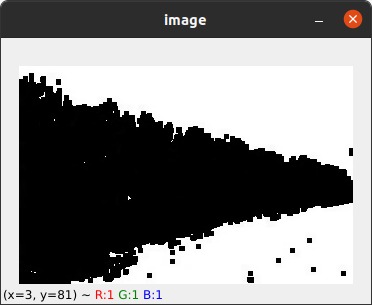
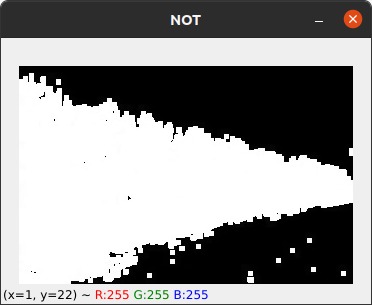
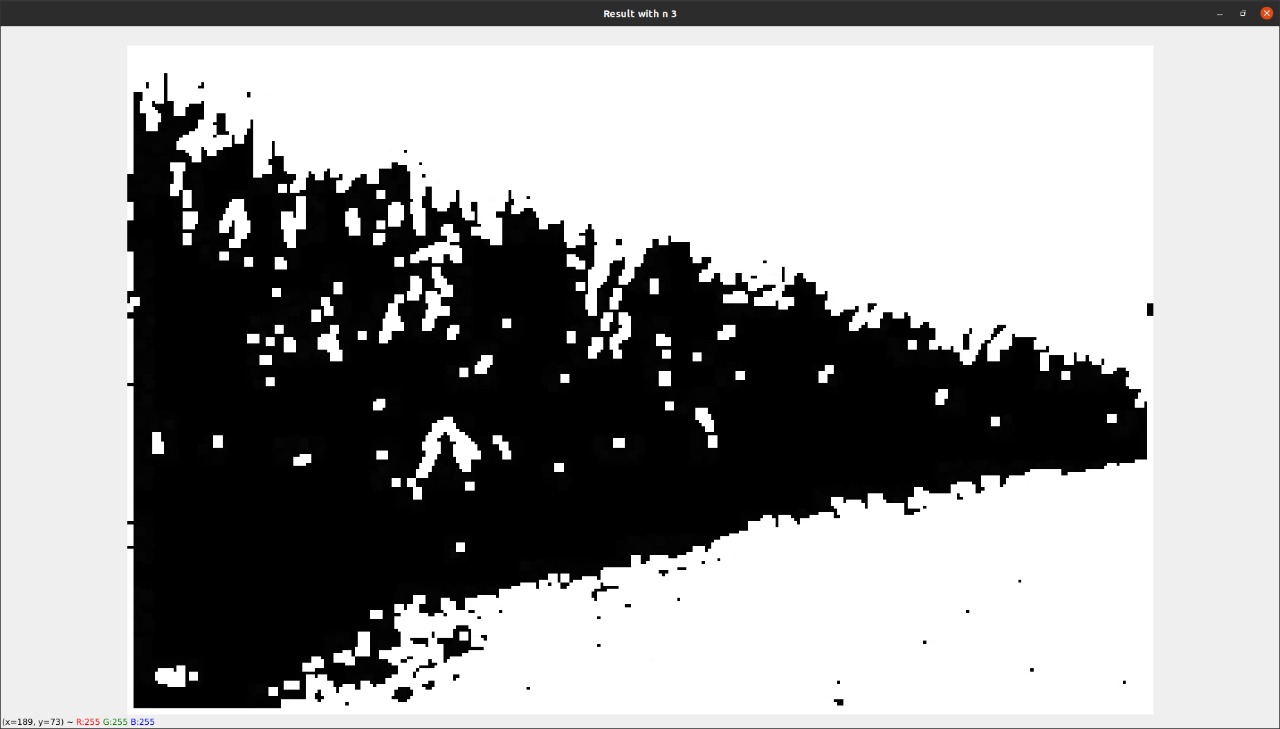
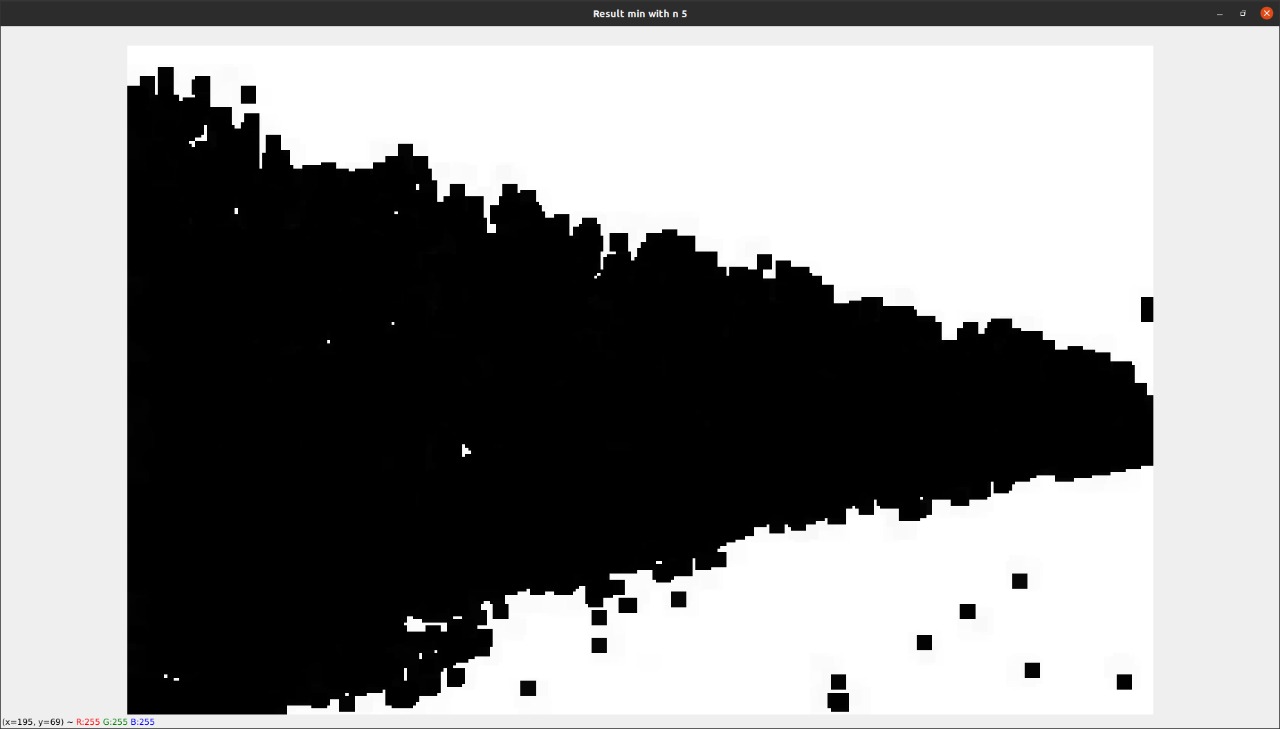
Primeramente, se convierte la imagen original a escala de grises y se binariza

Fig 21 . Imagen binarizada e invertida

Después, se aplica una dilatación con un filtro de 3x3 a la imagen anterior

Fig 22. Imagen binarizada e Imagen dilatada con kernel de 3x3  


Después, se aplica una erosión con un filtro de 5x5 a la imagen anterior

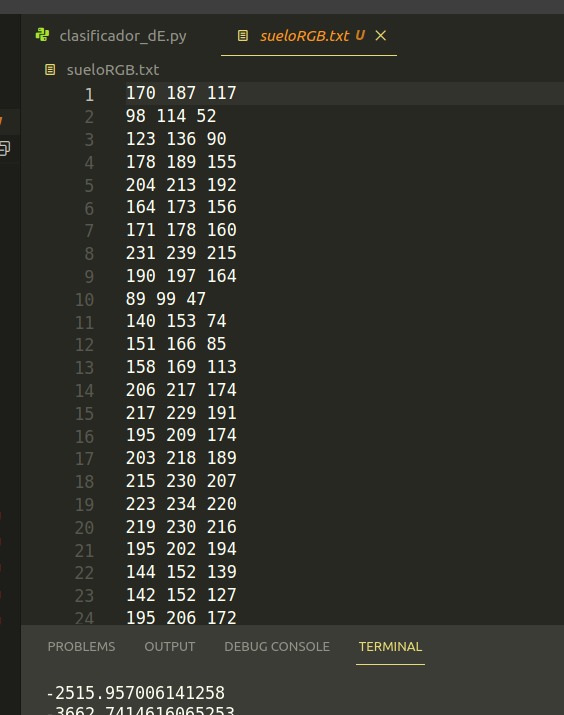
Fig 23. Imagen erosionada con kernel de 5x5

Después, se aplicó la operación lógica OR entre la imagen original y la imagen binarizada

Fig 24. Imagen resultante de la operación OR

Después, se aplicó la operación lógica OR entre la imagen original y la imagen binarizada invertida

Fig 25. Imagen resultante de la operación OR 

Acto seguido, se procede a la extracción de CMA, y dicha información se almacena en archivos .txt

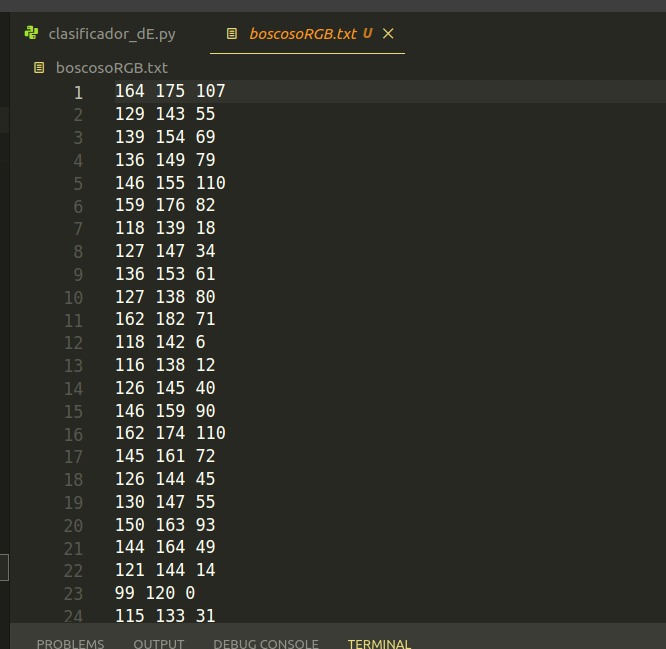
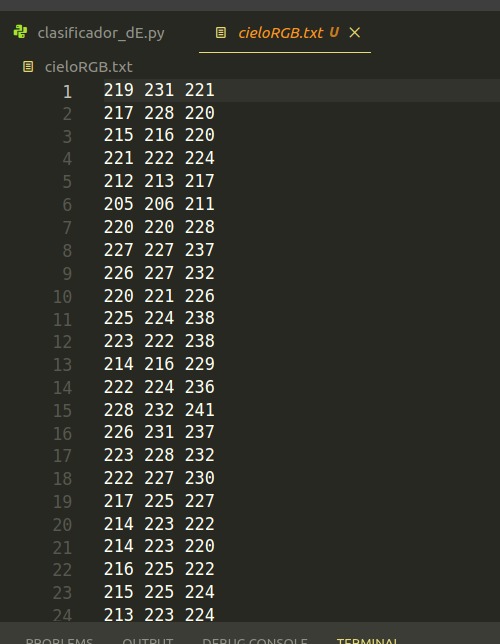


Fig 26. Conjunto de Muestras de Aprendizaje

A continuación se muestran algunas pruebas de escritorio y sus respectivos resultados

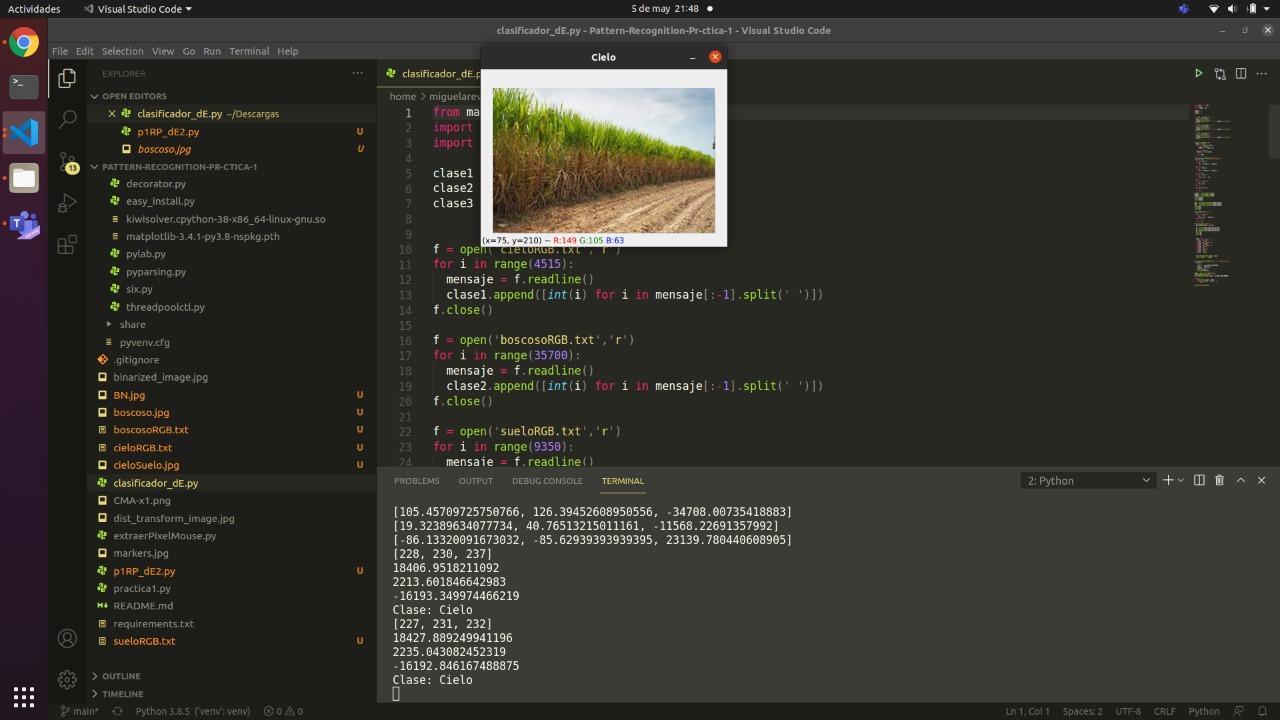
Fig 27. Primer prueba de escritorio

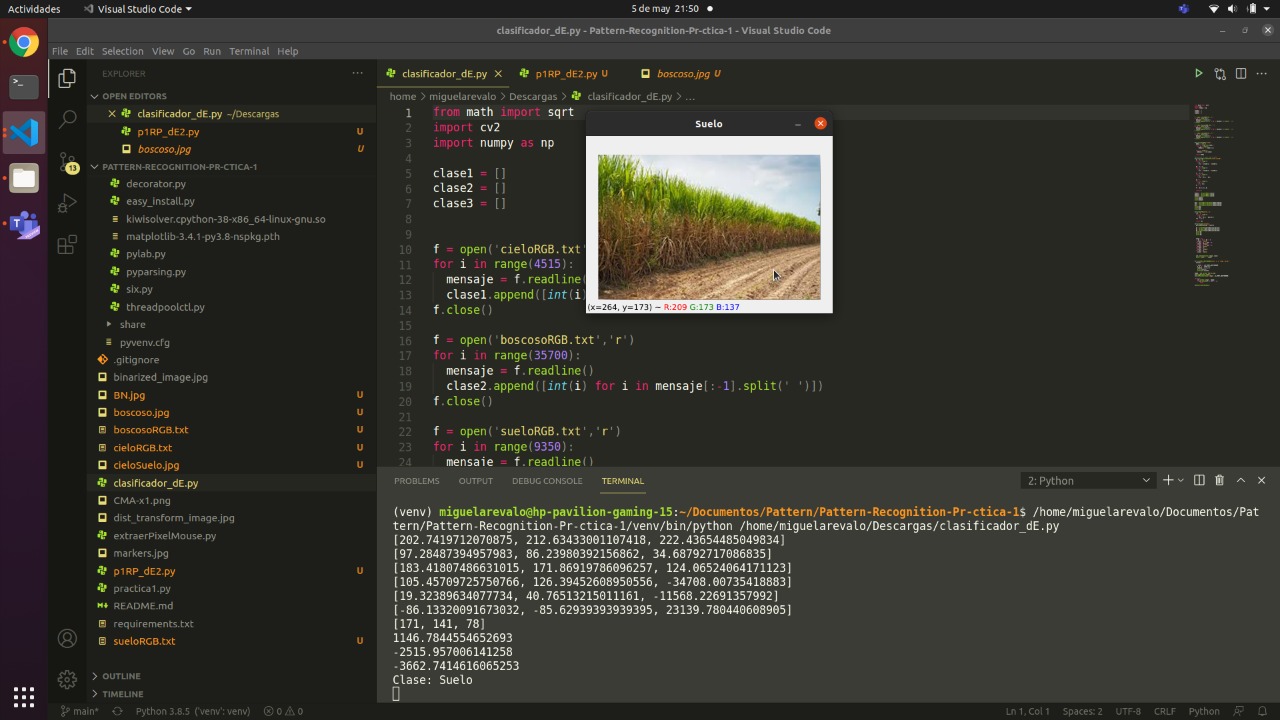
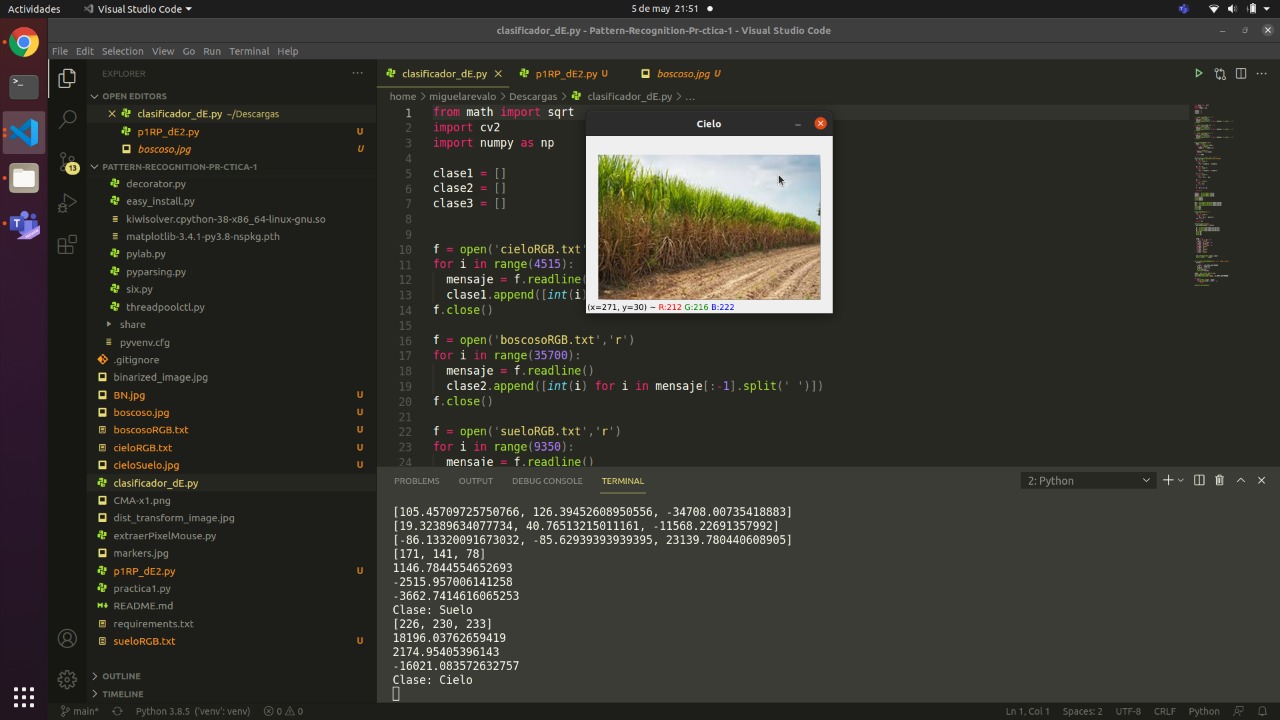
Fig 28. Segunda prueba de escritorio

Fig 29. Tercer prueba de escritorio

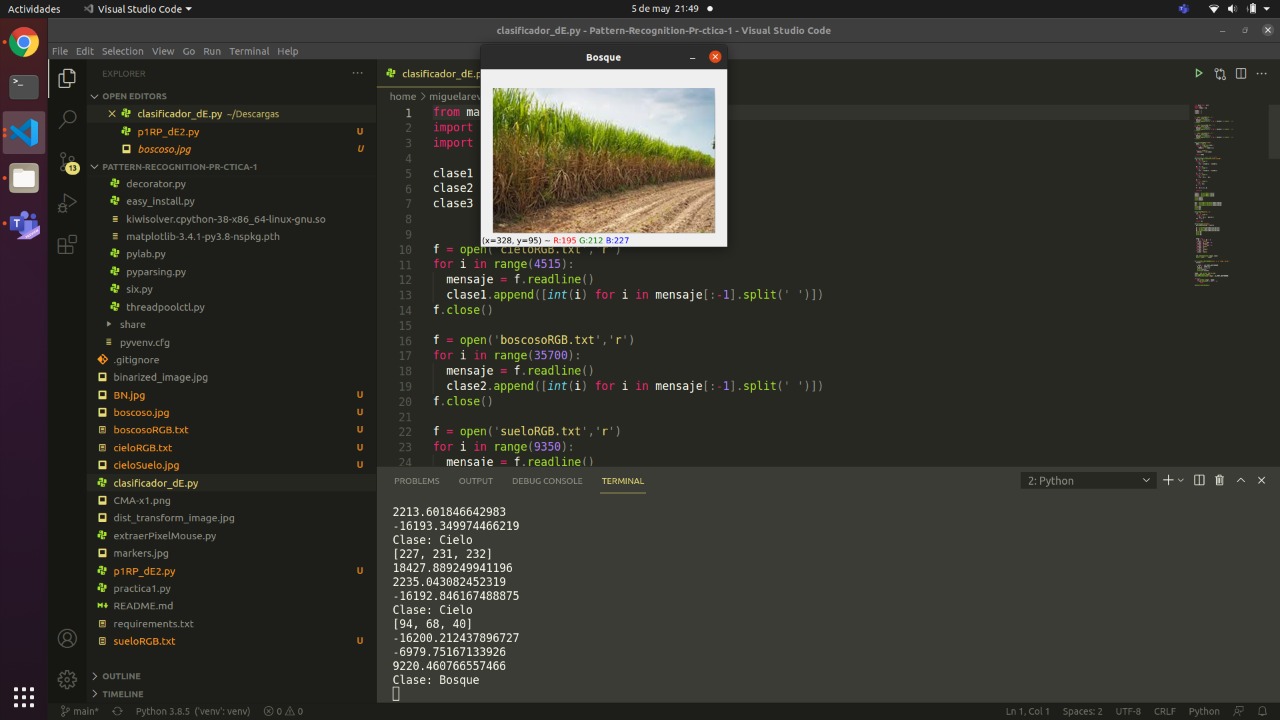
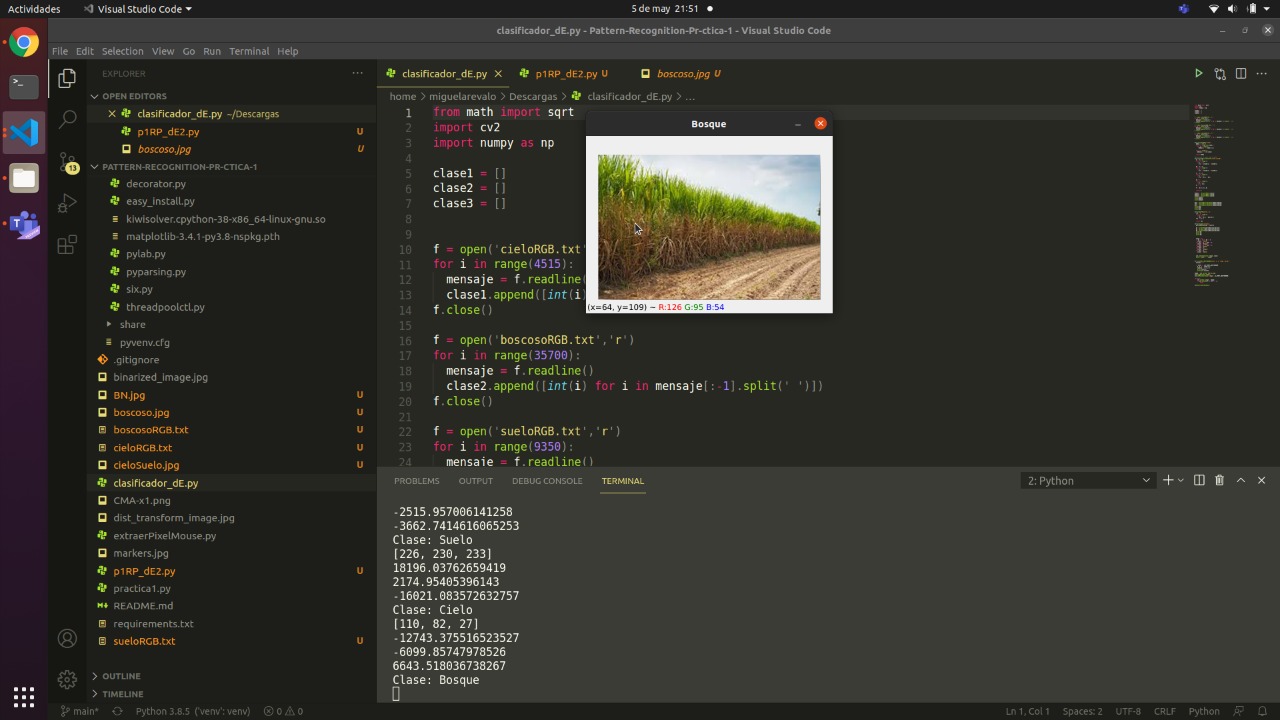


Fig 30. Cuarta prueba de escritorio

Fig 31. Quinta prueba de escritorio

# Conclusiones

**Objetivo General:**

* Diseñar un algoritmo basado en el enfoque estadístico probabilístico o basado en una norma con aprendizaje supervisado que permita identificar una zona o área de interés en una imagen digital.

**Objetivos Particulares:**

* Se deben identificar los parámetros requeridos que describan a las clases de las áreas boscosas, tierra y cielo presentes en una imagen digital.
* Analizar el tipo de aprendizaje supervisado que puede resolver el problema de la identificación para los casos de:
  + Clasificación basada en una norma, o bien,
  + Para los casos de aproximación paramétrica o no paramétrica,
  + Para el diseño de FD para clases con Distribución Normal o Gaussiana, o mediante,
  + La ventana de Parzen.

Retomando los objetivos previos a la realización de la práctica, podemos concluir lo siguiente:

* Primeramente, nosotros implementamos un Clasificador Paramétrico Bayesiano para una distribución normal (Distancia Euclidiana).
* Se logró procesar la imagen deseada para su:
  + Separación de canales
  + Conversión a escala de grises
  + Binarización aplicando multi-umbralización
  + Segmentación
* Se extrajo el Conjunto de Muestras de Aprendizaje de la imagen previamente procesada.
* Se aplicó la metodología vista en clase para la clasificación de vectores desconocidos usando el distancia Euclídea.
* Se muestra el resultado de la clasificación en consola.

En conclusión, se lograron satisfactoriamente los objetivos previamente establecidos.

# Fuentes consultadas

* Bielza, C., Li, G., Larranaga, P.: Multi-Dimensional ˜ Classification with Bayesian Networks. International Journal of Approximate Reasoning, (2011)
* Borchani, H., Bielza, C., Toro, C., Larranaga, P.: ˜ Predicting human immunodeficiency virus inhibitors using multi-dimensional Bayesian network classifiers. Artificial Intelligence in Medicine 57, 219–229 (2013)
* Cheng, J., Greiner, R.: Comparing Bayesian Network Classifiers. Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, 101–108 (1999)
* Friedman, N., Geiger, D., Goldszmidt, M.: Bayesian Network Classifiers. Machine Learning 29, 131–163 (1997)
* Gonzálo Pajares Martinsansz & Jesús M. de la Cruz García. Visión por computador. Ed. Alfaomega Ra-MA. Edición. Primera Edición. Abril 2008
* Silverman, B.W. (1986) Density Estimation for Statistics and Data Analysis. Chapman and Hall, London.
* Wand, M. P. and Jones, M. C. (1995).Kernel Smoothing. Chapman and Hall, London.
* Bowman, A.W. and Azzalini, A. (1997). Applied Smoothing Techniques for Data Analysis: the Kernel Approach with S-Plus Illustrations. Oxford University Press, Oxford