MÓDULO 2 SESIoN 2

PROCESADO DE IMAGEN Y VISIÓN POR COMPUTADOR



8 de noviembre de 2023

Ricardo Martínez Guadalajara

# Introducción

En esta sesión se va a trabajar con clasificación mediante SVM

# Objetivo

Desarrollar los apartados del guion de prácticas como programas de Python

# Contenido

## Apartado 1

El script `SVM.py` es un ejemplo de cómo usar la Máquina de Soporte Vectorial (SVM) en OpenCV para clasificar datos. Aquí está lo que hace cada parte del script:

Definición de la función `get\_data`: Esta función carga los datos de un archivo en formato svmlight/libsvm, convierte los datos y las etiquetas a los tipos de datos correctos y devuelve los datos y las etiquetas.

Carga de los datos de entrenamiento: Utiliza la función `get\_data` para cargar los datos de entrenamiento y las etiquetas del archivo "Puntos.txt".

Configuración de la SVM: Crea una SVM y configura sus parámetros. Establece el tipo de SVM a `SVM\_C\_SVC`, que es una SVM de clasificación con un parámetro de penalización C. Establece el criterio de terminación a un máximo de 100 iteraciones o una precisión de 1e-6. Establece el kernel a lineal y configura los parámetros gamma y C.

Entrenamiento de la SVM: Entrena la SVM con los datos de entrenamiento y las etiquetas, y guarda el modelo entrenado en "model.xml".

Visualización de los resultados del entrenamiento: Crea una imagen en la que los píxeles se colorean en función de la clase que la SVM predice para ellos. Dibuja los puntos de entrenamiento en la imagen. Si la SVM utiliza un kernel lineal, también dibuja los vectores de soporte.

Guardado y visualización de la imagen: Guarda la imagen en "Resultado.png", muestra el tiempo que tardó el script en ejecutarse, muestra la imagen en una ventana y espera a que el usuario cierre la ventana.

**¿Qué representan las regiones verde y azul?**

Las regiones verde y azul representan las áreas de decisión de la Máquina de Soporte Vectorial (SVM). Cada color representa una clase diferente. Por ejemplo, si estás clasificando datos en dos categorías, una categoría se representará con el color verde y la otra con el color azul. Cualquier punto nuevo que caiga en la región verde será clasificado en la categoría verde, y lo mismo para la región azul.

**¿Qué son los puntos blancos y negros?**

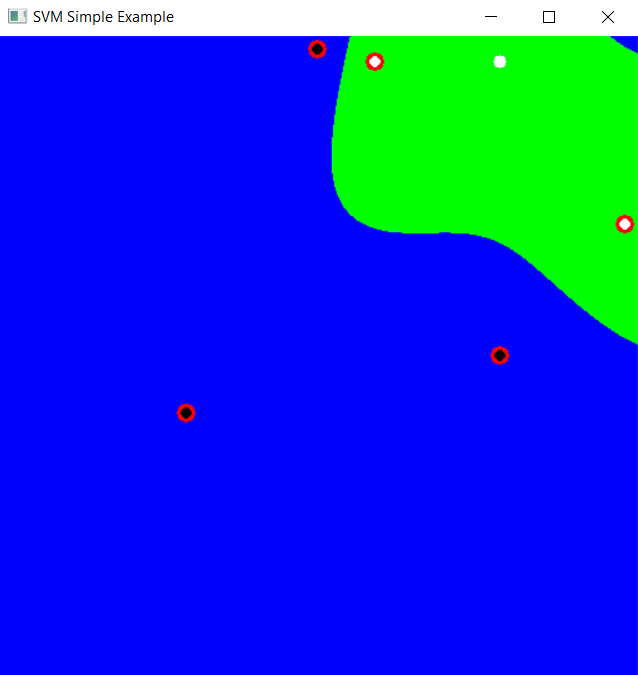
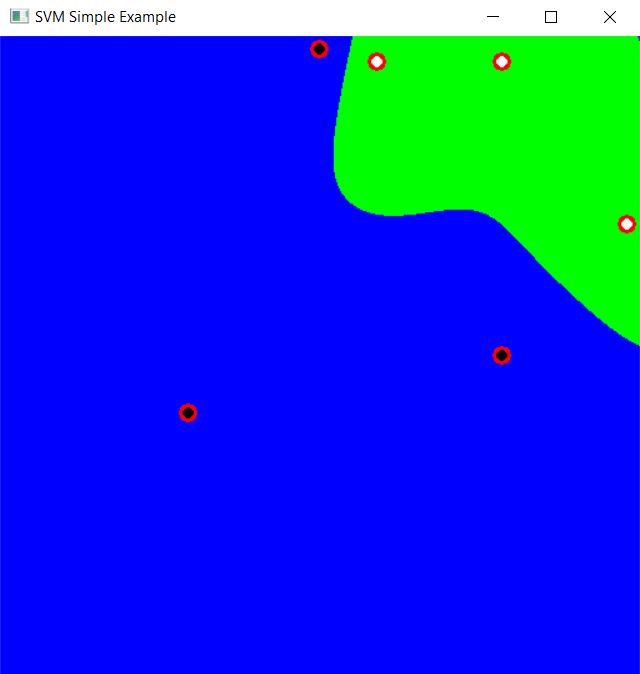
Los puntos blancos y negros representan los puntos de entrenamiento. Estos son los datos que se utilizaron para entrenar la SVM. Cada punto representa una entrada de datos, y el color del punto indica la categoría de esa entrada de datos. En este caso, los puntos blancos representan una categoría y los puntos negros representan la otra categoría.

**¿Qué representan los puntos marcados en rojo?**

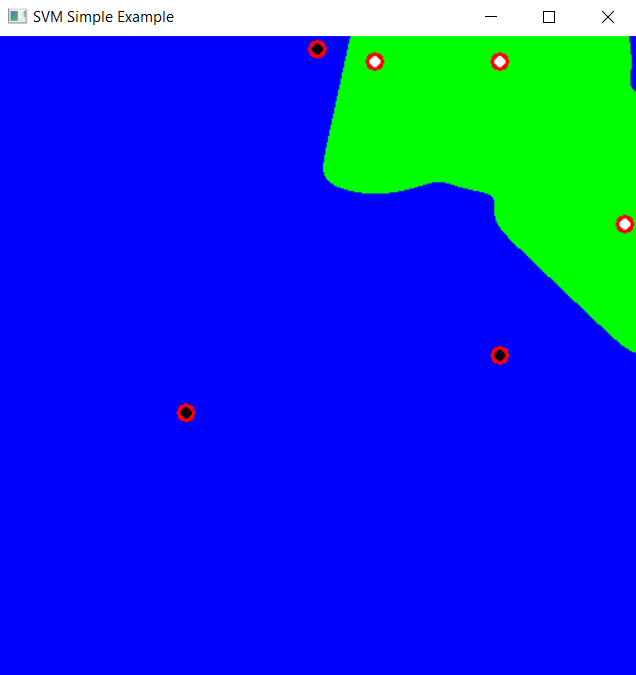
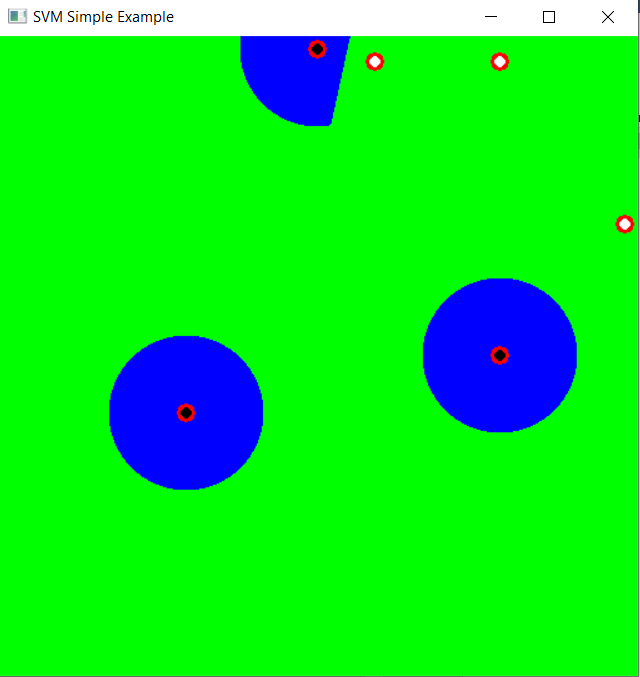
Los puntos marcados en rojo representan los vectores de soporte. En una SVM, los vectores de soporte son los puntos de datos que se encuentran en el límite entre las categorías y ayudan a definir el hiperplano de decisión. Estos puntos son críticos para la SVM, ya que si se eliminan, cambiaría la posición del hiperplano de decisión.

**Modifique el código para que en vez de ser una frontera lineal la frontera de decisión sea  
curva.**

ResultadoRBF\_Gamma0.0001 ResultadoRBF\_Gamma0.0002

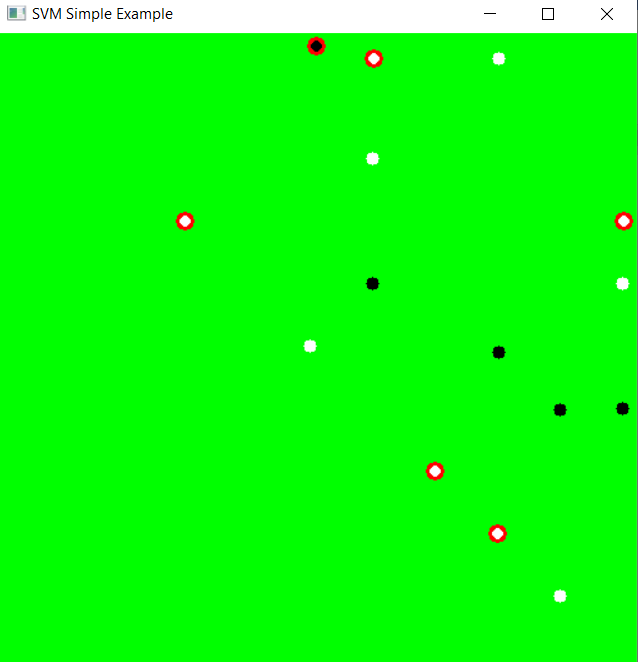
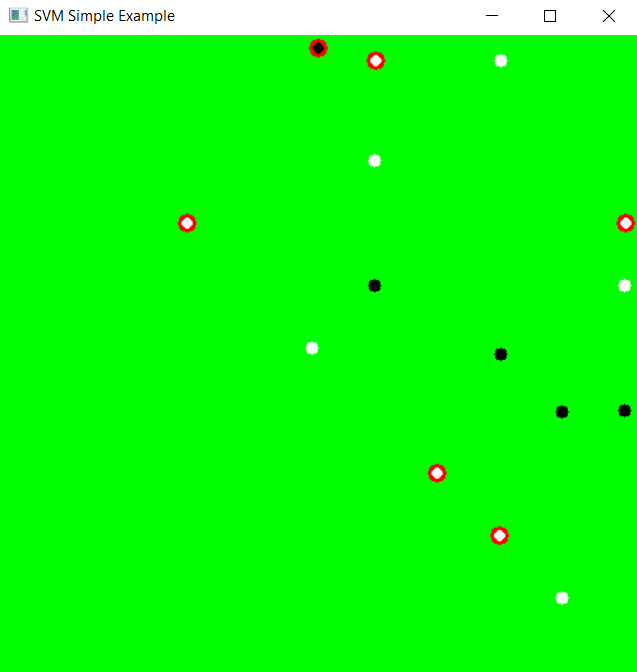
 

REsultadoRB\_Gamma0.001 ResultadoRBF\_Gamma0.01

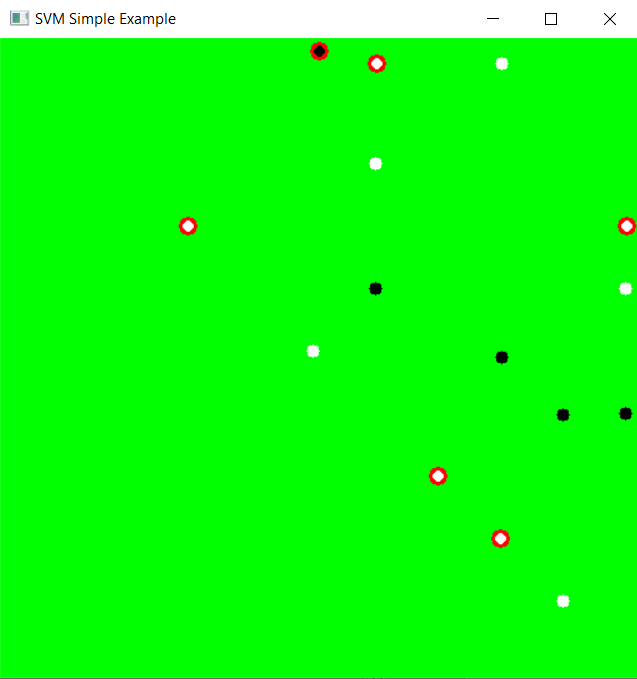
 

**Añada más puntos al problema de forma que este no sea linealmente separable (Fichero  
Puntos.txt). Con 4 puntos puede ser suficiente pero puede añadir más. Realice la  
clasificación con el kernel Lineal y con el RBF. Pruebe con distintos valores de Gamma y C.  
Guarde los resultados en imágenes con distintos nombres.**

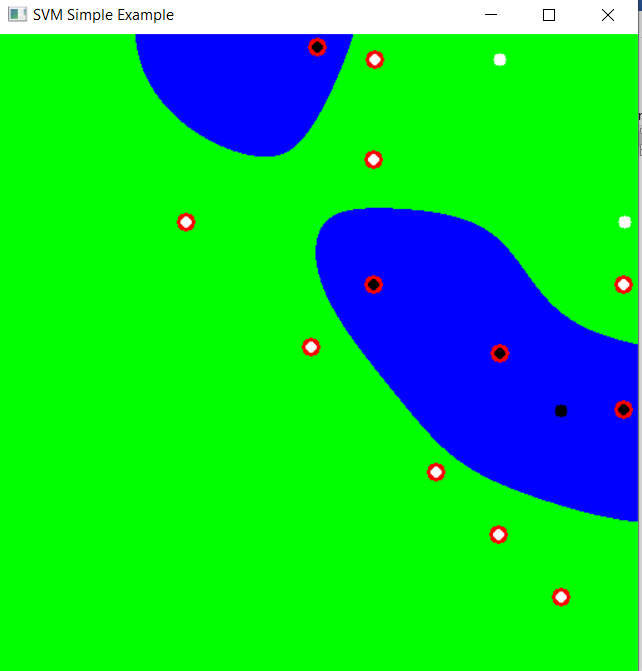
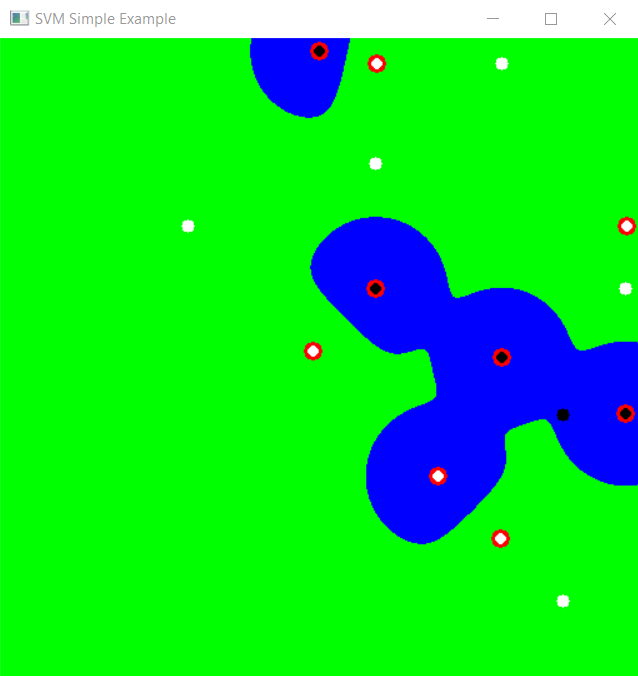
ResultadoLinealMorePoints\_Gamma0.0001 ResultadoLinealMorePoints\_Gamma0.001

ResultadoLinealMorePoints\_Gamma0.01



ResultadoRBFMorePoints\_Gamma0.0001 ResultadoRBFMorePoints\_Gamma0.001

ResultadoRBFMorePoints\_Gamma0.01

