



Nombre:

Samano Cardenas Andrea Guadalupe

Materia:

Inteligencia Artificial

Carrera:

Ing. Sistemas Computacionales

TAREA 5:

Clasificación con SVM con función radial

Clasificación con SVM con función radial

¿Qué es un SVM?

Las Máquinas de Vectores de Soporte (SVM) son un algoritmo de aprendizaje automático supervisado que se utiliza principalmente para problemas de clasificación.

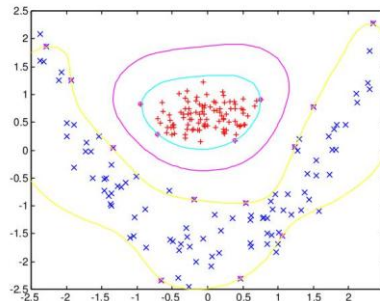
El objetivo de SVM es encontrar un hiperplano que separe las diferentes clases de datos con el mayor margen posible. Este margen se define como la distancia entre el hiperplano y los puntos de datos más cercanos de cada clase, conocidos como vectores de soporte.

El kernel RBF es una de las funciones de kernel más populares utilizadas en SVM.

Permite que el algoritmo maneje datos que no son linealmente separables al transformar el espacio de entrada en un espacio de dimensión infinita. Esto significa que el SVM puede encontrar un hiperplano que separe las clases de manera efectiva, incluso en situaciones complejas donde los datos están distribuidos de forma no lineal.

La función kernel de base radial (RBF) es una de las funciones de kernel más utilizadas en SVM. Se utiliza cuando los datos no lineales se pueden separar utilizando un hiperplano no lineal. La función kernel RBF se define como: donde γ es un parámetro que controla la amplitud de la función kernel. Un valor más alto de γ se ajustará perfectamente al conjunto de datos de entrenamiento, lo que provoca un sobreajuste. $\gamma = 0,1$ se considera un buen valor por defecto.

radial SVM



CODIGO

Python

```
# Importar las bibliotecas necesarias
import numpy as np
from sklearn import datasets
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.metrics import classification_report, accuracy_score

# Cargar un conjunto de datos (por ejemplo, Iris)
data = datasets.load_iris()
X = data.data # Características
y = data.target # Etiquetas

# Dividir los datos en conjuntos de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
random_state=42)

# Crear el modelo SVM con kernel radial (RBF)
svm_rbf = SVC(kernel='rbf', C=1.0, gamma='scale') # Puedes ajustar C y gamma
según sea necesario

# Entrenar el modelo
svm_rbf.fit(X_train, y_train)

# Realizar predicciones
y_pred = svm_rbf.predict(X_test)

# Evaluar el modelo
print("Accuracy:", accuracy_score(y_test, y_pred))
print("\nReporte de Clasificación:\n", classification_report(y_test, y_pred))
```

