Práctica 7: Máquina de vector soporte

Profesor: Lauro Reyes Cocoletzi

Diego Castro Elvira
dcastroe2100@alumno.ipnx.mx

UPIIT: Unidad Profesional Interdisciplinaria en Ingeniería Campus Tlaxcala Instituto Politécnico Nacional, Tlaxcala, Tlaxcala, México 9000

Ingeniera en Inteligencia Artificial

13 de diciembre 2023

***Resumen*— Las Máquinas de Soporte Vectorial (SVM) son modelos de aprendizaje supervisado utilizados para clasificación y regresión. Su objetivo es encontrar un hiperplano óptimo que maximice la separación entre clases en el espacio de características. Para permitir cierta flexibilidad y manejar errores de entrenamiento, las SVM introducen el concepto de "margen suave" mediante el parámetro C, que equilibra los errores y los márgenes rígidos. Las SVM son versátiles y pueden manejar datos no lineales mediante funciones kernel, que realizan transformaciones no lineales en el espacio de características original. Ejemplos de kernels incluyen el lineal, polinómico, radial (RBF), y sigmoidal**

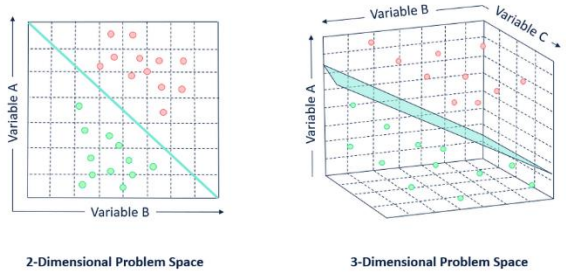
***Palabras clave* — Máquina de Vector Soporte, hiperparametros, kernel, margen**

I. Marco Teorico

A. ¿Qué son las Maquinas de Soporte vectorial?

Una máquina de soporte vectorial (SVM, por sus siglas en inglés Support Vector Machine) es un tipo de modelo de aprendizaje supervisado que se utiliza tanto para tareas de clasificación como de regresión.

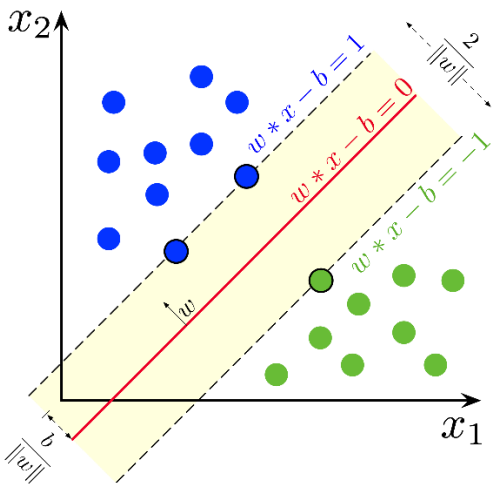
La idea principal detrás de las SVM es encontrar un hiperplano óptimo de separación entre las clases en el espacio de características. Un hiperplano es un subconjunto de dimensionalidad uno menor que el espacio de características y, en el caso de las SVM, se busca el hiperplano que maximice la separación entre las clases.



B. Soft margin: Errores de formación

Idealmente, el modelo basado en SVM debería producir un hiperplano que separe completamente los datos del universo estudiado en dos categorías. Sin embargo, una separación perfecta no siempre es posible y, si lo es, el resultado del modelo no puede ser generalizado para otros datos. Esto se conoce como sobreajuste (overfitting).

Con el fin de permitir cierta flexibilidad, las SVM manejan un parámetro C que controla la compensación entre errores de formación y los márgenes rígidos, creando así un margen blando (soft margin) que permita algunos errores en la clasificación a la vez que los penaliza.



C. Funciones Kernel

Las SVM pueden manejar eficientemente datos no lineales mediante el uso de funciones kernel. Estas funciones permiten que las SVM realicen transformaciones no lineales en el espacio de características original, lo que puede ser útil para separar clases que no son linealmente separables..

La formulación general de un kernel *K* es:

$$K(x, y) = \phi(x) * \phi(y)$$

donde x e y son dos vectores de características y ϕ es una función que mapea los vectores originales a un espacio de características de mayor dimensión.

Algunos ejemplos comunes de kernels incluyen:

- **Kernel lineal:** $K(x, y) = x * y$
- **Kernel polinómico:** $K(x, y) = (x * y + c)^d$, donde c es un término constante y d es el grado del polinomio.
- **Kernel radial (RBF):** $K(x, y) = e^{\frac{\|x-y\|^2}{2\sigma^2}}$, donde σ es un parámetro que controla la amplitud de la función gaussiana.
- **Kernel Sigmoid:** $K(x, y) = \tanh(\alpha * x * y + c)$ α es un parámetro que controla la pendiente de la tangente, y c es un término constante.

La elección del kernel en un SVM o en otros algoritmos de aprendizaje automático es crucial y puede tener un gran impacto en el rendimiento del modelo. La selección del kernel adecuado depende de la naturaleza del problema y de las características de los datos.

D. DATSET DIABETES.CSV

Este conjunto de datos proviene originalmente del Instituto Nacional de Diabetes y Enfermedades Digestivas y Renales. Su objetivo es predecir, basándose en medidas diagnósticas, si una paciente tiene diabetes.

El conjunto de datos incluye información sobre pacientes femeninas de al menos 21 años de herencia india Pima. Las variables proporcionadas son:

- Embarazos: Número de veces que la paciente ha estado embarazada.
- Glucosa: Concentración de glucosa plasmática después de 2 horas en una prueba de tolerancia oral a la glucosa.
- Presión Sanguínea: Presión arterial diastólica (mm Hg).
- Grosor de la Piel: Grosor del pliegue cutáneo del tríceps (mm).
- Insulina: Nivel de insulina en suero después de 2 horas (mu U/ml).
- IMC (Índice de Masa Corporal): Índice de masa corporal (peso en kg / (altura en m)^2).
- Función de Pedigrí de la Diabetes: Valor de la función de pedigrí relacionada con la diabetes.
- Edad: Edad de la paciente en años.
- Resultado: Variable de clase (0 o 1) indicando la presencia o ausencia de diabetes.

Restricciones:

El conjunto de datos se enfoca en pacientes femeninas mayores de 21 años de herencia india Pima, y se seleccionaron siguiendo ciertas restricciones en la base de datos más grande.

II. DESARROLLO

A. PREPROCESAMIENTO

Los datos extraídos fueron sometidos a un proceso de escalado debido a la sensibilidad de las Máquinas de Soporte Vectorial (SVM) a la escala de las características. Dado que las SVM buscan encontrar el hiperplano óptimo que separa las clases, el escalado es esencial para evitar que características con diferentes escalas dominen la determinación del hiperplano.

B. KERNELS:

El conjunto de datos de diabetes se evaluó utilizando diversos kernels en una primera instancia:

- Linear (Lineal)
- Radial (RBF)
- Polynomial (Polinómico)
- Sigmoid (Sigmoide)

Estos kernels se aplicaron con parámetros predeterminados, es decir, sin modificaciones en términos de regularización, gamma o degree.

C. GRID SEARCH Y RANDOMIZED SEARCH

Con el objetivo de mejorar la calidad de los resultados, se implementaron técnicas de búsqueda de hiperparámetros:

1. Grid Search (Búsqueda en Cuadrícula):
   Se realizó una búsqueda exhaustiva de todas las combinaciones posibles de hiperparámetros en una cuadrícula predefinida.
   - Parámetro de regularización: 0.1, 1, 5, 10, 20, 50, 70, 100.
   - Gamma (para kernels radial y sigmoide): 1, 0.2, 0.1, 0.01.
   - Degree (para kernel polinómico): 2, 3, 4.
   - Validación cruzada: Se empleó un valor de 10 para obtener estimaciones estables del rendimiento del modelo.
2. Randomized Search (Búsqueda Aleatoria):
   Se llevó a cabo una búsqueda aleatoria en un espacio de hiperparámetros especificado. Se utilizaron los mismos parámetros que en la búsqueda en cuadrícula para establecer un punto de comparación.

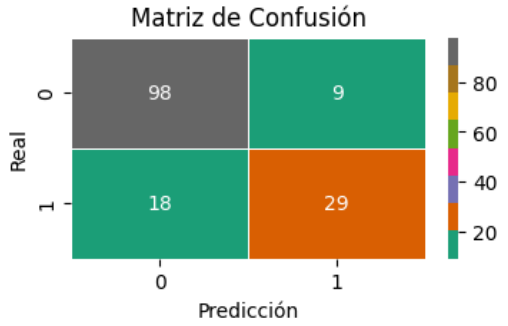
   Estos procesos buscan encontrar los mejores kernels e hiperparámetros para optimizar el rendimiento de las SVM en la predicción de la diabetes.

III. RESULTADOS

Dividiendo los datos en un 20% de prueba y 80% de entrenamiento, y sin modificar ningún hiper parámetro obtenemos el resultado de:

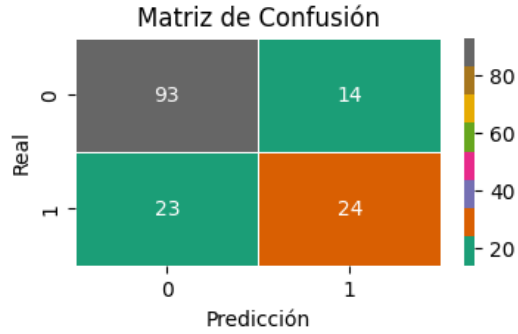
A. Kernel: linear

Precisión: 82.46 %



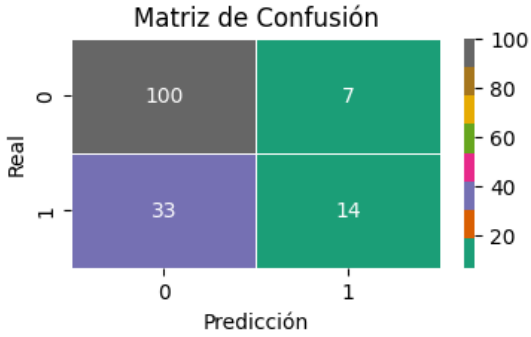
B. Kernel: radial (rbf)

Precisión: 75.97 %



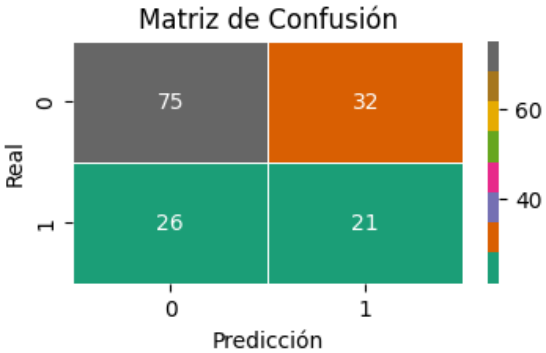
C. Kernel: Polinómico (poly)

Precisión: 74.02 %



D. Kernel: Sigmoid

Precisión: 62.33 %



Obtenemos un buen resultado con el kernel linear, donde podemos interpretar la matriz de confusión como:

- Hay 98 instancias que fueron correctamente clasificadas como negativas True Negative (TN).
- Hay 9 instancias que fueron incorrectamente clasificadas como positivas False Positive (FP).
- Hay 18 instancias que fueron incorrectamente clasificadas como negativas False Negative (FN).
- Hay 29 instancias que fueron correctamente clasificadas como positivas True Positive (TP).

Mientras el kernel de sigmoid obtuvo malos resultados, pero es momento de comprobar usando Grid Search y Randomized Search y encontrado los mejores hiperparametros donde al aplicar Grid Search obtenemos:

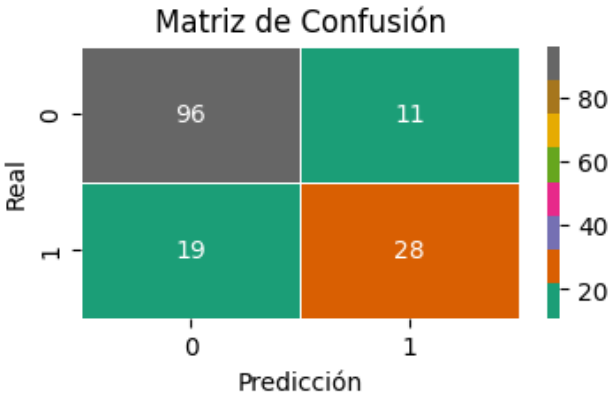
- C: 10,
- gamma: 0.01,
- kernel: sigmoid
- precisión: 76.05

Mientras que Randomized Search obtenemos:

- C: 21.98
- Gamma: 0.014
- Kernel: sigmoid
- Precisión: 76.73

Donde los dos métodos coinciden en el método, si volvemos al modelo de SVM y probamos con estos nuevos valores obtenemos:

- Kernel: sigmoid
- Precisión: 80.51 %



Con los resultados obtenidos, observamos una clara mejora en el kernel sigmoid, donde ajustando a mayor precisión los hiper parámetros, alcanzamos una precisión del 80.51% a diferencia del 62.23% obtenido con anterioridad. Comparando con el kernel de linear el cual nos dio 82.46 % podríamos pensar que tiene un sobre ajuste, pero debemos recordar que La precisión por sí sola no siempre es indicativa del rendimiento del modelo,

y hay varias razones por las cuales podrías observar diferencias en los resultados:

- Espacio de Hiperparámetros: La búsqueda de hiperparámetros en GridSearchCV y RandomizedSearchCV puede haber explorado un espacio más amplio de hiperparámetros, incluyendo configuraciones que podrían no ser las óptimas para el dataset
- Sensibilidad del Modelo: Los modelos con diferentes kernels (lineal, rbf, sigmoid, etc.) pueden comportarse de manera diferente según la naturaleza de los datos. Un kernel lineal podría funcionar bien si la relación entre las características y la variable objetivo es lineal, mientras que otros kernels pueden ser más adecuados para patrones no lineales.
- Número de Iteraciones: La cantidad de iteraciones en GridSearchCV y RandomizedSearchCV puede afectar los resultados. A veces, una mayor cantidad de iteraciones puede llevar a una búsqueda más exhaustiva de hiperparámetros
- Sensibilidad del Modelo a los Hiperparámetros: La sensibilidad del modelo a los hiperparámetros puede variar según la naturaleza de los datos. Algunos modelos pueden ser más sensibles a ciertos hiperparámetros que otros.

Ahora realicemos el experimento haciendo la validación cruzada para evaluar el rendimiento de los kernel, así obtenemos una estimación más robusta y fiable. Como resultados obtuvimos:

A. Kernel Lineal:
- Precisión Promedio: 75.72%
- Se observa un rendimiento relativamente constante en cada pliegue, indicando que el kernel lineal es razonablemente consistente en la predicción del conjunto de entrenamiento.

B. Kernel RBF:
- Precisión Promedio: 70.04%
- Se observa una variabilidad más marcada en las precisiones de los pliegues, lo que sugiere que el kernel radial (RBF) puede tener más sensibilidad a la variación en los datos.

C. Kernel Polinómico:
- Precisión Promedio: 69.07%
- Similar al kernel RBF, el kernel polinómico muestra cierta variabilidad en las precisiones, lo que sugiere que puede no adaptarse de manera uniforme a diferentes particiones del conjunto de entrenamiento.

Kernel Sigmoid:

- Precisión Promedio: 75.40%
- Muestra una precisión similar al kernel lineal, pero con cierta variabilidad. El kernel sigmoidal parece estar funcionando de manera consistente en diferentes particiones del conjunto de entrenamiento.

En general, el kernel lineal y el kernel sigmoidal muestran precisiones promedio más altas y una consistencia relativamente buena en la validación cruzada. Sin embargo, es importante tener en cuenta que estos resultados pueden variar según la elección específica de particiones de datos.

IV. Observaciones y Conclusiones

Algunas recomendaciones para mejorar la precisión de los resultados son:

1. Explorar más Hiperparámetros:
   Experimenta con una gama más amplia de valores para los hiperparámetros, como el parámetro de regularización (C), gamma en kernels RBF y sigmoidal, y el grado del kernel polinómico.
2. Kernel y Función de Pérdida Alternativos
   Se utilizo los kernels más comunes, sin embargo, existen otros que, dependiendo de la naturaleza de los datos, pueden dar mejores resultados
3. Optimización de Características:
   Realiza un análisis más detallado de la importancia de las características. Es posible que algunas características tengan un impacto significativo en la predicción, mientras que otras pueden no contribuir tanto. Eliminar o agregar características selectivamente puede mejorar el rendimiento.
4. Aumento de Datos:
   Si el tamaño del conjunto de datos lo permite, considera técnicas de aumento de datos para generar variaciones adicionales de las instancias existentes.

En general SVM es un clasificador robusto, es interesante la idea de encontrar un hiperplano óptimo para la separación entre las clases, además, al tener los diferentes kernels nos permite manejar aplicarlos para las diferentes naturalezas de los datos, en este caso el lineal y sigmoidal fueron muy efectivos, además tener la capacidad de ajustar los hiper parámetros permite tener una mayor personalización y opciones para mejorar la precisión

V. Bibliografia

[1] J. Amat Rodrigo. “Máquinas de Vector Soporte (Support Vector Machines, SVMs)”. Ciencia de Datos,net. Accedido el 13 de diciembre de 2023. [En línea]. Disponible:https://cienciadedatos.net/documentos/34_maquinas_de_vector_soporte_support_vector_machines

[2] C. Chique Rodríguez. “Máquina de Soporte Vectorial (SVM)”. Medium. Accedido el 13 de diciembre de 2023. [En línea]. Disponible: https://medium.com/@csarchiquerodriguez/maquina-de-soporte-vectorial-svm-92e9f1b1b1ac

[3] M. Sotaquirá. “Las Máquinas de Soporte Vectorial: una explicación completa”. Codificando Bits. Accedido el 13 de diciembre de 2023. [En línea]. Disponible: https://www.codificandobits.com/blog/maquinas-de-soporte-vectorial/