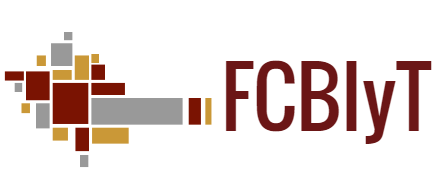
****

**UNIVERSIDAD AUTÓNOMA DE TLAXCALA**

**Facultad de Ciencias Básicas Ingeniería y Tecnología**

**INGENIERÍA EN COMPUTACIÓN**

**ACTIVIDAD:**

Uso de funciones de máquinas de soporte vectorial en Matlab para la clasificación de datos

**PRESENTA:**

**DOCENTE:**

M.C. Jorge Arturo Flores López

**ALUMNO:**

Christopher Rojano Jimenez

**SEMESTRE Y GRUPO:**

8 “B”

**Correo:**

20191414@uatx.mx

Apizaco, Tlaxcala, junio 2023

**Introducción**

En los últimos años, el crecimiento exponencial de los datos ha planteado nuevos desafíos en el campo de la clasificación y análisis de información. La capacidad de extraer conocimientos útiles de grandes volúmenes de datos se ha convertido en una necesidad para diversas áreas, como la medicina, la economía, la biología y la ingeniería, entre otras. En este contexto, las máquinas de soporte vectorial (SVM, por sus siglas en inglés) han demostrado ser una herramienta efectiva para la clasificación y análisis de datos en problemas complejos.

Las SVM son un tipo de algoritmo de aprendizaje supervisado que se utiliza para la clasificación y regresión. Su enfoque se basa en la idea de encontrar el hiperplano óptimo que mejor separa las diferentes clases de datos. El objetivo principal de las SVM es maximizar el margen entre los puntos de datos de diferentes clases, lo que permite una clasificación precisa incluso en situaciones en las que los datos son no linealmente separables.

En este proyecto, se explorará el uso de dos funciones de máquinas de soporte vectorial en Matlab para la clasificación de dos dataset. Matlab es un entorno de programación y desarrollo que ofrece una amplia gama de herramientas para el análisis de datos y la implementación de algoritmos de aprendizaje automático. La elección de Matlab se debe a su capacidad para manipular y visualizar datos de manera eficiente, así como su amplia disponibilidad de funciones y bibliotecas para la implementación de SVM.

**1. Metodología**

Para el desarrollo de este proyecto se hizo uso de dos dataset extraídos del repositorio LIBSVM el cual contiene muchos dataset de clasificación, regresión, etiquetas múltiples y cadenas almacenados en formato LIBSVM [CHIH2011], dicho formato permite el uso y lectura desde Matlab. Los dataset que fueron utilizados son:

**fourclass**

Este dataset fue utilizado en el proyecto “*Building projectable classifiers of arbitrary complexity*” el cual presenta un método que construye un clasificador hasta una complejidad arbitraria mientras presenta precisión de generalización. El dataset se compone de 862 datos y dos clases. [KLEINBERG1996].

**svmguide1**

Este dataset fue extraído del proyecto “*A practical guide to support vector classification*” el cual tiene como propósito dar a los novatos de SVM una receta para obtener rápidamente resultados aceptables. Además el dataset fue proveído, para uso del proyecto antes mencionado, por cortesía de Jan Conrad de la Universidad de Uppsala, Suecia y contiene datos de astro partículas. El dataset se compone de 3089 datos y dos clases. [HSU2003].

**2. Implementación**

Para el desarrollo de este proyecto se hizo uso de dos funciones de Matlab para VSM, las cuales son:

**fitcsvm**

La función fitcsvm en Matlab es una función que se utiliza para ajustar un modelo de máquina de soporte vectorial (SVM) a un conjunto de datos de entrenamiento. Esta función se encuentra en la toolbox de estadísticas y aprendizaje automático de Matlab y ofrece una implementación eficiente y flexible de SVM para la clasificación de datos.

La sintaxis básica de la función fitcsvm es la siguiente:

|  |  |  |
| --- | --- | --- |
|  | Imagen que contiene Texto  Descripción generada automáticamente |  |
|  | Figura 2.1. Sintaxis de la función fitcsvm |  |

Donde:

* X es una matriz de tamaño n x p que contiene los datos de entrenamiento, donde n es el número de observaciones y p es el número de características.
* Y es un vector de tamaño n x 1 que contiene las etiquetas de clase correspondientes a los datos de entrenamiento.

La función fitcsvm utiliza un enfoque de aprendizaje supervisado para ajustar el modelo SVM a los datos de entrenamiento. El algoritmo busca encontrar el hiperplano óptimo que mejor separa las diferentes clases de datos, maximizando el margen entre los puntos de datos de diferentes clases. Dependiendo de los parámetros especificados, fitcsvm puede ajustar tanto SVM lineales como no lineales mediante el uso de kernel.

La función fitcsvm proporciona una variedad de opciones y parámetros que permiten personalizar y ajustar el modelo SVM según las necesidades del usuario. Algunos de los parámetros más comunes incluyen la elección del tipo de kernel (lineal, polinomial, RBF, entre otros), el valor del parámetro de regularización C, los pesos de clase, entre otros. Estos parámetros pueden influir en la precisión y el rendimiento del modelo SVM.

**fitcecoc**

La función fitcecoc en Matlab es una función que se utiliza para ajustar un modelo de clasificación multiclase utilizando la estrategia de clasificación "uno contra todos" (one-vs-all) en Matlab. Esta función se encuentra en la toolbox de estadísticas y aprendizaje automático de Matlab y ofrece una implementación eficiente y fácil de usar para la clasificación multiclase utilizando SVM u otros clasificadores binarios.

La sintaxis básica de la función fitcecoc es la siguiente:

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | Figura 2.1. Sintaxis de la función fitcecoc |  |

Donde:

* X es una matriz de tamaño n x p que contiene los datos de entrenamiento, donde n es el número de observaciones y p es el número de características.
* Y es un vector de tamaño n x 1 que contiene las etiquetas de clase correspondientes a los datos de entrenamiento.

La función fitcecoc utiliza un enfoque de aprendizaje supervisado para ajustar un modelo multiclase a los datos de entrenamiento utilizando SVM o cualquier otro clasificador binario especificado mediante el parámetro 'Learners'. En la estrategia "uno contra todos", se ajusta un clasificador binario para cada clase, donde se considera una clase como positiva y todas las demás clases como negativas. Luego, se combina la salida de los clasificadores binarios para obtener una decisión final de clasificación multiclase.

La función fitcecoc proporciona una variedad de opciones y parámetros que permiten personalizar y ajustar el modelo multiclase según las necesidades del usuario. Algunos de los parámetros más comunes incluyen la elección del clasificador binario subyacente ('Learners'), la elección del tipo de kernel para SVM, los pesos de clase, entre otros. Estos parámetros pueden influir en la precisión y el rendimiento del modelo multiclase.

**3. Código**

Para la implementación de la primera función se realizó el siguiente código:

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | Figura 3.1. Lectura del dataset y uso de la función fitcsvm |  |

En la figura 3.1 podemos observar como se lee el dataset, del cual se obtiene las etiquetas que pertenecen a cada clase de lo puntos y los puntos en si mismos. Posteriormente se usa la función fitcsvm con la información del dataset y se realizan las predicciones para otros puntos a clasificar.

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | Figura 3.2. Obtención y graficado de los vectores de soporte |  |

Posteriormente, en la figura 3.2 se muestra como son graficados los puntos con la clase predicha en la figura 3.1, después se obtienen los vectores de soporte que fueron identificados y de igual manera se grafican.

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | Figura 3.3. Graficado de la región de decisión y pasos finales |  |

Finalmente, podemos observar en la figura 3.3 como se grafica la región de decisión para el conjunto y se agregan las descripciones a la gráfica final.

Para el uso de la función fitcecoc se realizó el siguiente código:

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | Figura 3.4. Lectura del dataset y clasificación con fitcecoc. |  |

En la figura 3.4 se observa como se lee el dataset y se extraen las características y las etiquetas de este, después se hace uso de la función fitcecoc la cual ajusta el modelo de los datos para realizar la clasificación

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | Figura 3.5. Calculo de los vectores de soporte. |  |

En la figura 3.5 se muestra como se obtienen los vectores de soporte a partir del modelo previamente ajustado.

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | Figura 3.6. Graficado de los puntos y vectores de soporte. |  |

Finalmente, en la figura 3.6 se muestra como son graficados tanto los vectores de soporte como los puntos ya clasificados.

**4. Resultados**

Para el dataset fourclass se obtuvieron los resultados de la imagen

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | a. fitcsvm |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | b. fitcecoc |  |
|  |  |  |
|  | Figura 4.1. Resultados para el dataset fourclass. |  |

Como podemos observar en la figura 4.1, ambas funciones clasifican de manera completamente similar entre ellas, lo que da una idea de cuan precisas son.

Para el dataset de svmguide1 se obtuvieron los siguientes resultados:

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | a. fitcsvm |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | b. fitcecoc |  |
|  |  |  |
|  | Figura 4.1. Resultados para el dataset fourclass. |  |

Para este segundo dataset se obtuvo lo mismo, ambas funciones clasificaron de manera similar y efectiva los datos.

En conclusión, el uso de las máquinas de soporte vectorial (SVM) a través de las funciones fitcecoc y fitcsvm en Matlab ofrece una poderosa herramienta para la clasificación de datos en problemas tanto binarios como multiclase.

Las SVM son conocidas por su capacidad para manejar datos no linealmente separables y su capacidad para generalizar bien en conjuntos de datos de alta dimensionalidad. Mediante la optimización del margen entre diferentes clases, las SVM pueden lograr un buen rendimiento de clasificación al encontrar la mejor separación posible entre las clases.

La función fitcsvm permite ajustar modelos SVM tanto lineales como no lineales mediante el uso de kernel, brindando flexibilidad para abordar diferentes tipos de problemas de clasificación. Con esta función, se pueden explorar diferentes configuraciones y ajustar los parámetros del modelo, como el tipo de kernel, el parámetro de regularización y los pesos de clase, para obtener el mejor rendimiento de clasificación.

Por otro lado, la función fitcecoc facilita la clasificación utilizando la estrategia "uno contra todos". Esta estrategia ajusta un clasificador binario para cada clase y combina sus resultados para obtener una clasificación final multiclase. Esto ofrece una forma eficiente y efectiva de abordar problemas de clasificación con múltiples clases utilizando SVM u otros clasificadores binarios especificados.

En conjunto, el uso de las funciones fitcecoc y fitcsvm en Matlab proporciona una implementación eficiente y fácil de usar de las SVM para la clasificación de datos. Estas funciones permiten la exploración de diferentes configuraciones y ajustes de parámetros, así como el análisis y la evaluación del rendimiento de los modelos resultantes.

Sin embargo, es importante tener en cuenta que el rendimiento de las SVM y la elección de los parámetros dependen en gran medida de la naturaleza y las características de los datos. Por lo tanto, es esencial realizar un análisis exhaustivo de los datos, incluido el preprocesamiento adecuado, para obtener resultados óptimos.

**5. Referencias**

[CHIH2011] Chih-Chung Chang and Chih-Jen Lin, LIBSVM: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2:27:1--27:27, 2011. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.

[KLEINBERG1996] Ho, T.K., & Kleinberg, E.M. (1996). Building projectable classifiers of arbitrary complexity. Proceedings of 13th International Conference on Pattern Recognition, 2, 880-885 vol.2.

[HSU2003] Hsu, C. W., Chang, C. C., & Lin, C. J. (2003). A practical guide to support vector classification.