TALLER DE PROCESAMIENTO DE SEÑALES

2do Cuatrimestre 2026 - Trabajo Práctico Nº 5 - SVM

IMPORTANTE: En este ejercicio solamente se podrá importar numpy, matplotlib.pyplot y el solver `solve_qp` de qpsolvers (el resto deberá ser implementación propia).

El gerente de producción de una fábrica de circuitos integrados desea predecir si un determinado integrado pasará el control de calidad. Los archivos `ControlDeCalidad.txt` y `ControlDeCalidad2.txt` poseen datos de la evaluación de dos pruebas diagnóstico de diferentes integrados, y una tercera columna que indica si pasaron el mencionado control (1 significa pasar la inspección).

(**a**) *Implementación:*

1. Explicar como deben ser confeccionadas las matrices para implementar, utilizando `solve_qp`, el problema dual de SVM.
2. Implementar una clase `SVM`, que resuelva el problema dual. El código debe estar estructurado de la siguiente manera.

```
class SVM:
    # Inicializar atributos y declarar hiperparámetros.
    def __init__(self,kernel,...):

    # Etapa de entrenamiento.
    def fit(self,X,y):

    # Etapa de testeo hard
    def predict(self,X):

    # Etapa de testeo soft
    def predict_soft(self,X):
```

   donde `kernel`(x_1, x_2, γ) es la función que define el kernel (γ es un hiperparámetro). El método `predict_soft` hace referencia a la función $z(x)$ que es utilizada para clasificar como $z(x) \lessgtr 0$. 👉: Es recomendable que la función kernel pueda funcionar con conjuntos de datos. Por ejemplo, si $x_1 \in \mathbb{R}^{n \times d_x}$ y $x_2 \in \mathbb{R}^{m \times d_x}$, la función debe devolver una matriz $K \in \mathbb{R}^{n \times m}$.

(**b**) *Base de datos ControlDeCalidad.txt:*

1. Entrenar un clasificador SVM para la tarea de control de calidad utilizando kernel lineal. Indicar el *accuracy* de entrenamiento y el *margen unilateral*.
2. Graficar la frontera de decisión aprendida sobre un `scatter` de los datos utilizando `contour` o `contourf`. Destacar los márgenes y los vectores soporte. ¿Por que se habla de curvas de nivel?
3. Repetir los incisos (**b**).1 y (**b**).2 para un kernel polinómico de la forma $k(x_1, x_2) = (\gamma \cdot x_1^T x_2)^3$ con $\gamma = 0.3$.
4. Se puede pensar a la función $\phi(x)$ que define un kernel $k(x_1, x_2) = \phi(x_1)^T \phi(x_2)$ como una transformación que proyecta a las muestras sobre un espacio de mayor dimensión o comlpejidad. ¿Sobre un espacio de qué dimensión se proyectan las muestras implícitamente para un kernel como el del inciso (**b**).3? Asumir que las muestras viven en $\mathbb{R}^2$. 👉: Desarrollar el producto interno de manera que quede expresado como suma distinguiendo los factores correspondientes a cada vector de entrada. Luego encontrar un $\phi(\cdot)$ tal que $\phi(x_1)^T \phi(x_2)$ defina el kernel deseado.

(**c**) *Base de datos ControlDeCalidad2.txt:*

1. Repetir el inciso (**b**).1 para el set de datos `ControlDeCalidad2.txt` ¿Qué es lo que sucede?
2. Repetir los incisos (**b**).1 y (**b**).2 para este nuevo set de datos con un kernel RBF de la forma $k(x_1, x_2) = e^{-\gamma \|x_1 - x_2\|^2}$ con un γ que considere conveniente.