

Nuestro código implementaba el patrón de diseño 'Estrategia' a través de la interfaz 'Algoritmos' en las clases 'KNN' y 'KMeans'. Específicamente, los métodos 'train' y 'estimate' permitían utilizar polimorfismo con estas dos clases. Para esta práctica, hemos aplicado el mismo patrón en las distancias 'Manhattan' y 'Euclidiana', lo que nos permite elegir sin problemas cuál de ellas implementar en nuestro código a través de la interfaz Distancia del método 'calcularDistancia'.

Por último, queremos destacar el método de la plantilla 'Método de Plantilla', que consiste en crear una clase abstracta con métodos que serán implementados por sus clases hijas. Además, incluye un método público que llama a todos los metodos internos para leer tablas permitiendo que, ya sean etiquetadas o no, facilitando así la escalabilidad del sistema.

Adjunto se encuentra la imagen del diagrama de la implementación del patrón de estrategia de algoritmo

