Capítulo 3

Máquinas de Vectores Soporte

3.1. Introducción

Como se ha visto en la parte teórica de esta subsección, las máquinas de vectores soporte son un paradigma de reciente creación en el campo del Reconocimiento de Patrones, basado fundamentalmente en el principio de Minimización del Riesgo Estructural mediante la obtención de separadores óptimos en el sentido de proporcionar márgenes entre clases máximos (se puede demostrar que ambos principios, minimizar el riesgo estructural y maximizar el margen, son equivalentes).

En su versión más sencilla, una MVS (a partir de aquí ésta será la abreviatura empleada para referirnos a las Máquina de Vectores Soporte) proporciona los hiperplanos de separación entre clases con mayor margen, lo que es indudablemente insuficiente cuando las fronteras de separación son no lineales. En estos casos, se puede extender el concepto de MVS de manera que ésta opere no sobre el espacio original de atributos, sino sobre una proyección en un espacio de Hilbert mediante el *kernel* o núcleo. Este espacio posee un número de dimensiones mayor en el que se espera que las clases sean separables linealmente.

Evidentemente, esta aproximación plantea dos dificultades obvias: en primer lugar, elegir una clase de núcleos que efectivamente haga separables las clases; en segundo lugar, seleccionar los parámetros del núcleo que optimizan alguna medida de la calidad del clasificador (por ejemplo, el error cuadrático medio).

3.2. Objetivos

En esta práctica pretendemos que el alumno se familiarice con la obtención de soluciones basadas en MVS tanto para clasificación como para regresión. Ello implica el análisis del problema para determinar el núcleo más apropiado y los

parámetros de este núcleo que optimizan la tasa de aciertos.

3.3. Material

1. El paquete de software WEKA (o Torch/SVMTorch para aquellos a quienes no les importe aprender otro programa para acelerar el cálculo de soluciones).
2. Los ficheros s-train.arff y s-test.arff proporcionados en la plataforma.
3. [Opcional]Los ficheros c- p- v- h- z- train/test.arff
4. Las páginas web:
   - http://bioinformatics.oxfordjournals.org/content/17/4/349.full.pdf
   - http://www.torch.ch/

   Atención, los ficheros originales han sido modificados (eliminando clases poco representadas) para acelerar el cálculo de soluciones.

3.4. Actividades

1. Cargue los ficheros de entrenamiento y validación s-train y s-test en WEKA
2. Emplee el filtro unsupervised.instance.RemoveWithValues para eliminar todas las instancias de las clases menos numerosas. Quédese sólo con las 3 más frecuentes.
3. **Obtención de un clasificador MVS lineal en el espacio de parámetros.** Seleccione la pestaña Classify y el clasificador CVParameterSelection de la carpeta meta. Este clasificador permite realizar una búsqueda en el espacio de parámetros para seleccionar el que produce mejores resultados evaluados mediante validación cruzada. Aunque esta metodología (la validación cruzada) no se verá en profundidad hasta el final del temario, baste decir que se basa en dividir nuestro conjunto de entrenamiento en n bloques de los que se emplean (n-1) para entrenar y el enésimo para evaluar. De esta forma, tendremos evaluadas muchas MVS, cada una obtenida con un valor de c particular, que podremos comparar entre sí. En la ventana emergente, seleccione el clasificador SMO de la carpeta functions. y en el campo relativo a los parámetros, introduzca C val1 val2 val3, donde val1-2-3 son tres números reales (val3 en realidad entero). Ello hará que se prueben val3 valores del parámetro C (ver teoría) equiespaciados entre val1 y val2. Haga en primer lugar exploraciones con valores de val3 pequeños (2 o 3). Cuando tenga una idea aproximada del tiempo empleado en cada exploración, puede ampliar el valor de val3 si lo cree oportuno. Valores orientativos para el inicio pueden ser diversos rangos pequeños

alrededor de 0.001, 0.01, 0.1,1 y 10. Alternativamente, se puede elaborar un script que genere los valores equivalentes de c y emplear SVMTorch en línea de comandos para entrenar el mismo número de MVS.

4. Realizar una búsqueda similar para núcleos polinómicos de grado 2 (variando el parámetro exponent) y comparar resultados. ¿Se trata de un problema separable linealmente?

5. **Obtención de un clasificador MVS lineal en el espacio proyectado mediante un núcleo gaussiano.** Repita los pasos del apartado anterior incluyendo en la ventana de parámetros explorados dos entradas, una para el parámetro c y otra para G (gamma) del núcleo gaussiano o de base radial. Previamente ha debido seleccionar en la carpeta functions el clasificador SMO con la opción *useRBF* TRUE. Para facilitar el procedimiento, comience fijando un rango alrededor del valor óptimo de c obtenido en la sección anterior y val3 correspondiente pequeño.

6. [Opcional] Genere un conjunto de entrenamiento con dos atributos, no separable linealmente. Por ejemplo, utilice dos distribuciones de probabilidad gaussianas para cada clase y simule el problema XOR. Pruebe distintos tipos de núcleo y analice y discuta los resultados ayudándose de gráficas. Puede variar los parámetros de las gaussianas aumentando o disminuyendo las regiones de solape entre clases y estudiar el valor óptimo de los parámetros en función del tamaño de las regiones de confusión.

7. [Opcional] Utilice los otros conjuntos de atributos (c- p- v- h- z-) por separado y conjuntamente y analice y discuta los resultados.