Introducción al Diseño de Experimentos para el Reconocimiento de Patrones

Capítulo 6: Combinación de Clasificadores

Curso de doctorado impartido por
Dr. Quiliano Isaac Moro
Dra. Aranzazu Simón Hurtado
Marzo 2004

Capítulo 6: Combinación de Clasificadores

Introducción

- **Modularización**: se ha dividido la tarea en subtareas y se ha creado un módulo para cada una de ellas. Los resultados luego son integrados.
- **Ensemble**: conjunto de clasificadores redundantes (todos realizando la misma tarea), quizás por diferentes métodos, cuyos resultados luego son integrados
  - Votación.
  - Media.
  - Suma ponderada.
  - ...

Introducción

- La redundancia de los ensembles es para evitar los fallos individuales en la clasificación.
  - Los fallos son debidos a la naturaleza limitada de los datos de entrenamiento.
  - Basado en la variabilidad en la respuesta al mismo problema por parte de distintos mecanismos de clasificación (o el mismo mecanismo con distintos parámetros).
  - Se deben de producir esos fallos de clasificación en distintos datos para distintos clasificadores.

Introducción

- El dilema "bias-variance" (caso de regresión):

  Error cuadrático medio = bias^2 + Variance
  - Bias = medida de la habilidad de generalizar correctamente (una vez entrenado).
  - Variance = medida de la sensibilidad del clasificador respecto a los datos usados en entrenamiento
    - ¿Se obtendrían los mismos resultados si hubiéramos utilizado otros datos de entrenamiento?
- Se busca algún equivalente para el caso de clasificación.
  - Se habla de "bias – spread".

Introducción

- Al combinar clasificadores lo que se busca es disminuir la varianza (dispersión) de las estimaciones ofrecidas por cada clasificador, considerado éste de manera individual, mientras que no se incrementa el sesgo.
  - Es una técnica ideal para las RNA.
    - Dado un conjunto de datos de entrenamiento, hay una multitud de RNA's que pueden ofrecer un error bajo en el dicho conjunto de entrenamiento, pero errores significativamente altos en la prueba con nuevos datos.
    - Al combinar las RNA's se disminuye la dispersión.
- "Principio de incertidumbre":

  $\Delta\text{exactitud} \cdot \Delta\text{sencillez\_modelo} = \text{constante}$
  - En modelos sencillos es más fácil justificar los resultados.

Clasificador base

- Clasificador Base : cada uno de los que constituyen en “ensemble”.
- Tiene que haber variabilidad en los clasificadores base:
  - Ya sea por la tecnología empleada, o
  - Por los datos de entrenamiento usados, o
  - Por los parámetros de aprendizaje empleados.

7

Bagging

- Dado un conjunto de datos de entrenamiento D, se crean N clasificadores usando la técnica Bootstrap.
  - Muestreo aleatorio con reemplazo.
    - Pueden encontrarse datos repetidos y no aparecer otros.
- La salida
  - Originalmente por medio de votación, es decir, aquella clase a la que indiquen la mayoría de los clasificadores.
  - Alternativas:
    - Media.
    - Sumas ponderadas.
    - ...

8

AdaBoosting y ARCing

- **Bagging** → la selección de un ejemplo de entrenamiento es aleatoria (“democrática”).
- **Boosting**: siguiendo la secuencia de creación de clasificadores, procuramos que el nuevo clasificador que se cree preste más atención a aquellos ejemplos en los que los anteriores han producido errores.
  - **ARCING** (Adaptive Reweighting and CombinING)
    - Reweighting → reponderación de las distintas probabilidades de selección de los ejemplos de entrenamiento.

9

AdaBoosting y ARCing

- Se exige un cierto grado mínimo de eficiencia en el clasificador base de partida (siempre mejor que la selección aleatoria).
  - En problemas de clasificación binaria, el clasificador base ha de proporcionar como mínimo un 50% de precisión.
  - En multiclase Precisión ≥ (k-1)/k, siendo k = número de clases.
- Si al crear la secuencia de clasificadores, se llega a uno que no ofrece una exactitud mejor del mínimo exigido, el proceso se para.

10

Algoritmo Boost1

- Históricamente es la primera técnica de boosting.

1. Se obtienen m datos y se crea un clasificador h1 entrenándole con ellos.
2. Mientras no tengamos m patrones en el conjunto C2
   - Tirar una moneda al aire:
     - Si sale cara:
       - Repetir
         - Obtener patrón nuevo y pasarlo por el clasificador h1.
         - Si el patrón ha sido clasificado erróneamente, añadirlo a C2 y pasar al paso 2.
     - Si sale cruz:
       - Repetir.
         - Obtener patrón nuevo y pasarlo por el clasificador h1.
         - Si el patrón ha sido clasificado correctamente, añadirlo a C2 y volver a 2.
3. Crear un clasificador h2 y entrenarlo con C2.
4. Mientras no tengamos m patrones en el conjunto C3
   - Obtener un patrón nuevo y pasarlo por h1 y h2.
   - Si h1 y h2 discrepan, añadimos el patrón a C3.
5. Crear un clasificador h3 y entrenarlo con C3.

- La salida es la del clasificador que ofrezca mayor certeza.

11

Algoritmo AdaBoost

T ={t_i} i=1, ...N conjunto de entrenamiento, cada instancia con una probabilidad de ser escogido para entrenar el clasificador p(i)=1/N

1. En la iteración k se muestrea con reemplazo el conjunto T para obtener datos de entrenamiento T^k, con una distribución de probabilidad p(i).
2. Se crea en clasificador f_k y se prueba con TODOS los datos de entrenamiento disponibles.
   Sea d(i)=1 si el ejemplo i se ha clasificado erróneamente, 0 en caso contrario.
3. Definimos $\varepsilon_k=\sum d(i)p(i)$, y $\beta_k=(1-\varepsilon_k)/\varepsilon_k$
4. Actualizar los pesos para la iteración k+1:
   $p(i) = p(i)\beta_k^{d(i)} / \sum p(i)\beta_k^{d(i)}$

La salida del sistema será la votación ponderada de todos los f_k con pesos log (β_k)

12

Comentarios

- Bagging es más rápido.
- En problemas multiclase a veces es difícil encontrar un clasificador base con precisión mejor que la selección aleatoria.
  - Es posible transformar un problema multiclase en varios binarios.
    - Técnicas de codificación.
- Muchas veces no interesa tener clasificadores base muy exactos.
- Se puede extrapolar inmediatamente la idea al caso de los problemas de regresión.

13