M2.2.2 Modelos Supervisados y No Supervisados

Programa Big Data y Business Intelligence

Enrique Onieva
enrique.onieva@deusto.es
https://twitter.com/EnriqueOnieva
https://www.linkedin.com/in/enriqueonieva/

Deusto
Facultad de Ingeniería
Ingeniaritza Fakultatea

Ensemble Learning

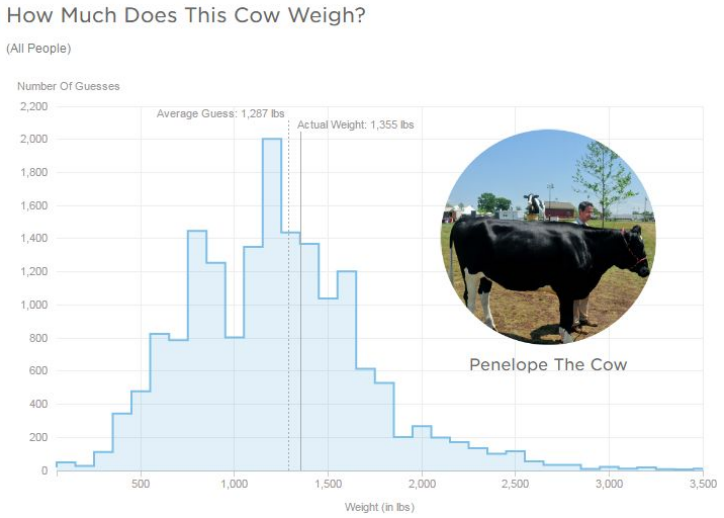
- Motivación e ideas básicas
- Modelos de ensemble
  - Bagging
  - Boosting
  - Stacking
- Random Forest

Motivación

- Imaginemos que quiero una sistema de predicción de peso
- ¿Cuánto pesan estas naranjas?
  - Puedo tantear a toda la clase a ver
    - ¿Quién cocina? ¿Quién come fruta? ¿Quién hace la compra normalmente? ¿Quién compra fruta al peso? ¿A quién le gustan las naranjas? ¿Alguien ha trabajado en una frutería?...
    - Con el fin de escoger a la persona (sistema) de predicción
  - También puedo preguntaros a todos y tomar el promedio
- ¿Ventajas e inconvenientes de cada enfoque?

Motivación

- Buscar un “experto”
  - Necesito mucho tiempo para elegirlo
  - Su predicción puede que no sea buena (igual no está inspirado)
- Preguntar al grupo
  - Es rápido
  - Es sencillo
- Muchas versiones de este “experimento”



http://www.npr.org/sections/money/2015/08/07/429720443/17-205-people-guessed-the-weight-of-a-cow-heres-how-they-did

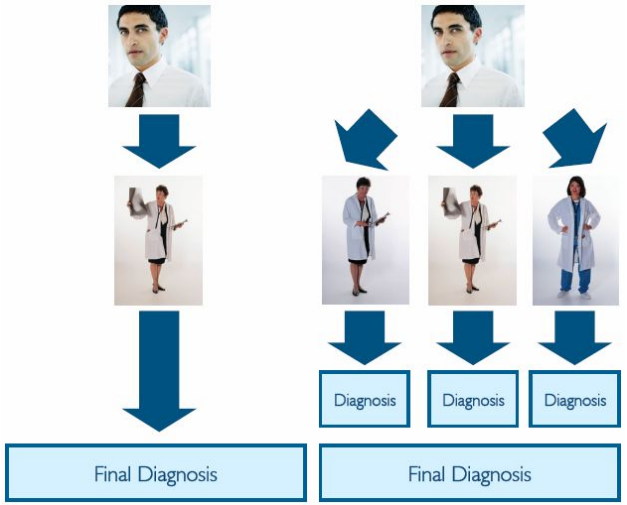
Motivación

- Haciendo números
  - Imaginemos que tenemos 25 modelos de clasificación (binaria)
    - Cada uno tiene una tasa de acierto del 65% (muy malos)
    - Son independientes entre sí
    - Todos votan sobre el resultado a dar
    - La probabilidad de que la mayoría (13 o más) se equivoquen en su predicción es del 6%
- Entre todos, tienen una tasa de acierto del 94% !
  - Es más sencillo encontrar 25 modelos con un 65% de acierto que un único modelo con un 94% de acierto

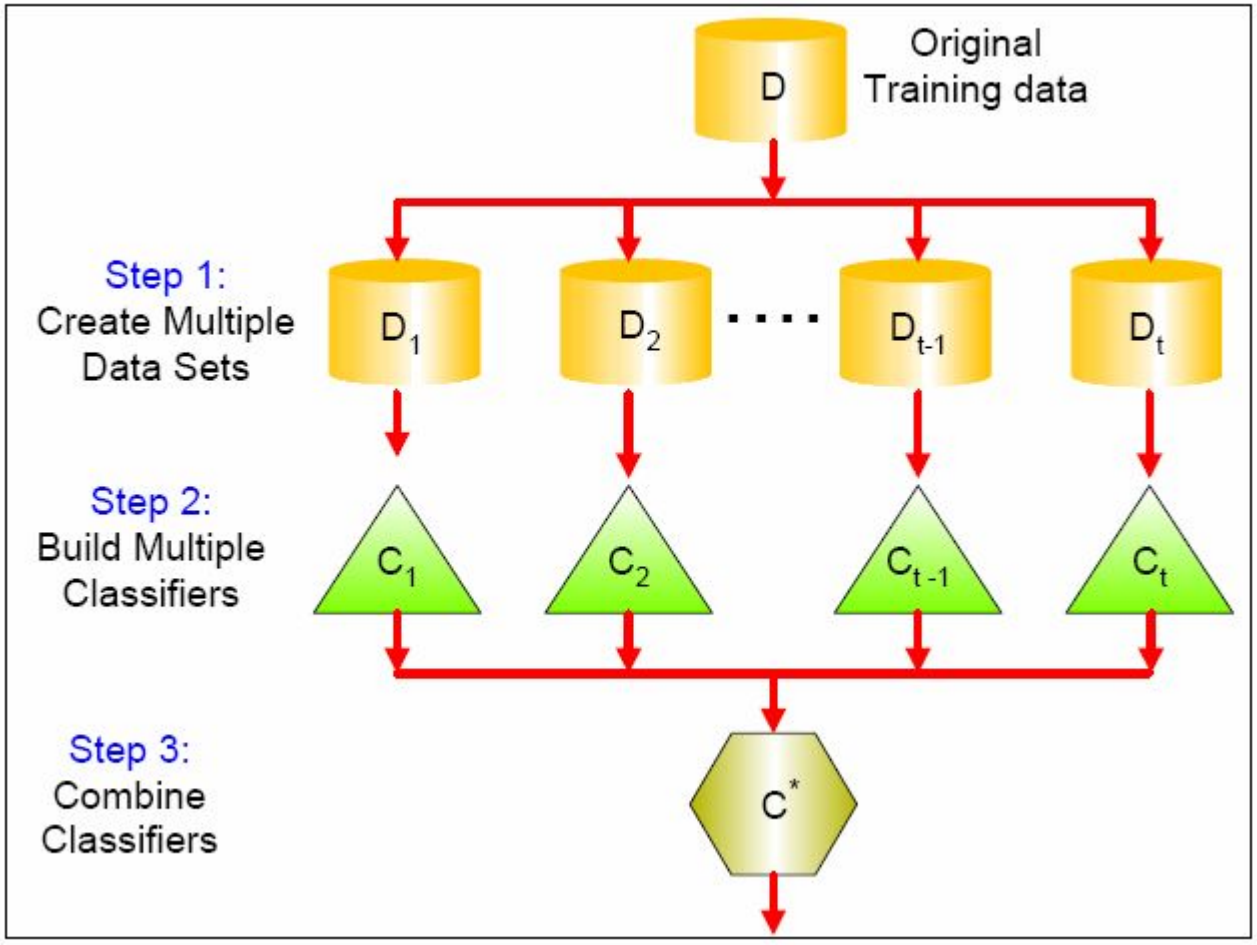
$$\sum_{i=13}^{25} \epsilon^i (1-\epsilon)^{25-i} = 0.06$$

Idea básica

- Idea básica:
  - Construir un conjunto de sistemas diferentes
  - Preguntarle a todos y tomar la decisión
    - Por lo general, mejora la capacidad de predicción.
    - (Muy) Fácilmente paralelizable
  - Inconveniente:
    - El resultado puede ser difícilmente analizable



Idea básica


D
Original
Training data
Step 1:
Create Multiple
Data Sets
D1
D2
Dt-1
Dt
Step 2:
Build Multiple
Classifiers
C1
C2
Ct -1
Ct
Step 3:
Combine
Classifiers
C*

3 enfoques diferentes

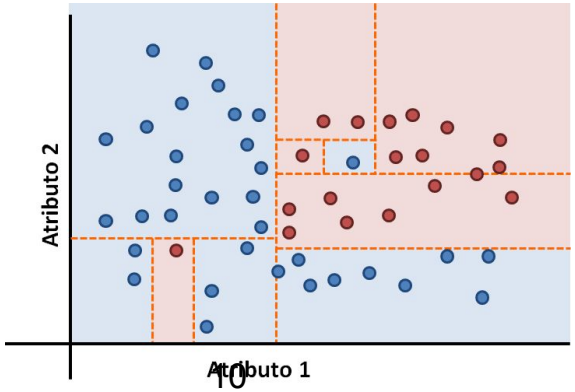
- Bagging (Bootstrap Aggregating)
  - Dividimos el conjunto de datos en porciones
  - Utilizamos cada porción para entrenar un sistema (normalmente del mismo tipo)
- Boosting
  - Construimos diferentes modelos (del mismo tipo)
  - Cada uno intenta “enmendar” los errores de los previos
- Stacking
  - Construimos diferentes modelos (de diferentes tipos)
  - Se construye un “supervisor” que elija la salida a dar a partir de los votos

Bagging

- Analogía: predicción según votación mayoritaria
- Entrenamiento
  - En cada iteración, se toma una porción de los datos (aleatoriamente), y se usa para entrenar un modelo
  - Un modelo se genera en cada iteración
- Predicciones
  - Clasificación: Normalmente por voto mayoritario
  - Regresión: Normalmente por promedio de salidas
- ¿Cuándo creéis que funcionará mejor?
  - ¿Cuándo las muestras que obtengamos de los datos de entrenamiento sean parecidas o distintas?

Bagging

- Mejora los resultados si el método es “inestable”
  - Un pequeño cambio en los datos puede variar mucho la salida del modelo (árboles, redes neuronales,...)
- Extensión:
  - Se puede aleatorizar el método, en lugar de los datos
  - Pesos iniciales en redes neuronales
  - Dar a cada modelo ciertas “columnas” de los datos



Boosting

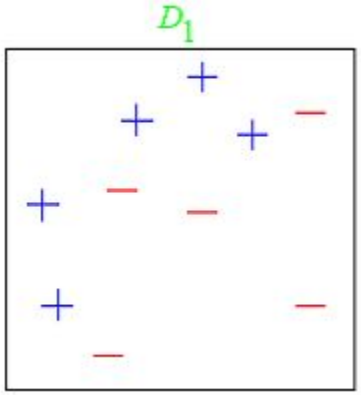
- Analogía: votación donde cada votante sabe el voto de los votantes anteriores
- Funcionamiento
  - Asignamos pesos a los ejemplos
  - Se entrenan k clasificadores
  - Los pesos de los datos donde un clasificador acierta, disminuyen, los datos donde falla, aumentan
  - La salida final es el voto ponderado (en función del acierto)

Boosting

- Ejemplo
  - Tenemos 5 ejemplos {1,2,3,4,5}
  - Inicialmente cada uno tiene un peso de 0.2
    - (probabilidad de ser elegidos)
  - Elegimos datos aleatoriamente {2, 4, 4, 3, 2}
  - Entrenamos un clasificador, lo probamos con los datos
    - Acierta en datos 2, 3 y 5 → reducimos sus pesos
    - Falla en 1 y 4 → aumentamos sus pesos
  - En la siguiente ronda, se vuelven a elegir 5 ejemplos, pero ahora, 1 y 4 se escogerán con más probabilidad
    - El siguiente clasificador se "fijará" más en ellos

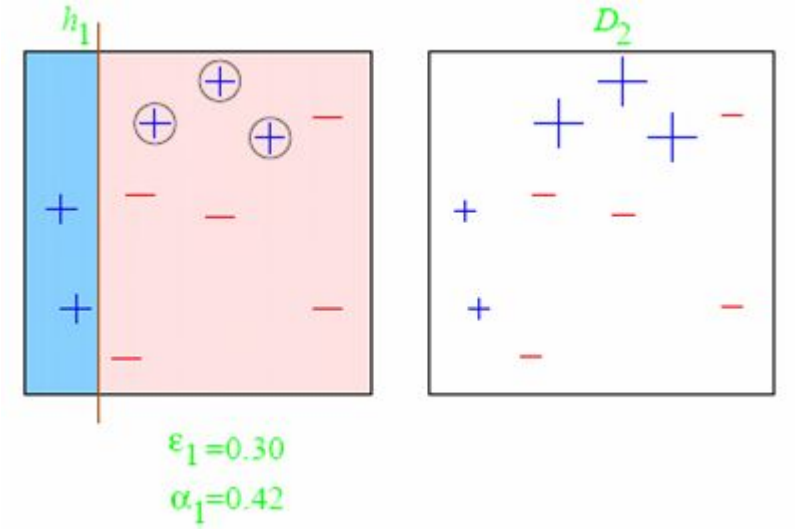
Boosting - Adaboost

- Ejemplo: Datos iniciales



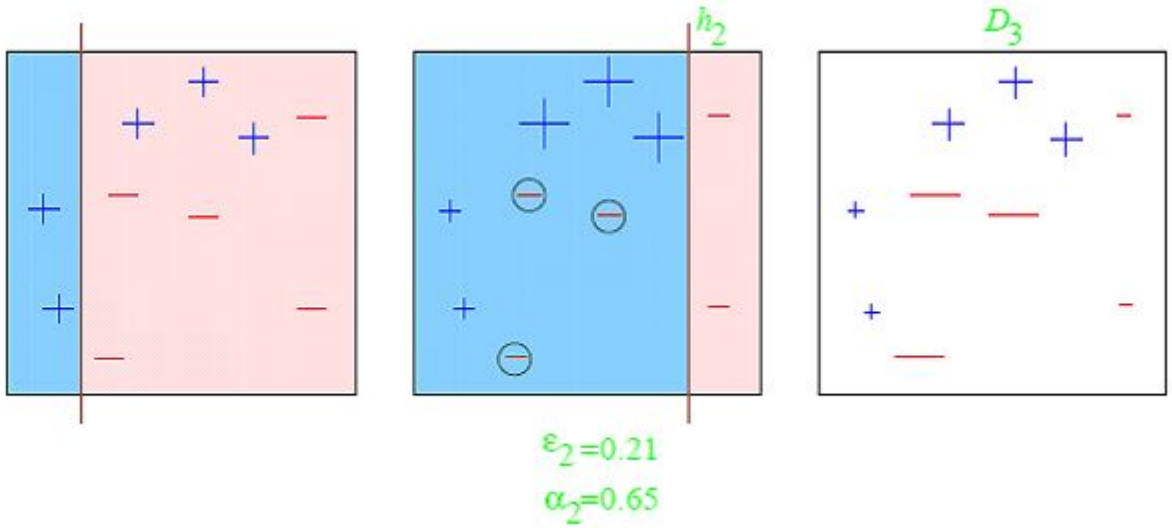
Boosting - Adaboost

- Ejemplo: Primer clasificador



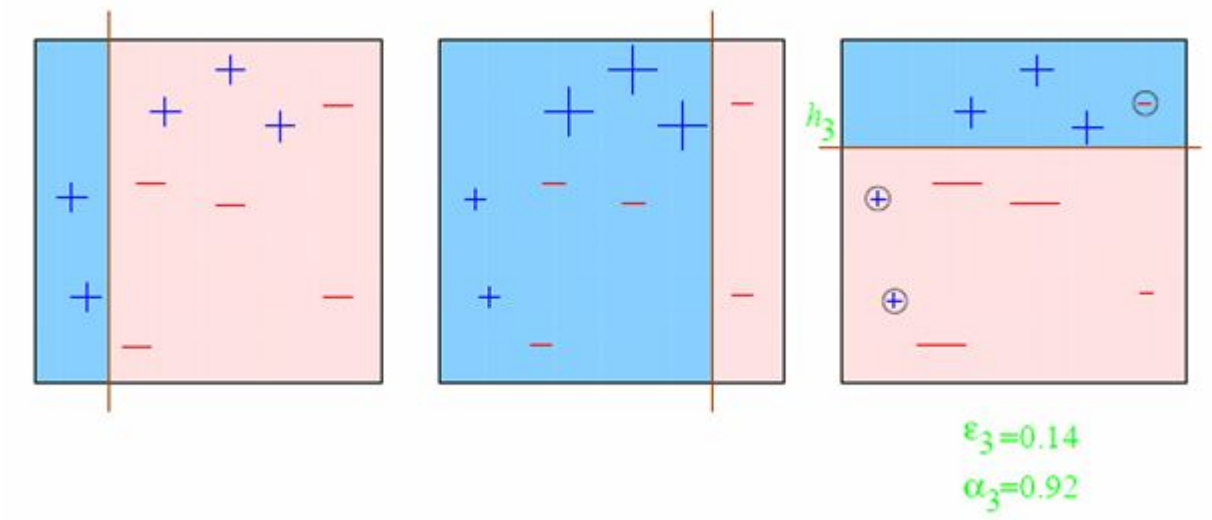
Boosting - Adaboost

- Ejemplo: Segundo clasificador



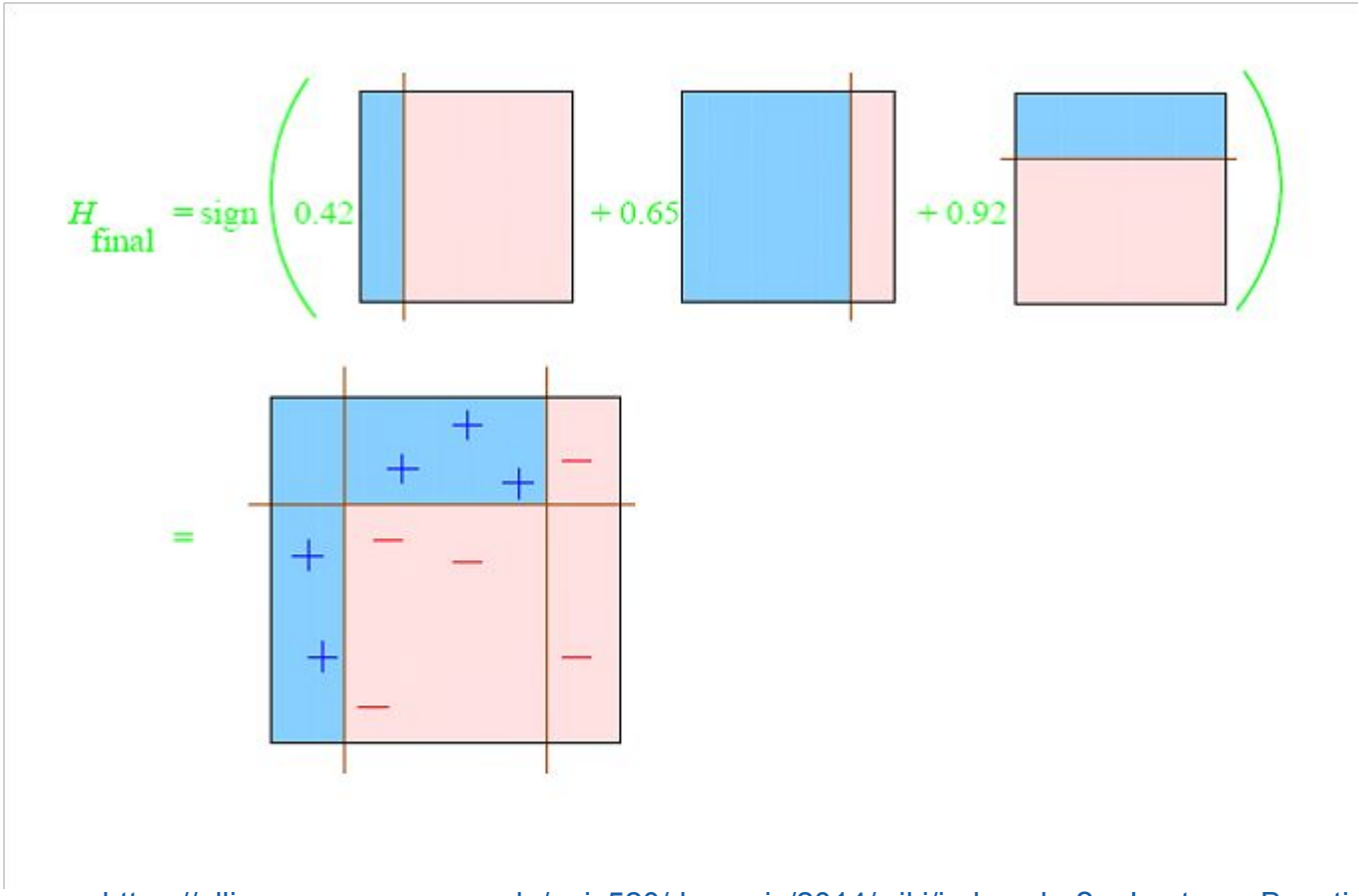
Boosting - Adaboost

- Ejemplo: Tercer clasificador



Boosting - Adaboost

- Ejemplo: Resultado Final



https://alliance.seas.upenn.edu/~cis520/dynamic/2014/wiki/index.php?n=Lectures.Boosting

Resumen

- También utiliza un esquema de votos
- Pondera a los modelos según su calidad
- Iterativo: nuevos modelos se influencian de la calidad de los anteriores
  - Guían al nuevo modelo a ser “experto” en los datos para los que otros fallan
  - Justificación intuitiva: los modelos deben de complementarse los unos a los otros
  - Variantes:
    - Boosting por datos: los pesos influyen en los ejemplos para entrenar
    - Boosting por peso: los pesos son usados por el método de aprendizaje

Random Forest

- Es una variante de algoritmo Bagging basado en árboles
  - Se basa en la combinación de árboles de decisión
  - Cada árbol depende de los valores de datos elegidos aleatoriamente
  - Cada árbol se construye con una porción de todos los atributos disponibles
  - Observar que:
    - Los datos que entrena cada árbol pueden variar
    - Las variables que utiliza cada árbol pueden variar
  - Se evita el overfitting

Enrique Onieva
enrique.onieva@deusto.es
https://twitter.com/EnriqueOnieva
https://www.linkedin.com/in/enriqueonieva/