Electiva Bases de datos

Ensambles

Jimmy Mateo Guerrero[1]

[1]Departamento de Sistemas
Facultad de Ingeniería

Ensambles, 2019

Contenido

Ensambles

- Conjunto de modelos que se usan juntos como un meta modelo.
- Usar conocimiento de distintas fuentes al tomar decisiones.

Ensambles

- Comité de expertos:
  - muchos elementos
  - todos con alto conocimiento
  - todos sobre el mismo tema
  - votan

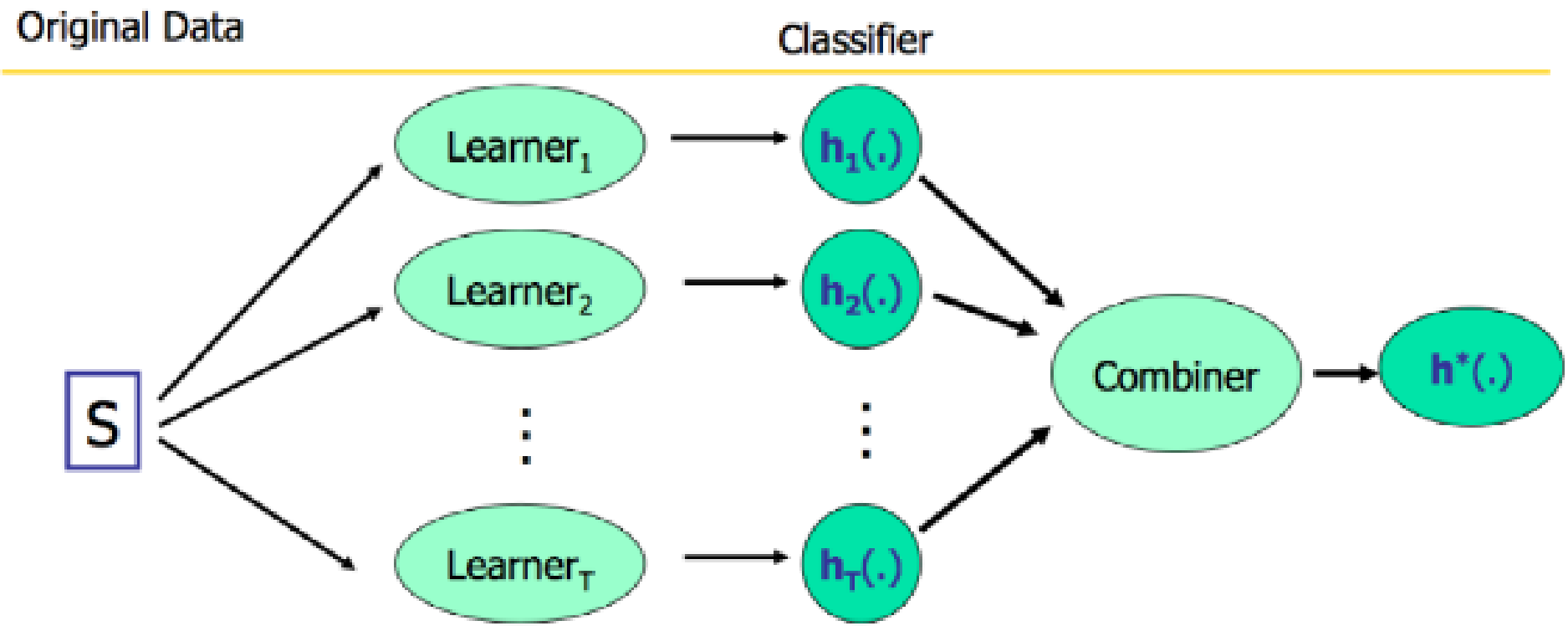
Ensambles

- Dos componentes base
  - Un método para seleccionar o construir los miembros
    - Distintos datasets x distintos modelos x distintas configuraciones
  - Un método para combinar las decisiones
    - Votación simple, votación ponderada, promedio, función específica, selectividad ...

Ensambles

- Ensambles Planos:
  - Muchos expertos, todos buenos.
  - Necesito que sean lo mejor posible individualmente (De lo contrario, usualmente no sirven.).
  - Pero necesito que opinen distinto en algunos casos (Si todos opinan siempre igual... me quedo con uno solo!).

Ensambles


Original Data
Classifier
S
Learner1
Learner2
LearnerT
h1(.)
h2(.)
hT(.)
Combiner
h*(.)

Ensambles

- Un aprendiz se dice inestable si el clasificador que produce sufre cambios importantes ante pequeñas variaciones en los datos de entrenamiento.
  - Inestables: árbol de decisiones, redes neuronales.
  - Estables: La regresión lineal, el vecino más cercano.
- Subsampling es mejor para los alumnos inestables.

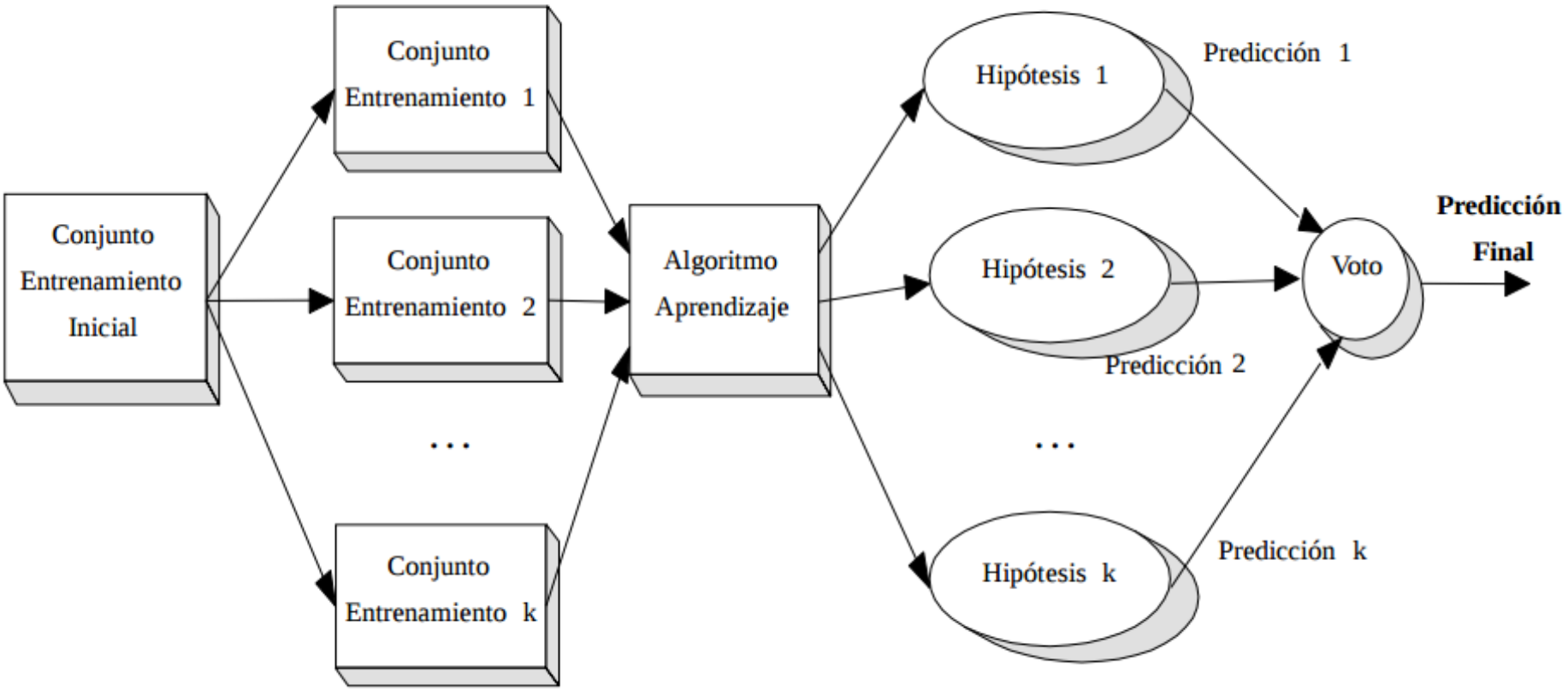
Ensambles

- Existen dos tipos de algoritmos de votación:
  - aquellos que cambian adaptativamente la distribución del conjunto de entrenamiento basado en el desempeño de clasificadores anteriores (boosting).
  - y los que no (como en Bagging).

Bagging

- Bootstrap aggregating (Breiman 96)
- Clasificadores de votos generados por diferentes muestras de bootstrap (réplicas)
- Se generan T muestras de bootstrap B1, B2, ..., BT y se construye un Ci clasificador a partir de cada muestra de bootstrap Bi
- Un último clasificador C * se construye a partir de C1, C2, ..., CT cuya salida es la clase predicha mas frecuente o votada por los clasificadores.

Bagging


Conjunto
Entrenamiento
Inicial
Conjunto
Entrenamiento 1
Conjunto
Entrenamiento 2
Conjunto
Entrenamiento k
Algoritmo
Aprendizaje
Hipótesis 1
Hipótesis 2
Hipótesis k
Predicción 1
Predicción 2
Predicción k
Voto
Predicción
Final

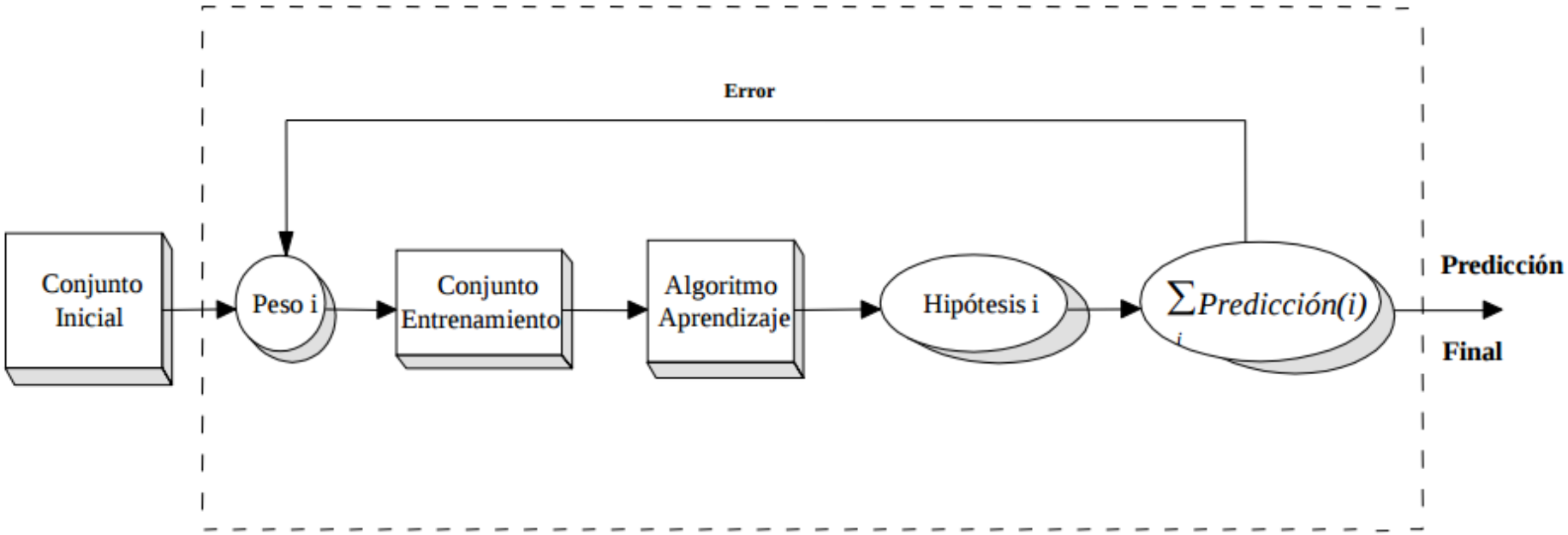
Boosting

- Boosting (Schapire 90), AdaBoost M1 (Freund y Schapire 96)
- Genera los clasificadores secuencialmente, mientras que Bagging puede generarlos en paralelo.
- AdaBoost cambia los pesos de las instancias de entrenamiento proporcionadas como entrada para cada inductor en función de los clasificadores que se construyeron previamente.
- El objetivo es obligar al inductor a minimizar el error esperado sobre diferentes distribuciones de entrada.
- C * = votación ponderada. El peso de cada classfier depende de su rendimiento en el conjunto de entrenamiento utilizado para construir

Boosting

- Boosting: buscar nuevos modelos para las instancias mal clasificadas por los anteriores.
- Fuerza al algoritmo a centrarse en los ejemplos mal clasificados por las hipótesis anteriores.
- Las instancias incorrectas son ponderadas por un factor inversamente proporcional al error en el conjunto de entrenamiento, es decir, 1 / (2Ei). Pequeños errores de entrenamiento, como 0.1 %, harán que los pesos crezcan en varios órdenes de magnitud.

Boosting


Error
Conjunto Inicial
Peso i
Conjunto Entrenamiento
Algoritmo Aprendizaje
Hipótesis i
$\sum_i Predicción(i)$
Predicción
Final

Bibliografía I

Aurélien Géron

Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems.

Publisher: O'Reilly Media, Year: 2017

Jake VanderPlas.

Python Data Science Handbook: Essential Tools for Working with Data

O'Reilly Media, Year: 2016

Brett Lantz.

Machine learning with R

Packt Publishing, Year: 2013