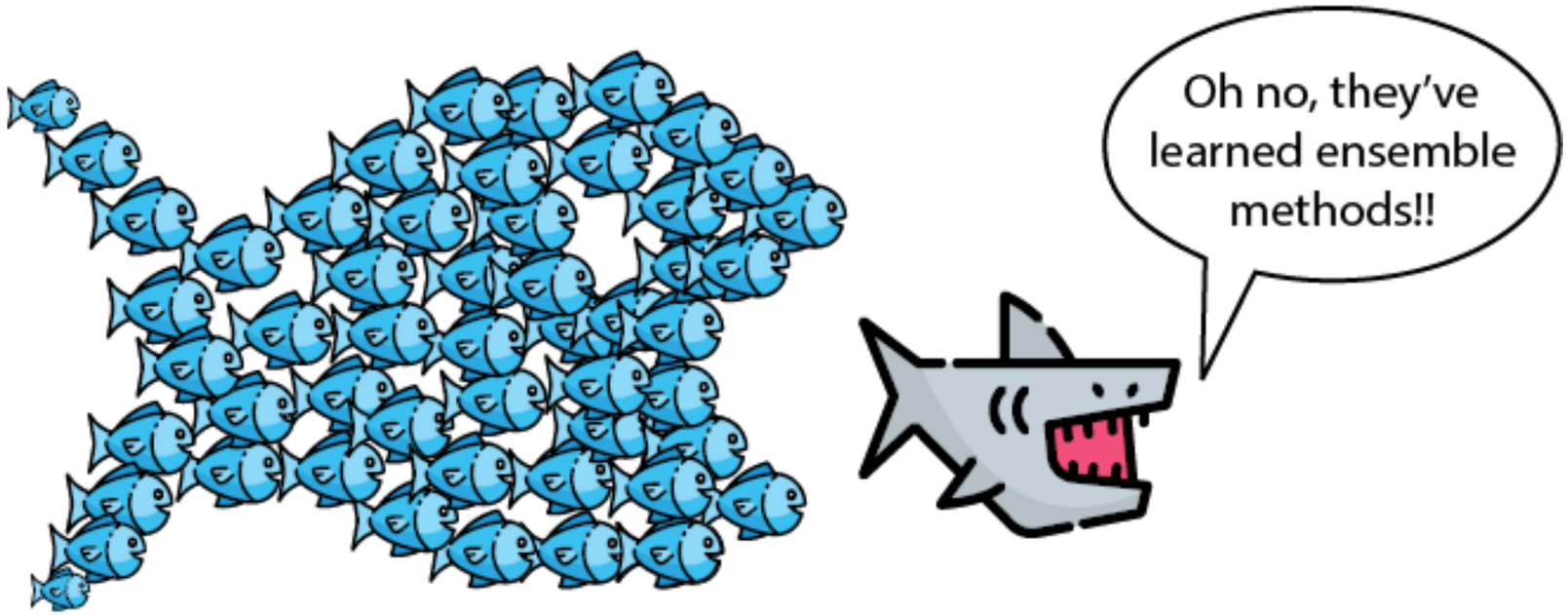
Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

IIC2613 - Inteligencia Artificial

Boosting

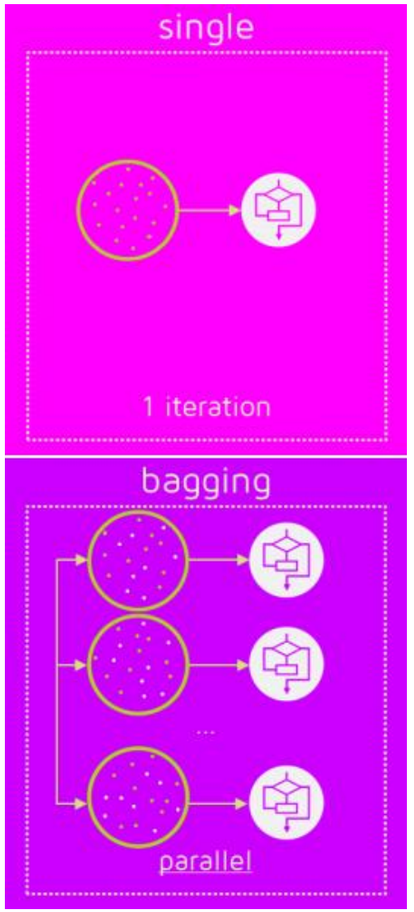
Hans Löbel
Dpto. Ingeniería de Transporte y Logística
Dpto. Ciencia de la Computación

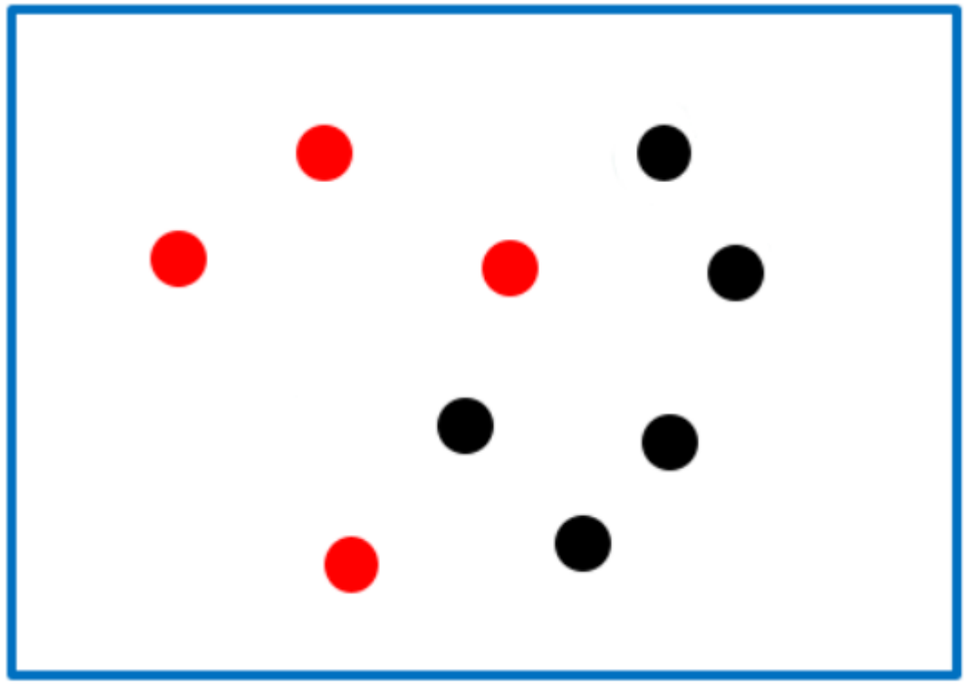
A modo general, métodos de ensamble buscan combinar modelos para generar uno mejor

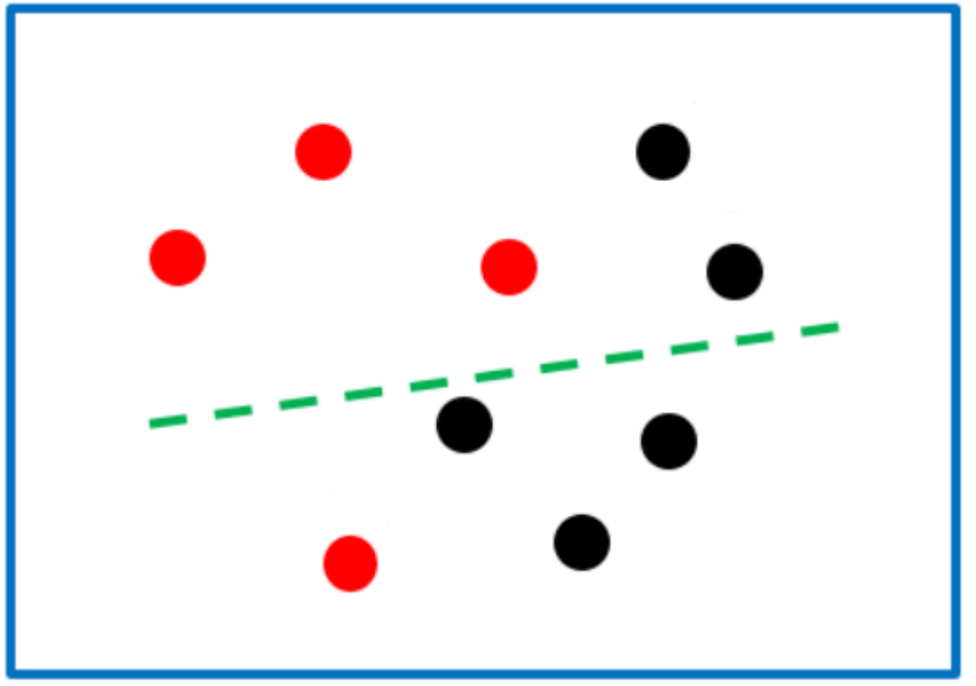


Random Forests es solo uno de muchos tipos posibles de ensamble

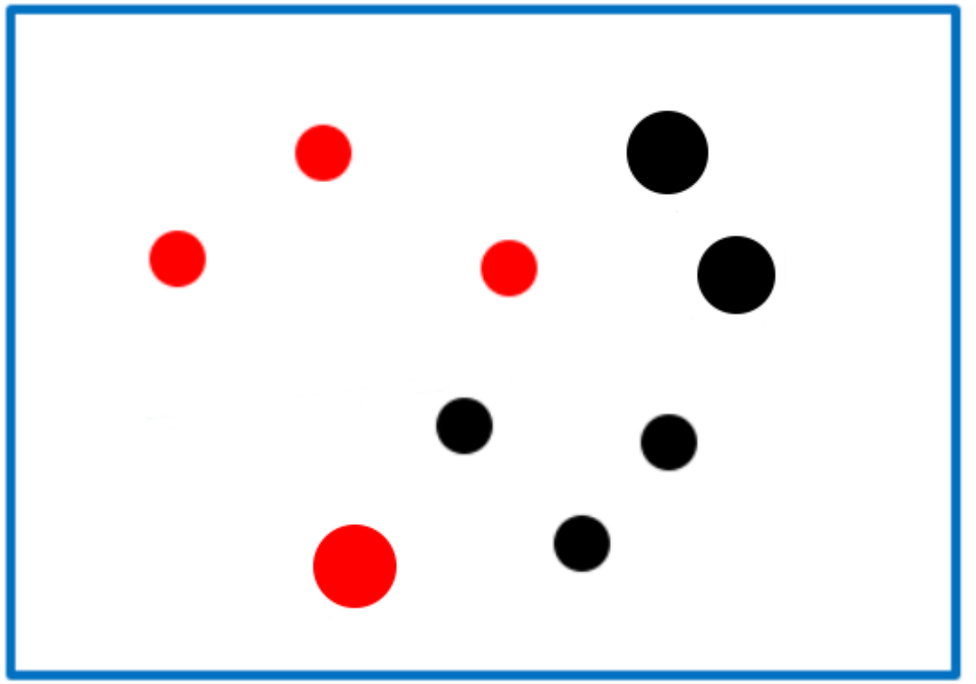
- Random Forests utilizan una estrategia llamada *bagging*, que busca ensamblar múltiples modelos, donde la baja correlación entre ellos permite mejorar el rendimiento.
- Si el ensamble (combinación) de estos modelos se hace utilizando otro clasificador, entonces el esquema se conoce como *stacking*.
- Una manera distinta de hacer ensambles es intentar que exista una correlación explícita entre los modelos, pero enfocada en la reducción del error.
- Este esquema de ensamble se conoce como *boosting*.



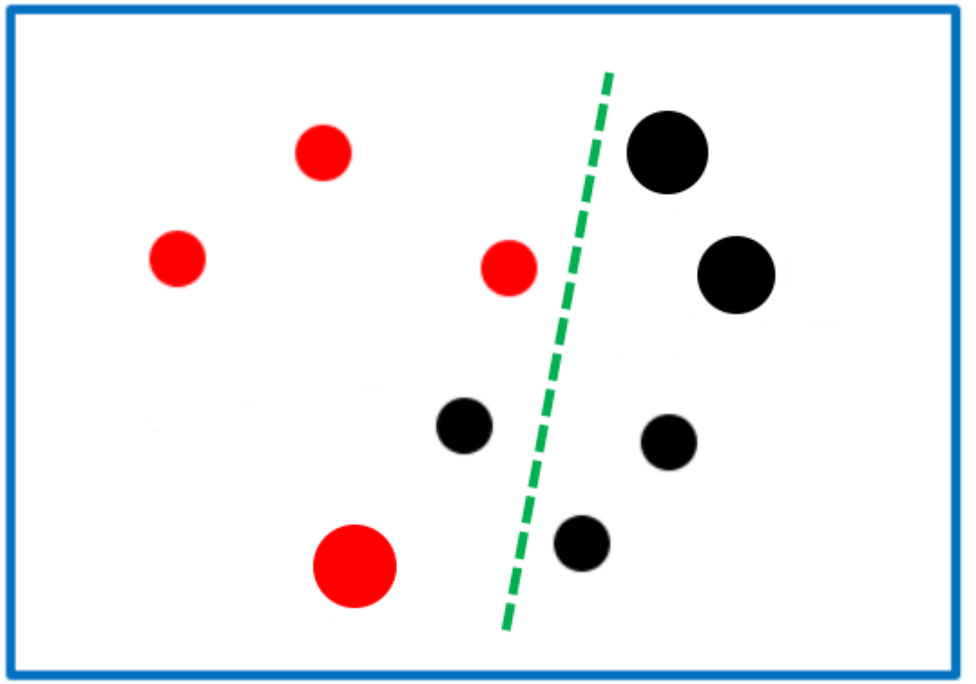




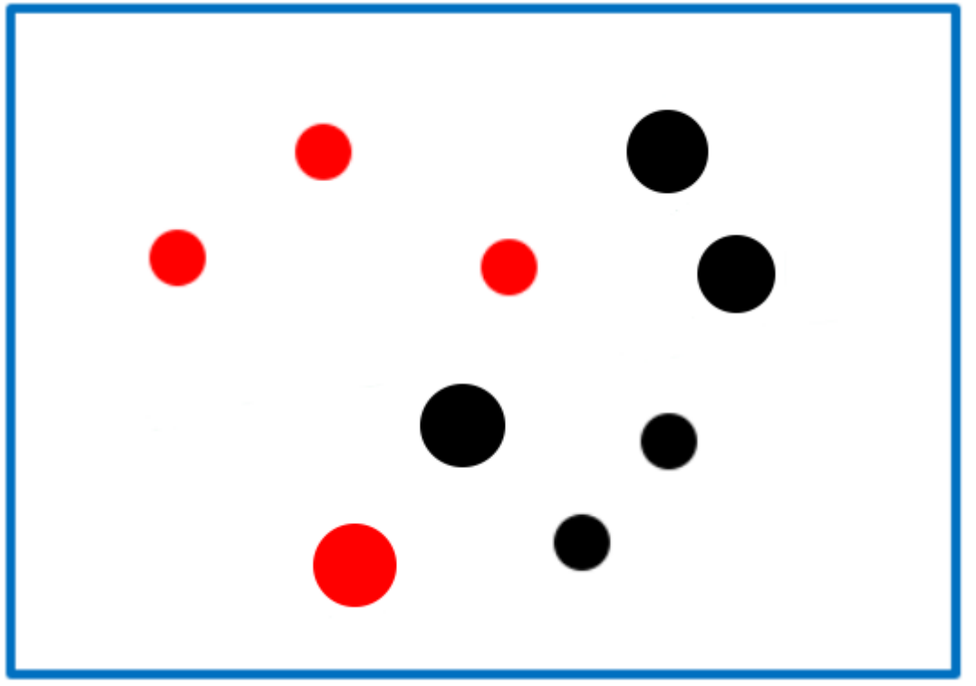
Construyo un modelo inicial



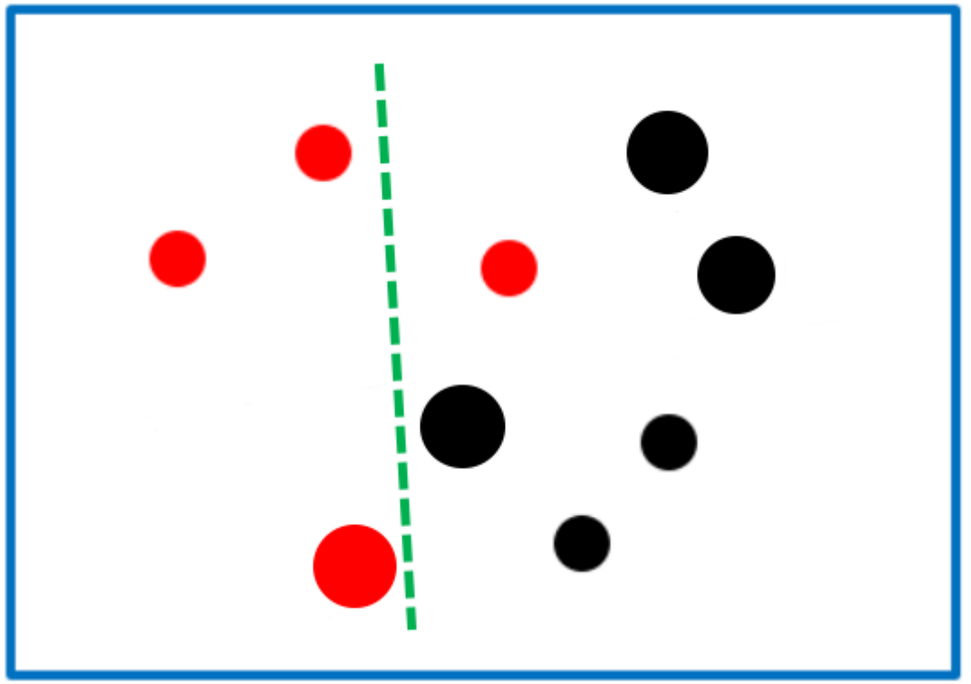
Doy más peso a ejemplos mal clasificados



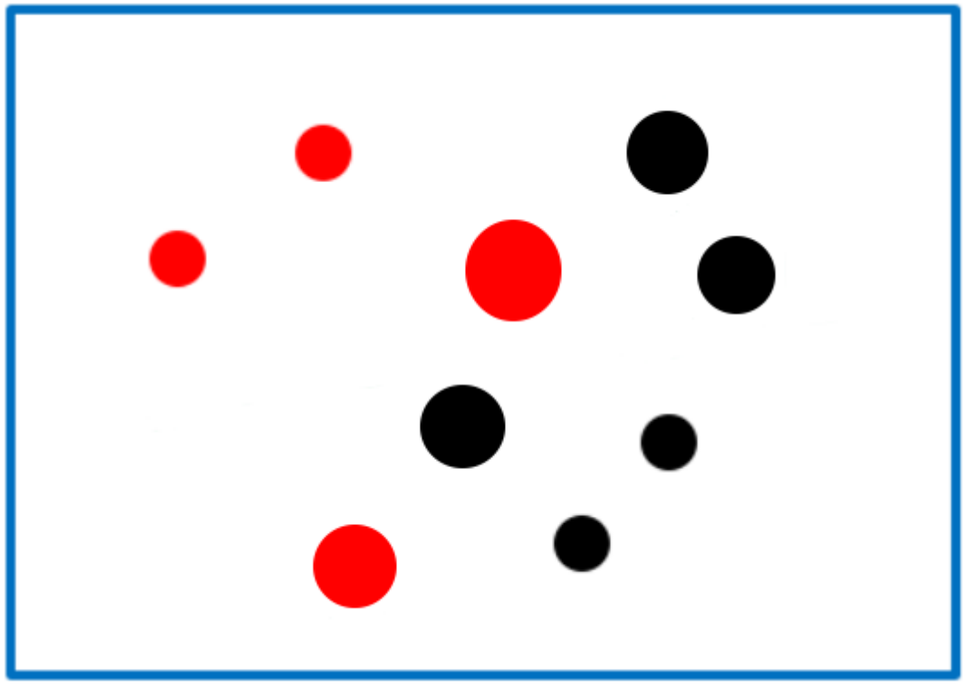
Construyo un nuevo clasificador,
que se combinará con el anterior



Doy más peso a los ejemplos mal clasificados



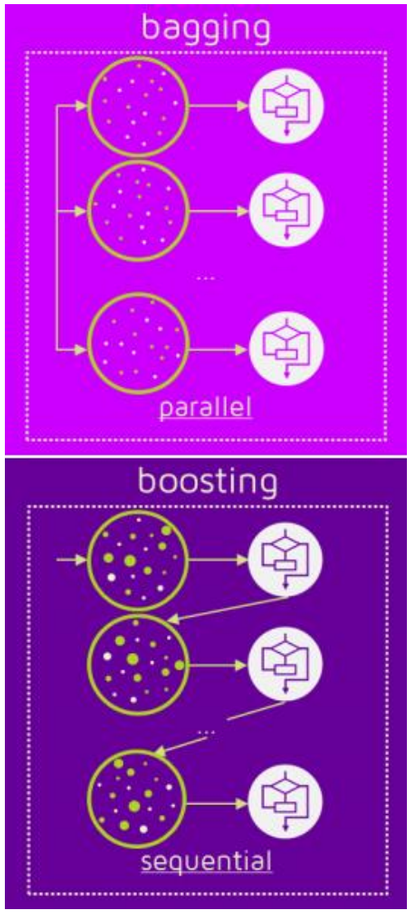
Construyo un nuevo clasificador,
que se combinará con los anteriores



Doy más peso a los ejemplos mal clasificados……

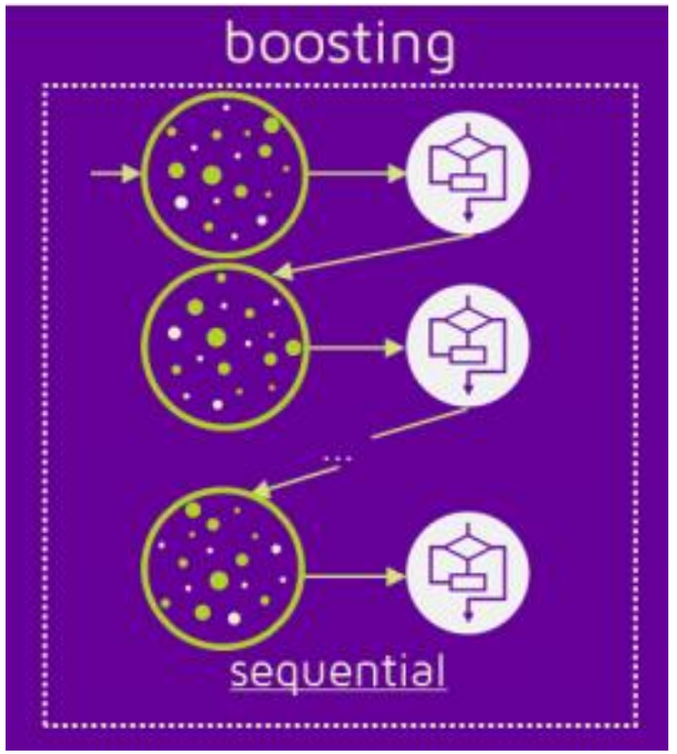
Boosting explora la construcción iterativa de ensambles

- En boosting, se agregan progresivamente modelos débiles, donde cada uno busca mejorar el rendimiento actual del ensamble.
- Para hacer esto, cada nuevo modelo ve una “versión modificada” de la señal de supervisión.
- El cómo modificar la supervisión es la base de la diferencia entre los distintos enfoques de *boosting* existentes.
- El mecanismo de *boosting* más exitoso en la actualidad es *Gradient Boosting*



Boosting explora la construcción iterativa de ensambles

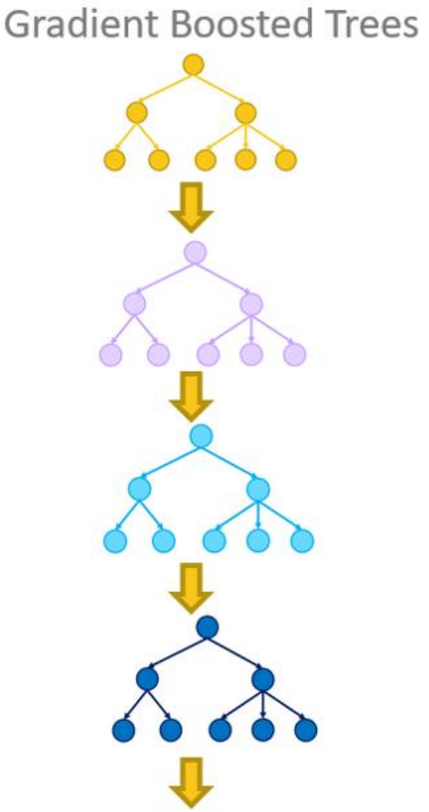
- Conceptualmente, lo métodos de *gradient boosting* realizan descenso de gradiente sobre un espacio vectorial de funciones (?).
- Cada nuevo modelo que se agrega busca aprender la diferencia entre las etiquetas y la predicción actual del ensamble, es decir, el error residual de este.
- Al agregar aditivamente este modelo al ensamble (descender en la dirección del menos gradiente), se disminuye el error de este.
- Este proceso se repite hasta alcanzar una condición de fin.

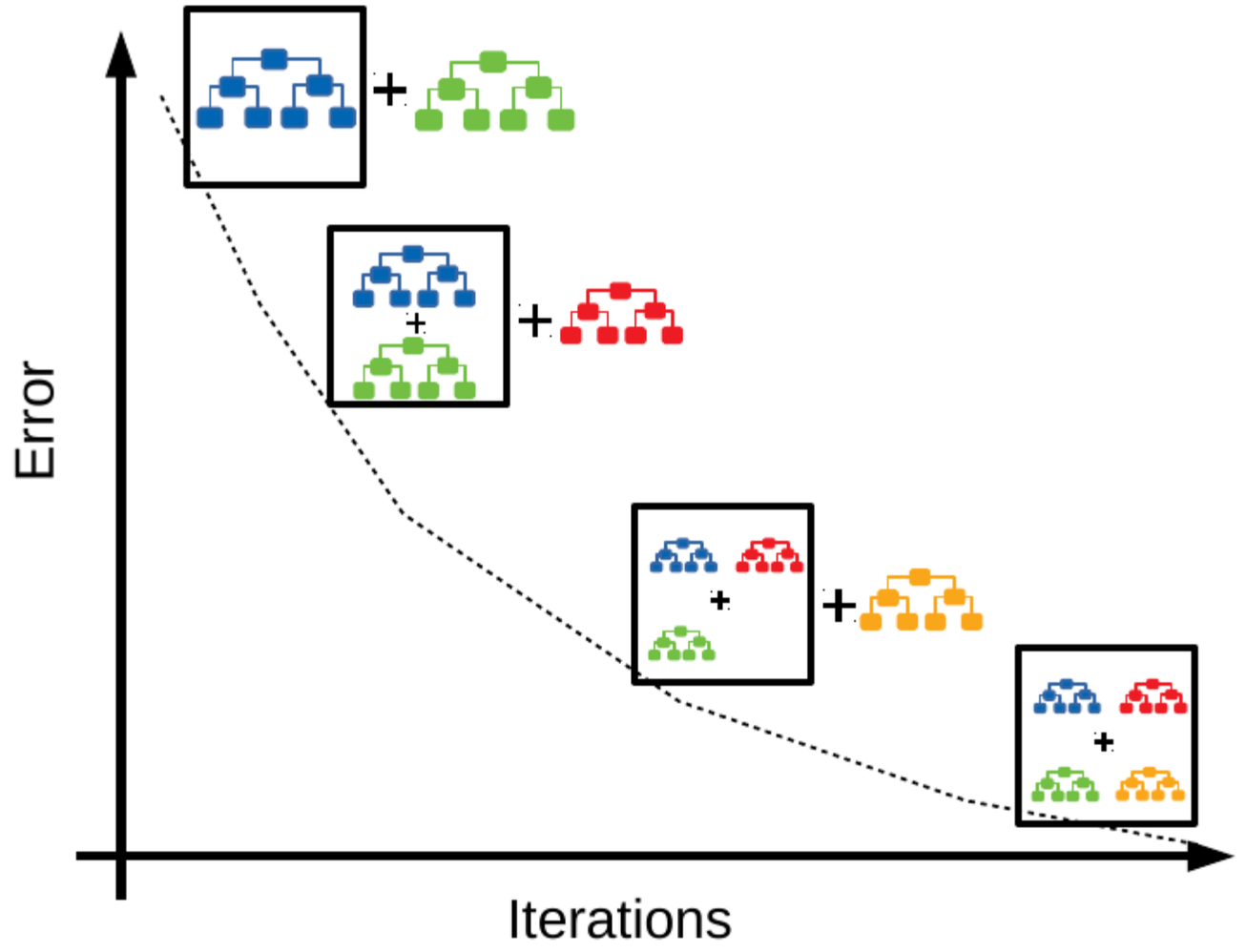


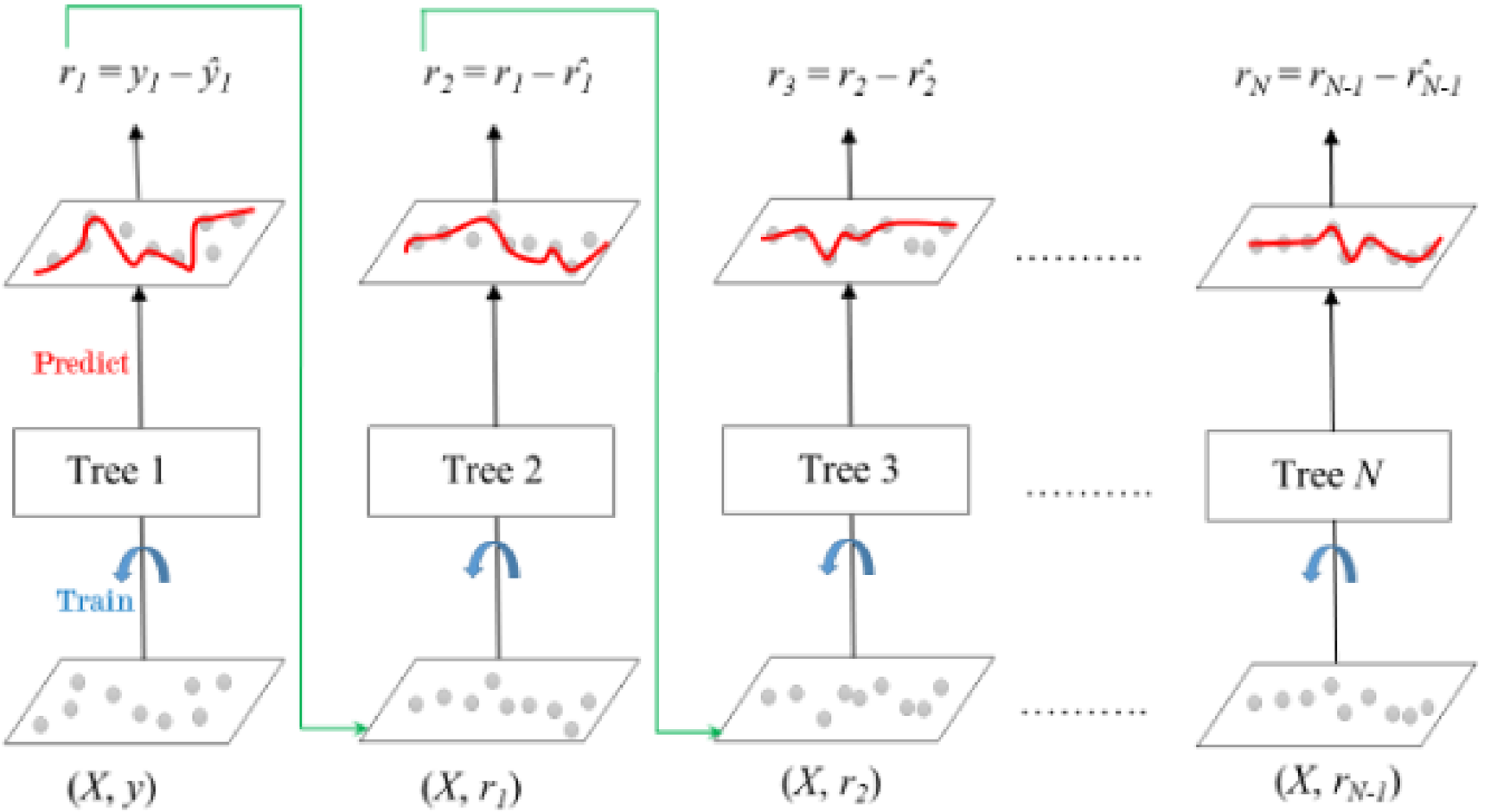
¿Cómo funciona este *Gradient Boosting*?

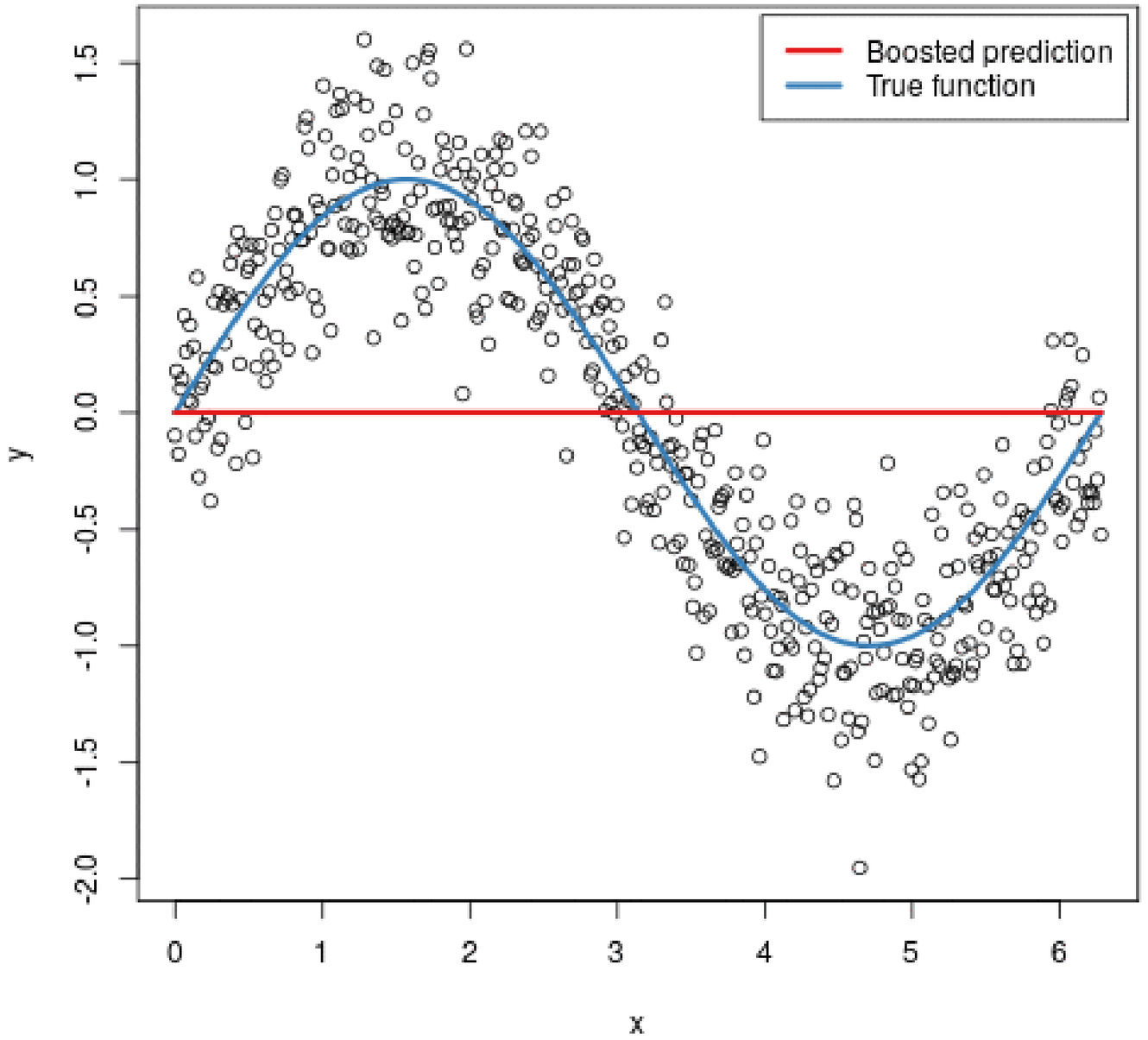
1. Entrenar un clasificador/regresor: $F_1(x) = h_0(x) \simeq y$
2. Entrenar un nuevo clasificador/regresor, pero ahora usando el residuo de F_1: $h_1(x) \simeq y - F_1(x)$
3. Sumar este nuevo clasificador/regresor al ensamble: $F_2(x) = F_1(x) + h_1(x)$
4. Entrenar un nuevo clasificador/regresor con los residuos de F_2: $h_2(x) \simeq y - F_2(x)$
5. Sumar este nuevo clasificador/regresor al ensamble: $F_3(x) = F_2(x) + h_2(x)$
6. Continuar este proceso, hasta cumplir algún criterio de fin. El clasificador/regresor final estará dado por:

$$f(x) = \sum_{i=1}^{N} h_{i-1}(x)$$



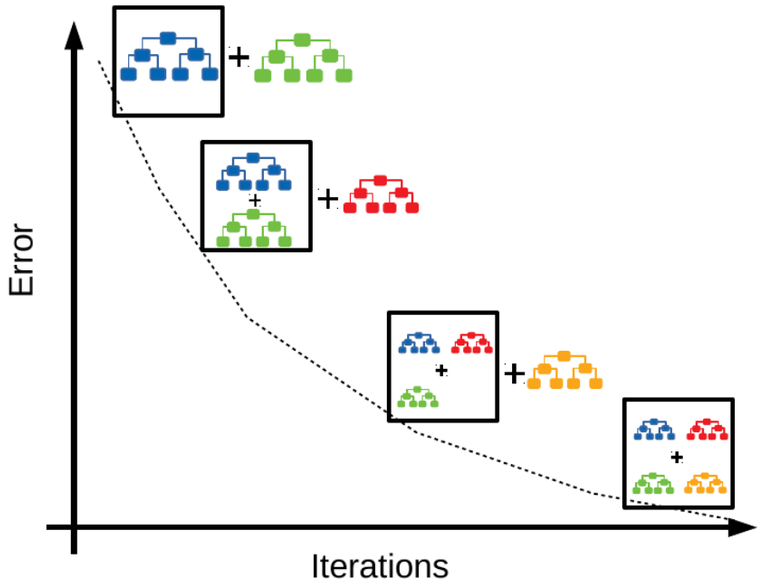

Error
Iterations


$r_1 = y_1 - \hat{y}_1$
$r_2 = r_1 - \hat{r}_1$
$r_3 = r_2 - \hat{r}_2$
$r_N = r_{N-1} - \hat{r}_{N-1}$
Predict
Tree 1
Tree 2
Tree 3
Tree N
Train
(X, y)
(X, r_1)
(X, r_2)
(X, r_{N-1})


Boosted prediction
True function
y
x
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5
-2.0
0
1
2
3
4
5
6

¿Cómo es el rendimiento de *Gradient Boosting*?

- En la práctica, el proceso anterior funciona muy bien (con ciertos agregados).
- Variedades modernas, como XGBoost o LightGBM, son ideales para procesar grandes volúmenes de datos tabulados, con rendimiento muchas veces superior a las redes neuronales profundas tradicionales.



Vamos a Colab...

Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

IIC2613 - Inteligencia Artificial

Boosting

Hans Löbel
Dpto. Ingeniería de Transporte y Logística
Dpto. Ciencia de la Computación