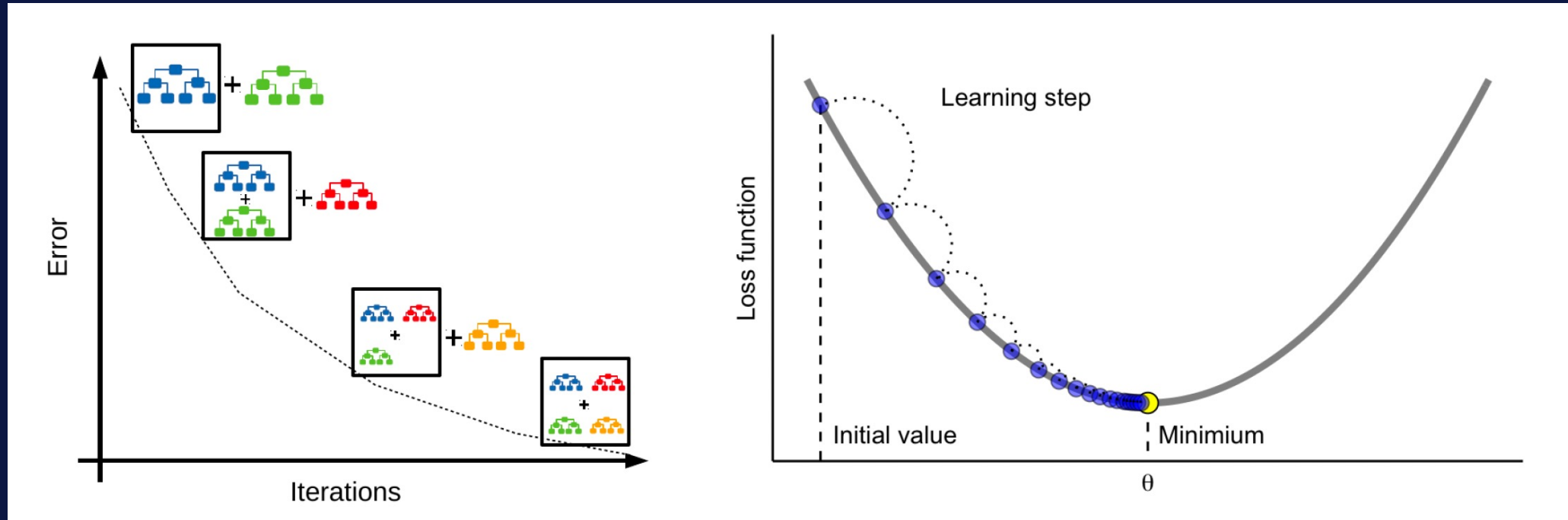


# GRADIENT BOOSTING MATEMATICA



# EXPLICACIÓN

1. Empezamos con un decision tree básico ( $M_0$ ), el modelo predice la probabilidad:  $\widetilde{y}_0 = M_0(X)$
2. Comparamos la predicción ( $\widetilde{y}_0$ ) con la realidad ( $y$ ) y calculamos el error ( $r_0$ )  $\rightarrow r_0 = y - \widetilde{y}_0$
3. El siguiente decision tree intentará predecir  $r_0$ , en vez de  $y$ .
4. Iteraciones de trees (n iteraciones):
  1. El modelo  $M_1(X)$  intenta predecir con los atributos ( $X$ ) los residuos del anterior ( $\widetilde{r}_0$ ), en vez del target ( $y$ )
  2. Para hacer una nueva predicción de la probabilidad ( $\widetilde{y}_1$ ) debemos utilizar:
    1.  $\widetilde{r}_0 = M_1(X)$  (en este primer paso entrenamos un árbol, con target =  $r_0$ )
    2. La predicción inicial  $\widetilde{y}_0$
    3. Learning rate  $g$  (parametro que marca cuanto nos importa  $\widetilde{r}_0$  con respecto a  $\widetilde{y}_0$ )

El resultado es  $\widetilde{y}_1 = f(\widetilde{r}_0, \widetilde{y}_0, g)$

En esta función, entre otras transformaciones agregamos los resultados:  $\widetilde{y}_1 = h(\widetilde{y}_0 + g^* g(\widetilde{r}_0))$ .

Nota: la función  $h()$  es la función sigmoide, que obliga al resultado a ser entre 0 y 1, porque estamos en clasificación.

La función  $g()$  permite agregar predicción de probabilidad con la predicción de residuos.

3. Calculamos los residuos de nuevo:  $r_1 = y - \widetilde{y}_1$

La predicción final es  $\widetilde{y}_n$ , que es una gregación de las predicciones de todos los árboles que se han generado de forma iterativa.