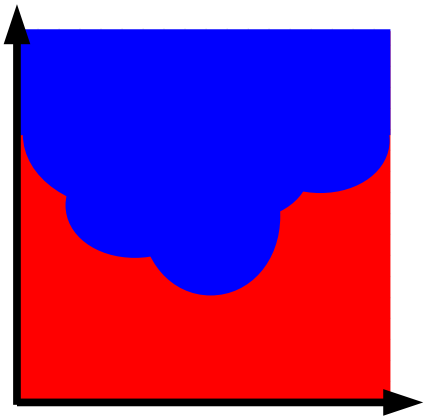
Aprendizaje Automático
Segundo Cuatrimestre de 2018

Ensambles de Modelos
(Ensemble Learning)

DEPARTAMENTO
DE COMPUTACION
Facultad de Ciencias Exactas y Naturales - UBA

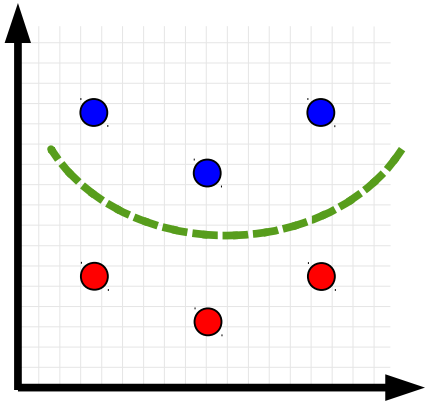
Sesgo Inductivo

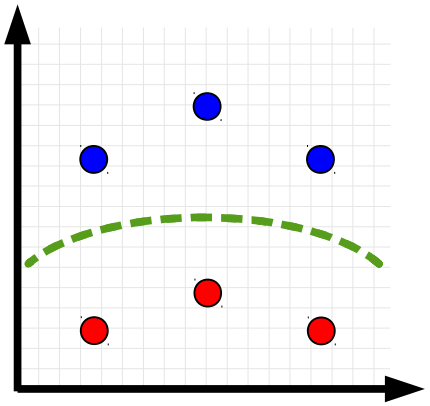
- El sesgo inductivo de un algoritmo de aprendizaje es el conjunto de afirmaciones que el algoritmo utiliza para construir un modelo.
- El sesgo inductivo incluye:
  - forma de las hipótesis (número y tipo de parámetros);
  - características de funcionamiento del algoritmo (cómo recorre el espacio de hipótesis hasta elegir un único modelo).
- Si un algoritmo de aprendizaje tiene sesgo fuerte:
  - Mayores restricciones al poder expresivo de las hipótesis.
  - Menores chances de aproximar bien a la función objetivo.

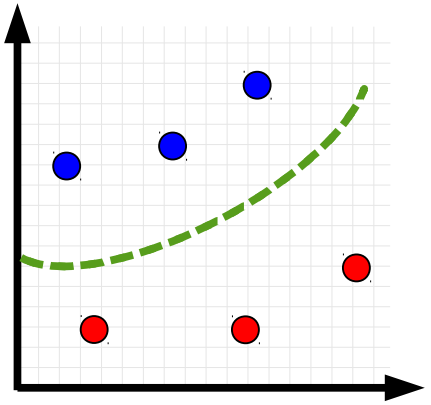


Función objetivo
(desconocida)

Posibles conjuntos de
datos de entrenamiento
(muestras)



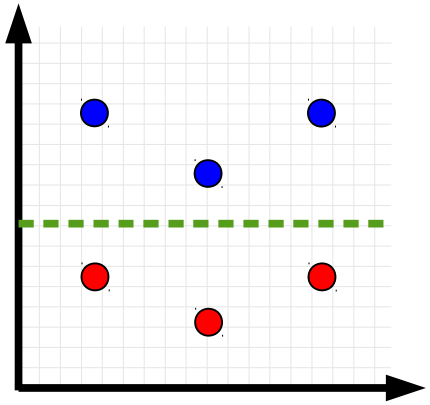


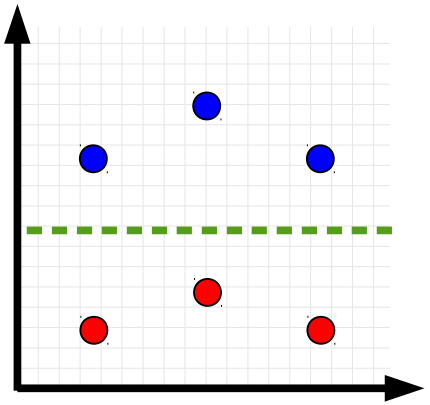


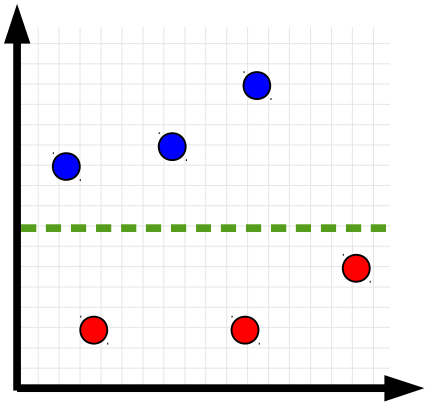
Línea de corte: **parábola.**

Bajo sesgo: mejores chances de acercarse al objetivo.

Alta varianza: los modelos construidos **cambian mucho** según la muestra.





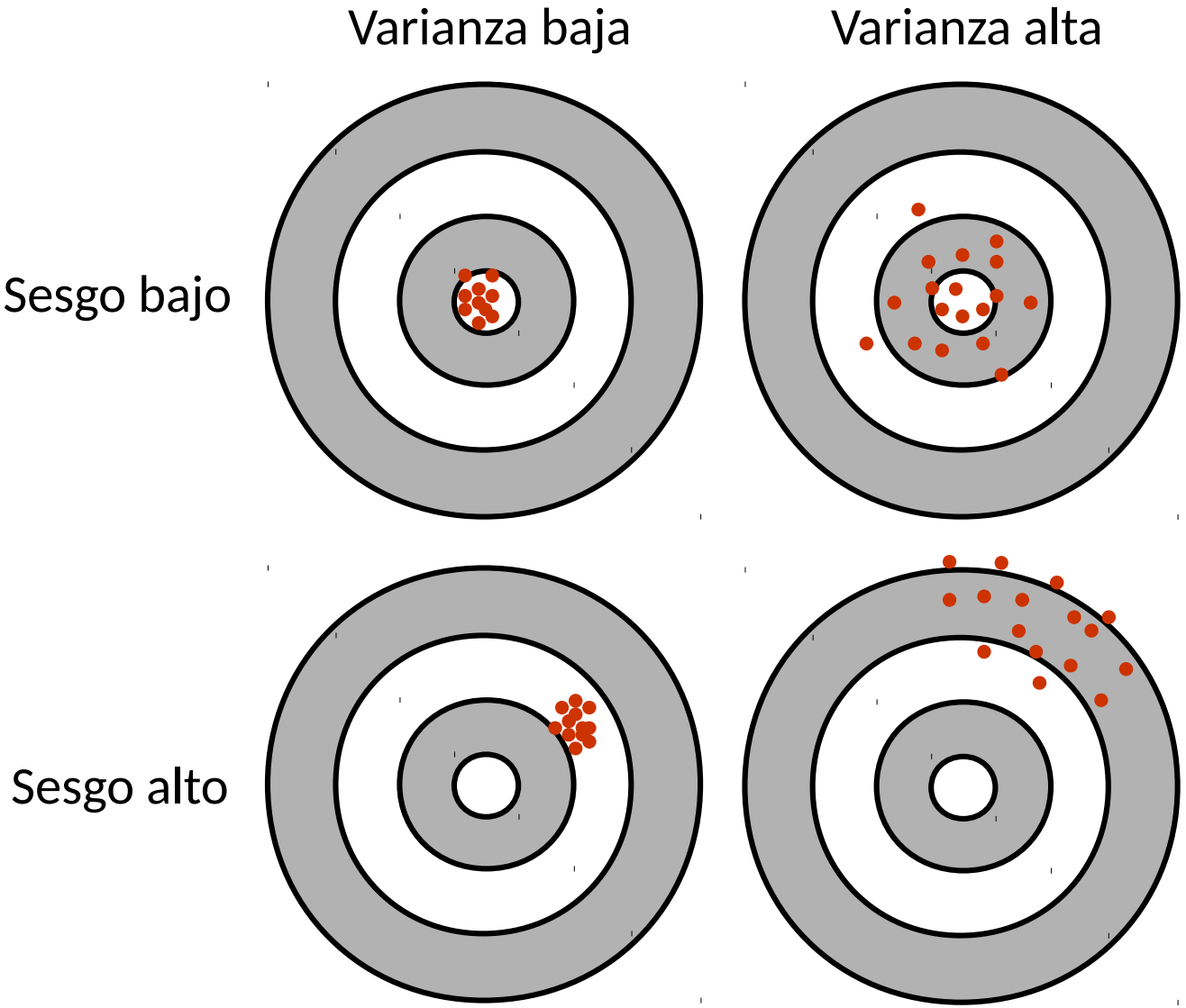


Línea de corte: **recta horizontal.**

Alto sesgo: peores chances de acercarse al objetivo.

Baja varianza: los modelos construidos **cambian poco** según la muestra.

Sesgo vs. Varianza



Puntos: modelos construidos sobre muestras distintas por 4 algoritmos.

Formalizando un poco...

(Veremos esto en mejor detalle después de introducir regresión.)

observaciones

ruido en los datos + información no capturada por los atributos

$$Y = f(X) + \epsilon$$

función objetivo

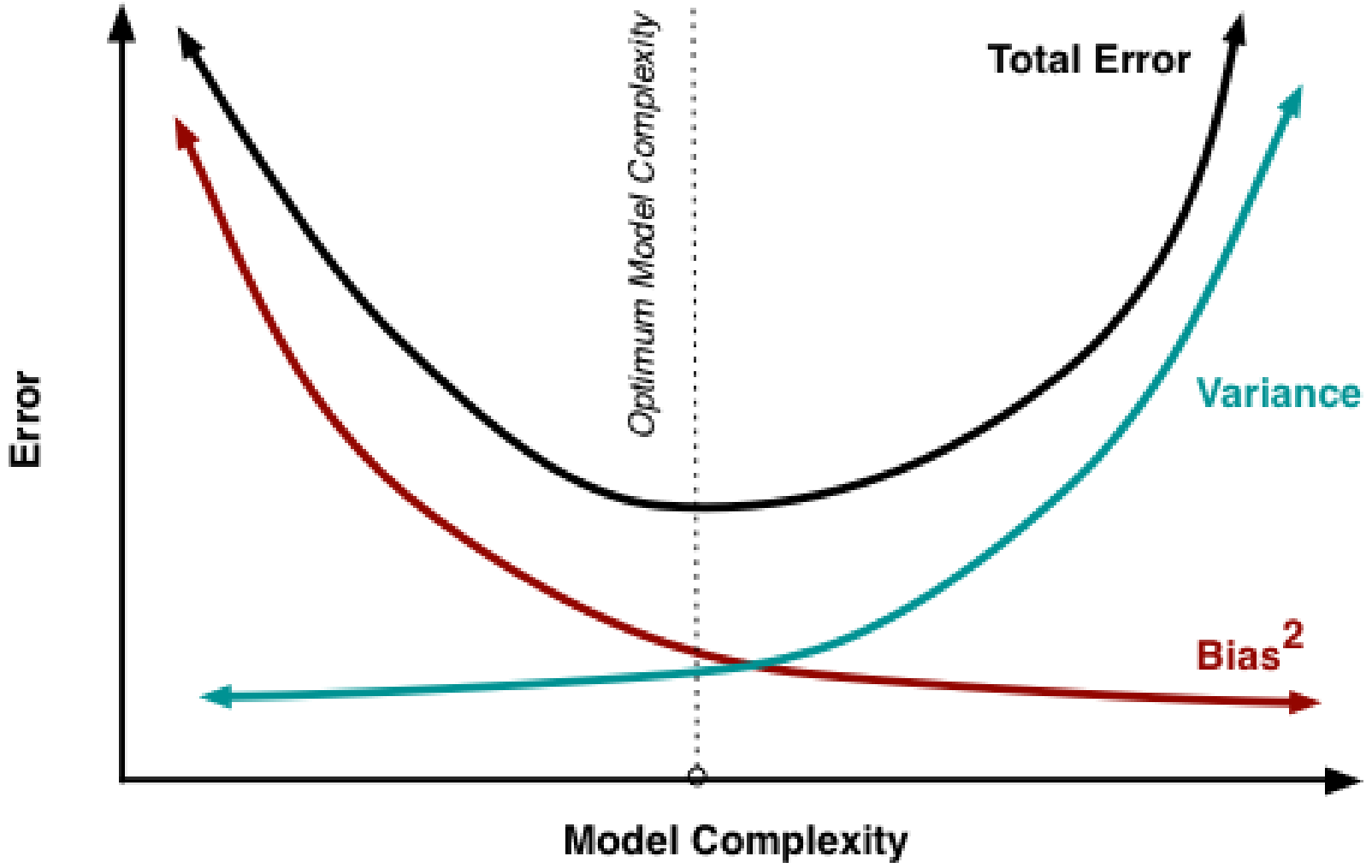
$$\mathbf{E}\Big[\big(y - \hat{f}(x)\big)^2\Big] = \mathrm{Bias}(\hat{f}(x))^2 + \mathrm{Var}(\hat{f}(x)) + \mathrm{Var}(\epsilon)$$

error² del modelo

$$\mathrm{Bias}(\hat{f}(x))^2 = \big(\mathbf{E}[\hat{f}(x)] - y\big)^2$$

$$\mathrm{Var}(\hat{f}(x)) = \mathbf{E}\Big[\big(\hat{f}(x) - \mathbf{E}[\hat{f}(x)]\big)^2\Big]$$

Sesgo vs. Varianza



http://scott.fortmann-roe.com/docs/BiasVariance.html

Repaso de Estadística

- Sea $X^{(1)}$, ..., $X^{(n)}$ una muestra de n observaciones independientes tomadas de una población con media μ y varianza σ^2.

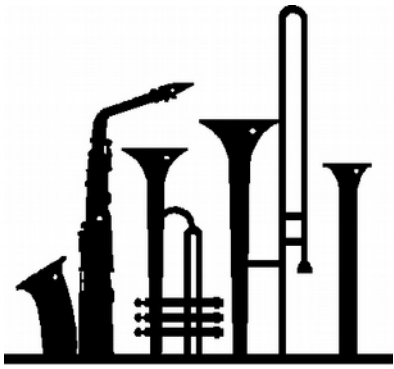
- Media muestral: $$\overline{X} = \frac{X^{(1)} + X^{(2)} + ... + X^{(n)}}{n}$$

- Esperanza de la media muestral: $$E(\overline{X}) = \mu$$

- Varianza de la media muestral: $$Var(\overline{X}) = \frac{\sigma^2}{n}$$

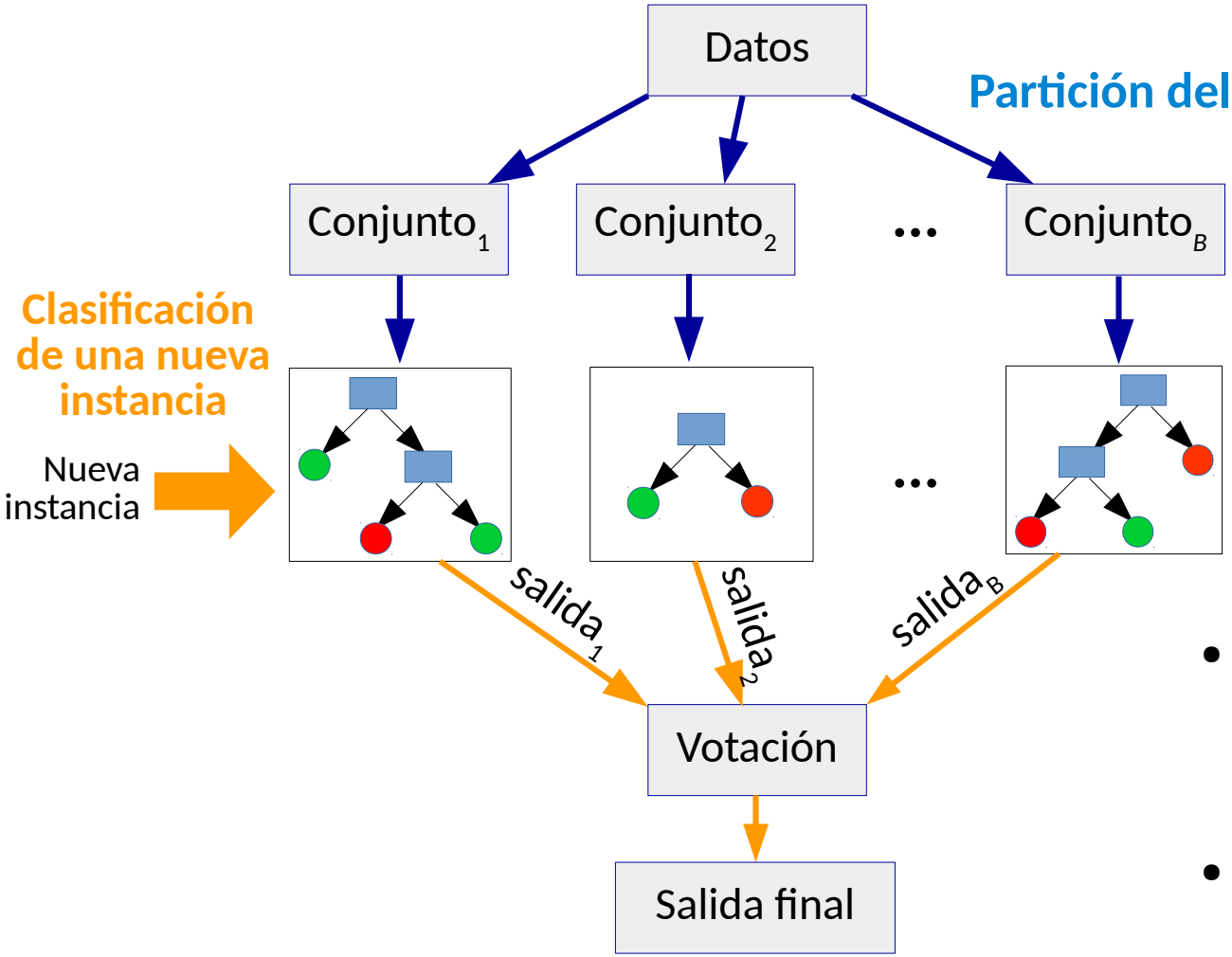
Al aumentar n, disminuye la varianza del estimador.

Ensambles de Modelos



- Entrenar *varios* modelos, c/u sobre datos distintos.
- Cada modelo sobreajusta de manera *diferente*.
  - Cada modelo: bajo sesgo, alta varianza.
  - Por ejemplo: árboles profundos.
- Votación: Para una nueva instancia, clasificarla con todos los modelos, y devolver la clase más elegida.
  - Esta votación reduce la varianza de la clasificación. ¡Magia!
  - Si los modelos individuales devuelven probabilidades, se puede hacer una votación ponderada.
  - En regresión, se puede devolver el promedio de los valores devueltos por los modelos individuales.

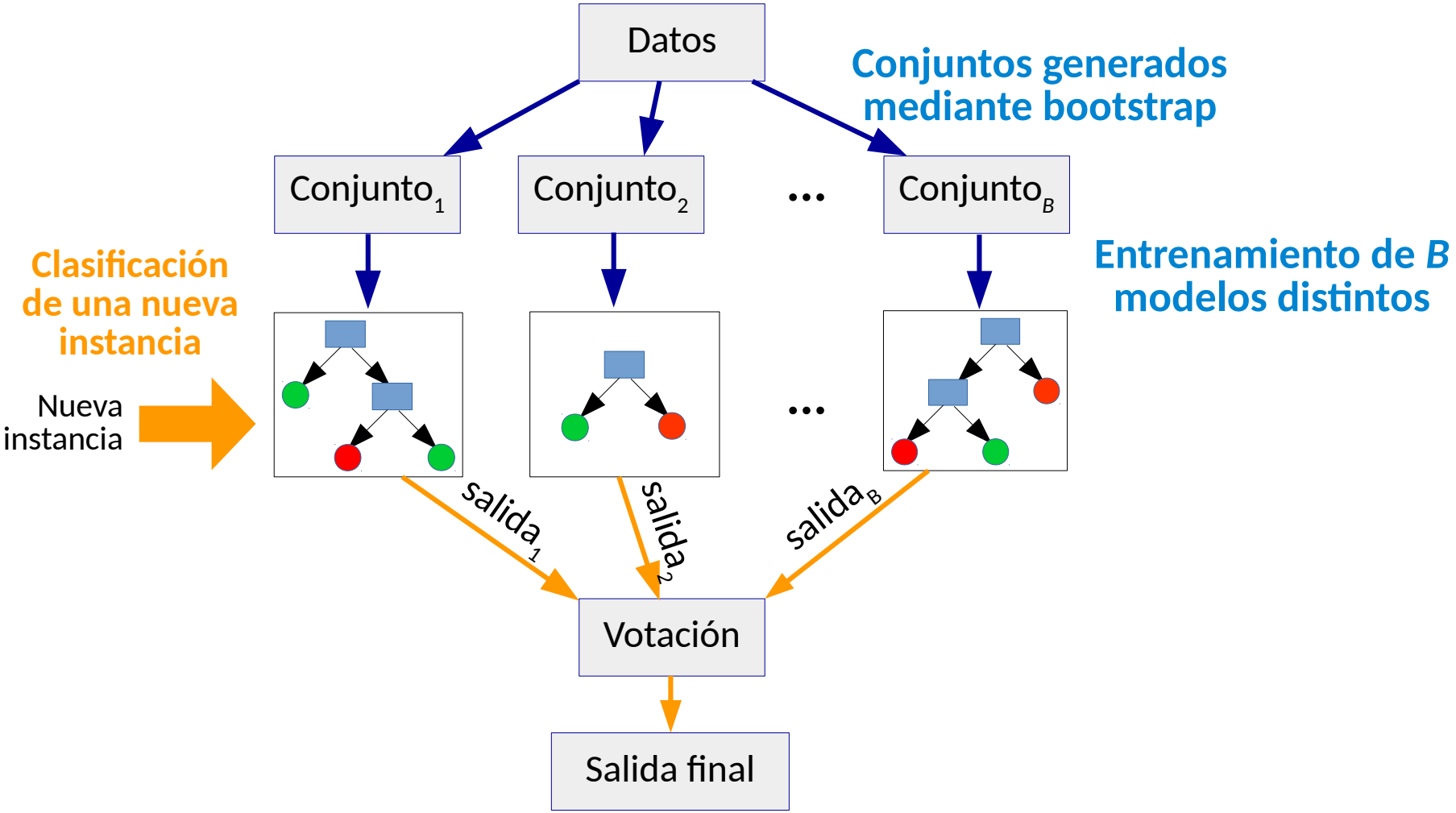
Ensambles de Modelos



- Poco práctico: los datos son caros. Veamos cómo refinar esta idea.
- Es importante que los modelos estén poco correlacionados.

Bagging (Bootstrap Aggregating)

- Construir nuevos conjuntos de entrenamiento usando bootstrap: muestreo con reemplazo de las instancias.



Random Forest

- Problema de bagging con árboles:
  - Si pocos atributos son predictores fuertes, todos los árboles se van a parecer entre sí! :-(
  - Esos atributos terminarán cerca de la raíz, para todos los conjuntos generados con bootstrap.
- Random Forest:
  - Igual a bagging, pero en cada nodo, considerar sólo un subconjunto de *m* atributos elegidos al azar.

Bagging vs. Random Forest

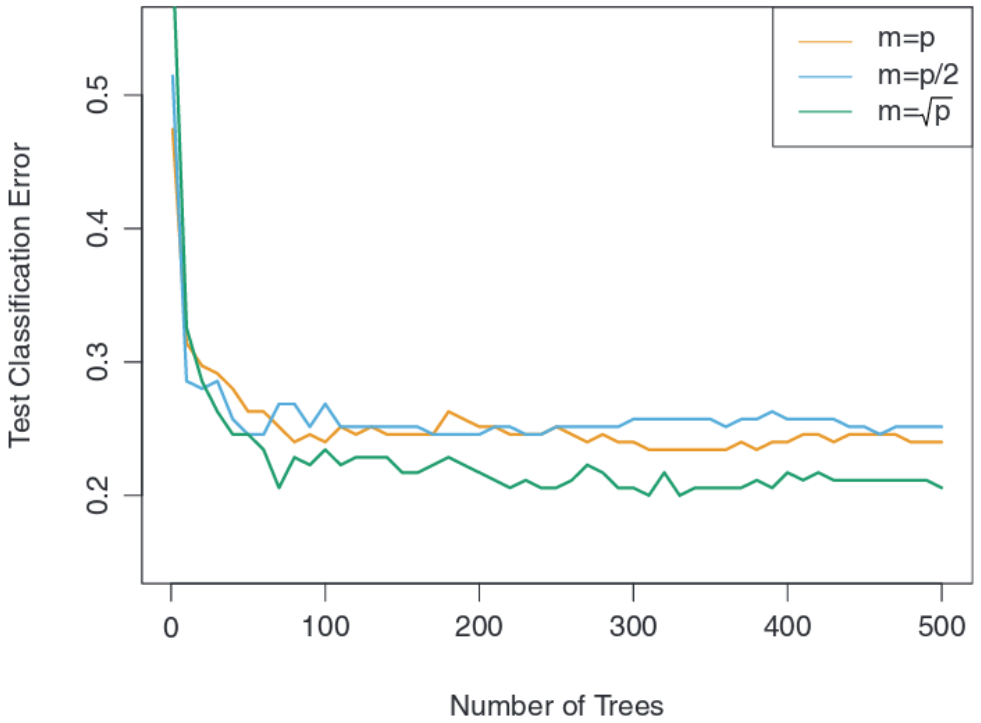
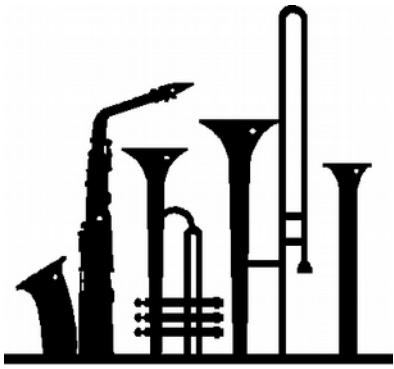


FIGURE 8.10. *Results from random forests for the 15-class gene expression data set with $p = 500$ predictors. The test error is displayed as a function of the number of trees. Each colored line corresponds to a different value of m, the number of predictors available for splitting at each interior tree node. Random forests ($m < p$) lead to a slight improvement over bagging ($m = p$). A single classification tree has an error rate of 45.7%.*

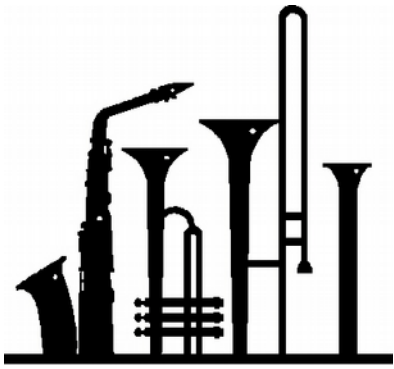
Además, algo positivo de bagging y random forest es que no sobreajustan a medida que se agregan modelos al ensamble.

Ensambles de Modelos

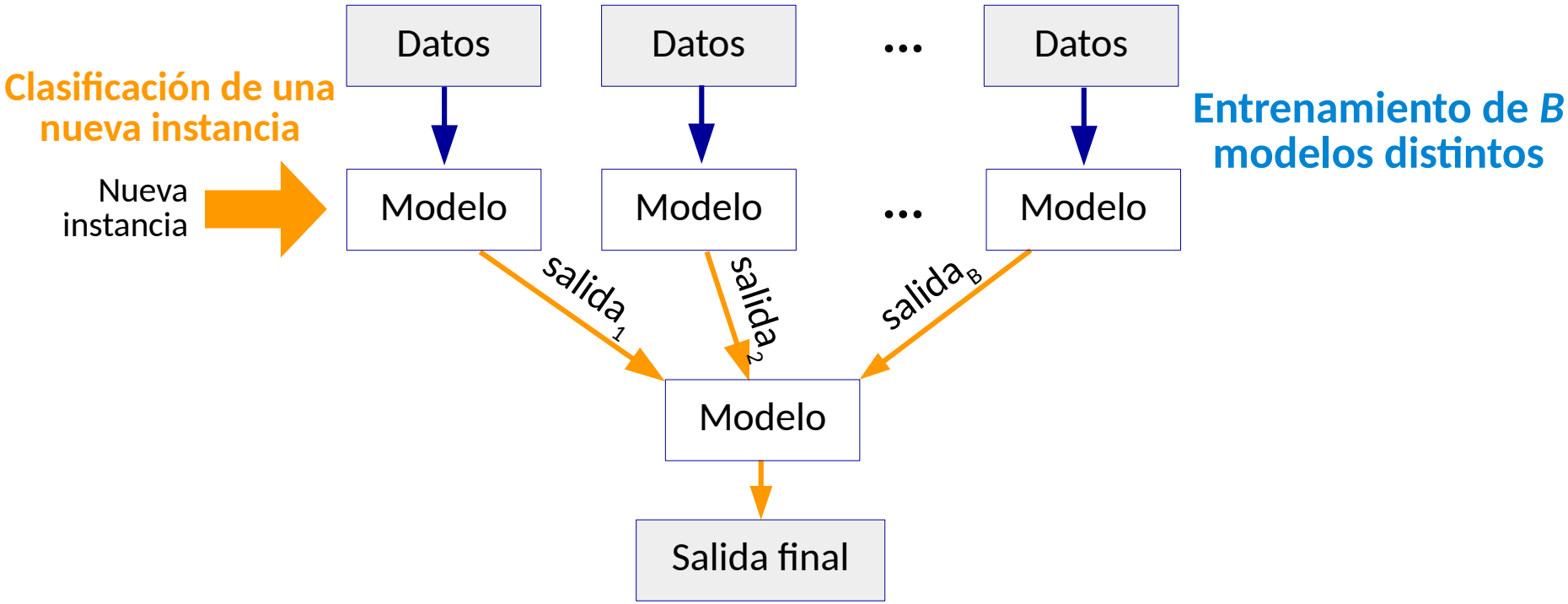


- Boosting
  - Comenzar con un modelo (simple) entrenado sobre todos los datos: h_0
  - En cada iteración i, entrenar h_i dando mayor importancia a los datos mal clasificados por las iteraciones anteriores.
  - Terminar al conseguir cierto cubrimiento, o luego de un número de iteraciones.
  - Clasificar nuevas instancias usando una votación ponderada de todos los modelos construidos.
  - Para pensar: ¿cómo hace Boosting para minimizar el sesgo y la varianza?

Ensambles de Modelos



- Stacking
  - Entrenar diferentes modelos (modelos base) y un modelo más, que decide, dada una instancia nueva, qué modelo usar.



Resumen

- Sesgo vs. varianza
- Ensambles de modelos:
  - Bagging
  - Random Forest
  - Boosting
  - Stacking