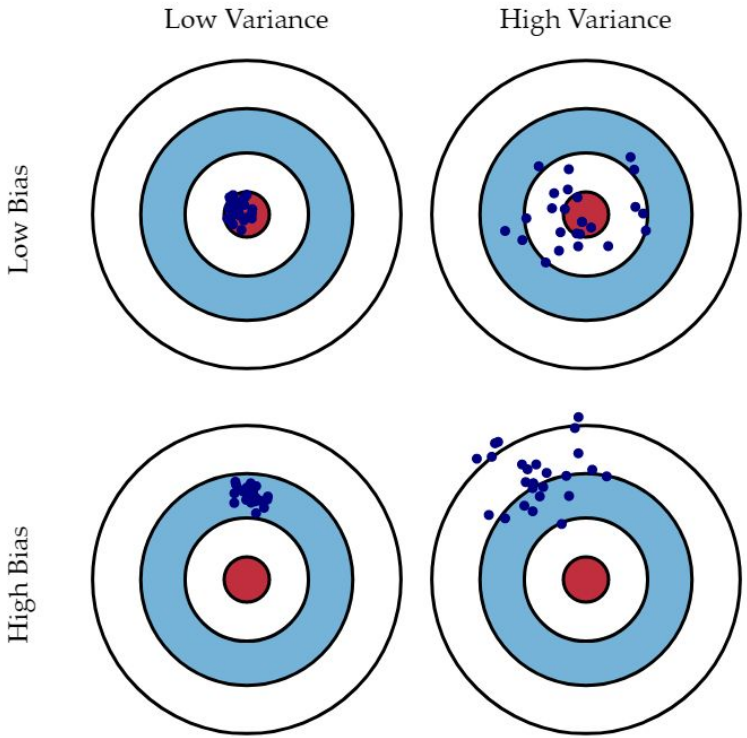
Aprendizaje Automático

Ensambles de modelos

Viviana Cotik

1er cuatrimestre 2019

Sesgo y Varianza



Error debido a sesgo (o bias):
debido a diferencia entre predicción del modelo (o promedio de predicciones) y valor correcto

Error debido a varianza:
la variabilidad de la predicción de un modelo para unos datos dados. Cuánto varían los resultados para distintos datos.

Sesgo y varianza

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

Y, X datos de entrada

$$Err(x) = \text{Bias}^2 + \text{Variance} + \text{Irreducible Error}$$

Para disminuir el error requerimos de un método que tenga **bajo sesgo** y **baja varianza**.

Sesgo y varianza

Algoritmos de aprendizaje inestables: aquellos que sufren **cambios importantes** antes **pequeñas variaciones en datos de entrenamiento** (ej: árboles de decisión, redes neuronales). Ej. de algoritmos estables: regresión lineal, vecino más cercano.

- **Predictores rígidos:** **poca flexibilidad** (menos complejos). **Mayor error de sesgo.** (estimación de un error muy complejo con un modelo simple, por ej. lineal)
- **Predictores flexibles:** más complejos. **Mayor error de varianza y menor error de sesgo.**

Flexibilidad de modelos

Dilema: Bias - Varianza: a bajo sesgo alta varianza y a alta varianza bajo sesgo.

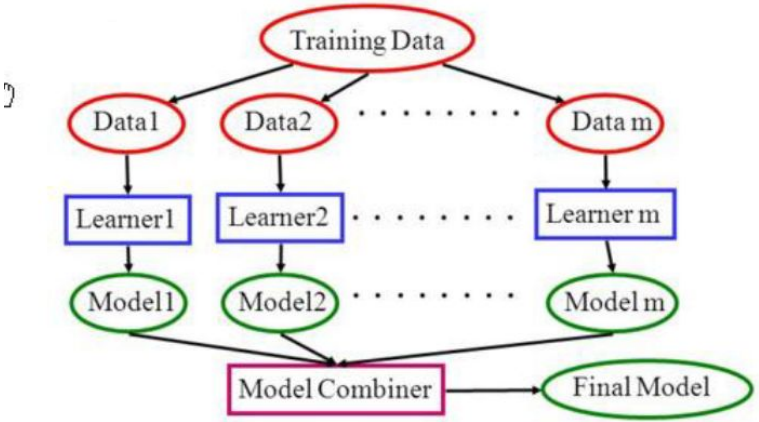
- **bajo bias y alta varianza:** curva que pasa por cada ejemplo de entrenamiento
- **baja varianza alto bias:** línea recta que atraviesa los datos.

Soluciones:

- usando predictores con bajo sesgo, disminuir la varianza
  (i.e. construir muchos predictores y promediarlos. ej. bagging y random forest)
- reducir sesgo de predictores
  (i.e. construir predictores en serie, de forma tal de disminuir el sesgo. ej. boosting)

Ensambles

- Uso de un **conjunto de modelos*** para construir un **meta-modelo*2**.
  * distintos datos o distintos algoritmos o parámetros
  *2 combinando decisiones (ej. por votación o promedio)
- **Sabiduría de la multitud**. Usar conocimiento de distintas fuentes para tomar decisiones.



- Muy usados en competencias

Ensambles

A definir:

- **“expertos” en una misma o en distinta área**
- **cómo combinar decisiones** (votación simple, votación ponderada, promedio, promedio pesado, promedio condicional...)

Ensambles Planos:

. comité de expertos en **un mismo tema**
. con distinta opinión en algunos casos

- Bagging
- Boosting
- Random Forest
- ...

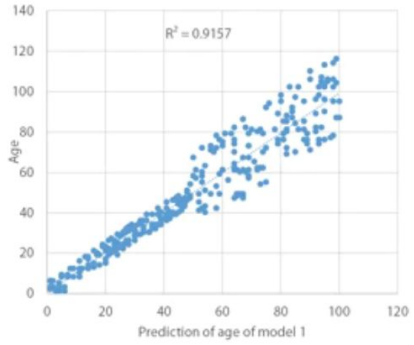
Ensambles Divisivos

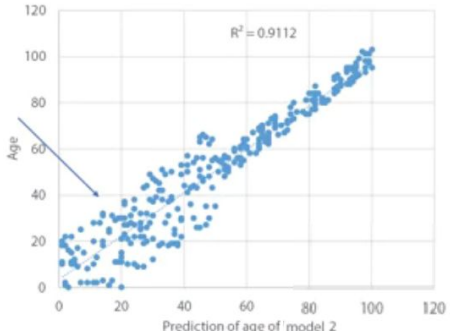
.Comité de expertos en **distintas áreas.**
.Problema dividido en **subproblemas** con poca superposición.
.Se necesita decisión acerca de **cómo combinar los resultados**

- Stacking
- Mezcla de expertos

Ensambles

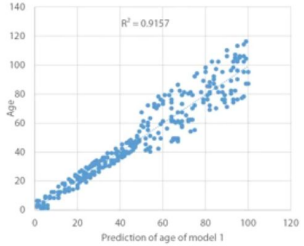
Tenemos dos modelos.

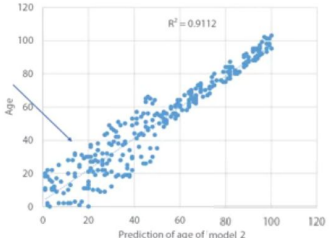




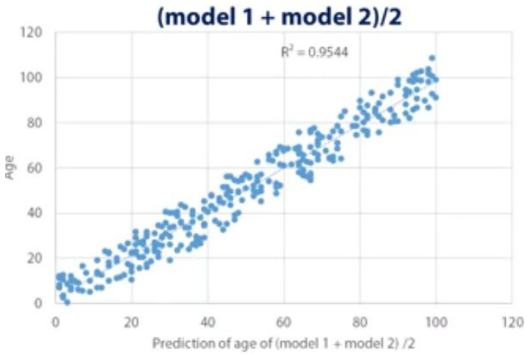
Ensambles

1. promedio
2. promedio pesado
3. promedio condicional

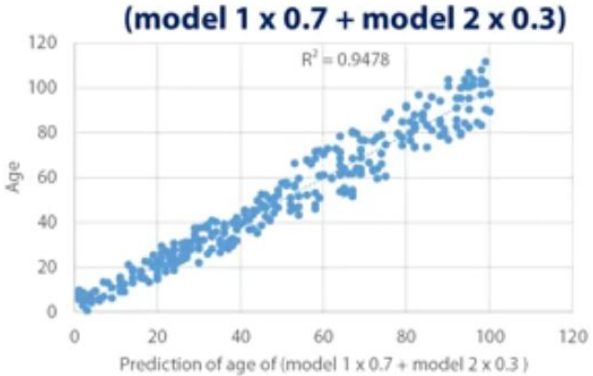




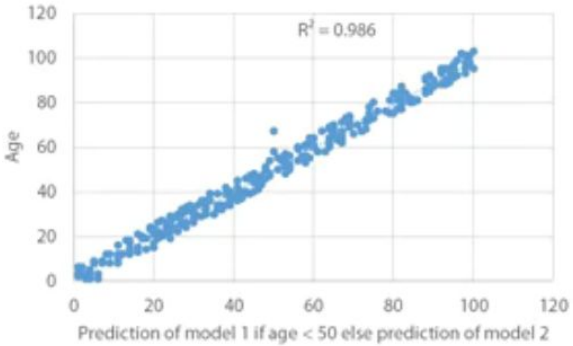
1



2



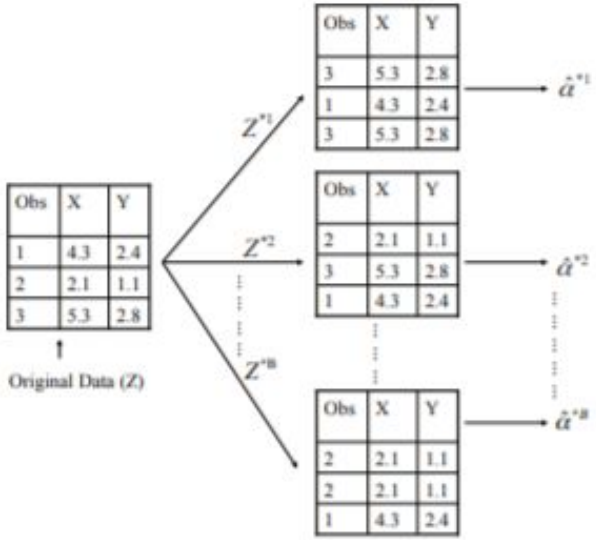
3



How to Win a Data Science Competition: Learn from Top Kagglers. https://www.coursera.org/learn/competitive-data-science#syllabus

Bootstrapping

- **Herramienta estadística** (1979)
- Técnica de **resampleo a partir de un conjunto de datos** con **reemplazo.**



En el ejemplo hay tres datos. Se toman tres muestras con resampleo

Se puede utilizar para mejorar modelos de aprendizaje (por ej. árboles de decisión).

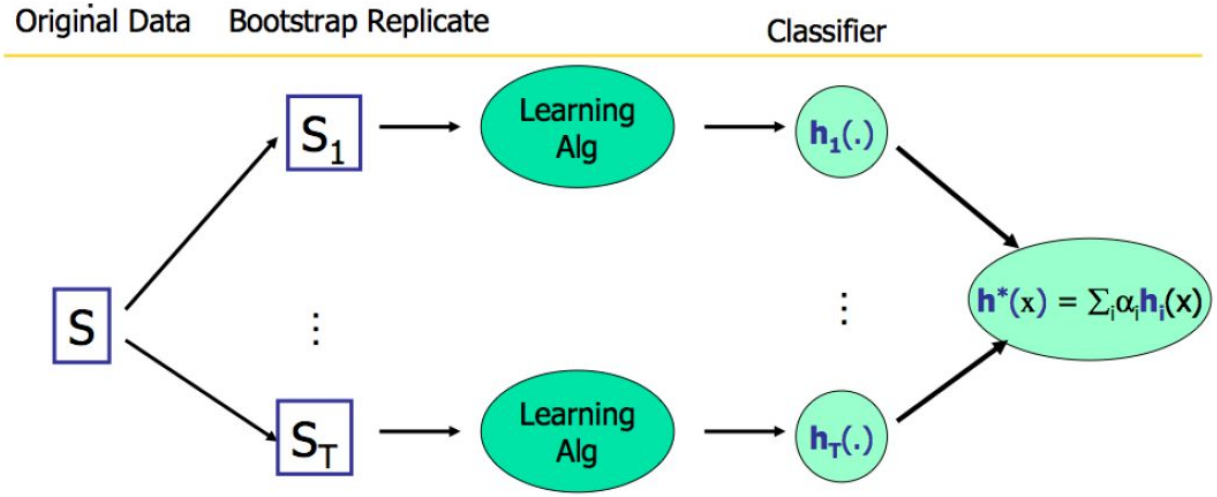
Bagging (Bootstrap aggregating)

Método para reducir varianza de un modelo

- En cjto. de observaciones independientes: **z1, ..., zn** con varianza σ^2, la varianza de la media de las observaciones es σ^2/n
- Entonces, **promediar un cjto de observaciones reduce la varianza**. Podemos tomar muchos conjuntos de entrenamiento y hacer un promedio de predicciones.
- **Pero..** no tenemos muchos conjuntos de entrenamiento. Entonces: **bootstrap** (tomamos muestras repetidas de un único conjunto de entrenamiento)

Útil para casos en que resultados son sensibles a los cjtos de entrenamiento

Bagging (Bootstrap aggregating)



- Each S_i is bootstrap replicate
- h_i = classifier based on S_i
- $\alpha_i = 1/T$

Se hace **promedio** en caso de **predicción numérica**.

Se hace **votación** en caso de **predicción de variable categórica**.

Bagging (Bootstrap aggregating)

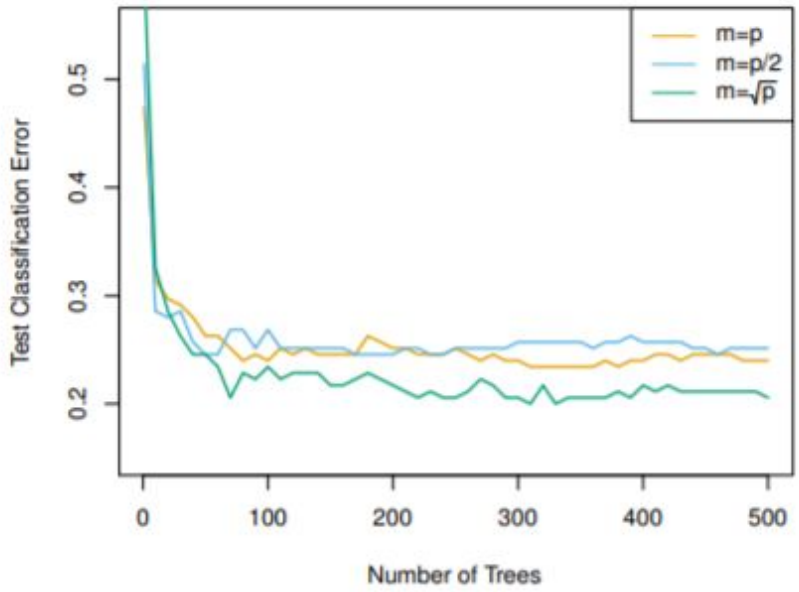
$$\hat{f}_{\text{bag}}(x) = \frac{1}{B}\sum_{b=1}^{B}\hat{f}^{*b}(x)$$

- Se generan **B datasets de entrenamiento** con **bootstrapping**.
- Se entrena el método con el **b-ésimo cjto. de entrenamiento** y obtenemos **ˆf *b (x)**, la predicción en el punto x.
- Se promedian todas las predicciones.

Random Forest

- Mejora a bagging cuando se usa con árboles de decisión. Allí:
  - si hay atributos que son predictores fuertes (ej: alto Information Gain) los árboles serán muy similares.
- Se intenta **eliminar correlación de los árboles**. Para reducir varianza es **mejor promediar cantidades no correlacionadas**.
- **Random Forest**
  - En **cada split** de un nodo se **considera sólo un subconjunto m de los p atributos** (m elegido al azar)**. m ≅ √p**

Bagging vs. Random Forest



500 árboles. Expresión de genes

bagging: m=p

random forest: m= **√p**

An Introduction to Statistical Learning, James, Witten, Hastie, Tibshirani

Boosting

Difiere de bagging en que **cada modelo se arma usando información de modelos anteriores**. Se computan pesos para mejorar los casos en que el algoritmo dio mal.

Procedimiento:

- h_0: modelo simple entrenado sobre todos los datos
- En cada iteración i, entrenar h_i dando mayor importancia a los datos mal clasificados por la iteración anterior (dándoles distinto peso). (Se le da peso a la aparición del dato en el modelo)
- Terminar luego de un número de iteraciones.
- Clasificar nuevas instancias usando una votación de todos los modelos construidos.

Esto es weight-based boosting. Ej: AdaBoost

Ejemplo de Residual Based Boosting

Rownum	x0	x1	x2	x3	y	pred	error
0	0.94	0.27	0.80	0.34	1	0.80	0.20
1	0.84	0.79	0.89	0.05	1	0.75	0.25
2	0.83	0.11	0.23	0.42	1	0.65	0.35
3	0.74	0.26	0.03	0.41	0	0.40	-0.40
4	0.08	0.29	0.76	0.37	0	0.55	-0.55
5	0.71	0.76	0.43	0.95	1	0.34	0.66
6	0.08	0.72	0.97	0.04	0	0.02	-0.02

Rownum	x0	x1	x2	x3	y	new pred
0	0.94	0.27	0.80	0.34	0.2	0.15
1	0.84	0.79	0.89	0.05	0.25	0.20
2	0.83	0.11	0.23	0.42	0.35	0.40
3	0.74	0.26	0.03	0.41	-0.4	-0.30
4	0.08	0.29	0.76	0.37	-0.55	-0.20
5	0.71	0.76	0.43	0.95	0.66	0.24
6	0.08	0.72	0.97	0.04	-0.02	-0.01

. tenemos datos con valor esperado y el error de correr un modelo.

. ahora el error es el valor esperado (y).
re-entrenamos con las mismas features.

Para predecir la fila 1, con este ejemplo. Predicción final = 0.75 +0.20 = 0.95

Ej: XGBoost

https://www.coursera.org/lecture/competitive-data-science/boosting-Ra7di

Stacking

Hacer varias predicciones a un held-out dataset

Usar esas predicciones para armar un nuevo dataset, con el cuál se entrena un nuevo modelo.

1. Dividir el dataset en entrenamiento y validación (aparte queda el held-out).
2. Entrenar distintos modelos (modelos base) con el dataset de entrenamiento
3. Hacer predicciones con los modelos entrenados sobre el conjunto held-out.
4. Usar los resultados de (3) para entrenar otro modelo. (meta modelo)

Resumen

- Ensambles, Bagging, Boosting, Random Forest, Stacking
  - **Arquitecturas:** paralelas (ej: bagging) vs. secuenciales (ej: boosting)
  - **Mismo clasificador** (la mayor parte) **vs. distinto clasificador** (ej: stacking)
  - Distintas formas de combinar resultados.
- Menor interpretabilidad
- Para clasificadores inestables (ej. árboles de decisión) usar **bagging** o **random forest**.
- Para clasificadores estables y simples (ej.: Naive Bayes): usar **boosting**.
- Solución al dilema bias-varianza
  - disminuir varianza usando predictores con bajo sesgo (i.e. construir muchos predictores y promediarlos. ej. bagging y random forest)
  - reducir sesgo de predictores (i.e. construir predictores en serie, de forma tal de disminuir el sesgo. ej. boosting)

Bibliografía

Capítulos de libros:

- ISLR 2 (2.2.2), 5 (5.2), 8 (8.2, 8.3.3, 8.3.4)
- Seni, Elder, "Ensemble Methods in Data Mining: Improving Accuracy Through Combining Predictions", Morgan & Claypool, 2010.

Otros:

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