Aprendizaje Estadístico

Ignacio Alvarez-Castro - Natalia da Silva

Instituto de Estadística-FCEA-UdelaR
Curso AEPE 2021

nachalca@iesta.edu.uy - nachalca.netlify.com - @nachalca
natalia@iesta.edu.uy - natydasilva.com - @pacocuak

Julio-2021

1. Introducción
2. Bosque aleatorio: algoritmo
3. Importancia y selección de variables
4. Propiedades estadísticas
5. Variantes de Random Forest

Métodos de agregación

- Métodos de agregación: combina múltiples modelos individuales entrenados independientemente para construir un modelo de predicción.
- Algunos ejemplos bien conocidos son; bagging, boosting, random forest entre otros.
- La diferencia principal entre los métodos de agregación son; el tipo de modelos individuales a ser combinados, las tareas que se asignan a cada modelo y la forma en que estos modelos son combinados.

A este tipo de modelos se les llama *ensembles*.

Bagging

Mencionamos que:

- la idea de bagging es promediar modelos ruidosos, aproximadamente insesgados, para reducir la varianza.
- Los árboles de tipo CART son *inestables*, cambios en el conjunto de entrenamiento puede resultar en cambios grandes del modelo final.

Los árboles son **buenos** candidatos para hacer *bagging*.

- capturan interacciones complejas
- tienen poco sesgo (si crecen lo suficiente)

Cada árbol es generado id, la esperanza del promedio es igual a la esperanza de un árbol, entonces el sesgo es el mismo pero hay una posible reducción en la varianza.

Idea general, bosque

Leo Breiman. Random forests.
Machine learning, 45(1):5–32, 2001

- Método supervisado de agregación Breiman [2001] ampliamente utilizado (más de 76 mil citas).
- Pueden ser usados tanto para problemas de clasificación como de regresión en una amplia variedad de problemas.
- En regresión, Random Forest (RF) suaviza la estimación promediando en un conjunto de árboles.

Se basa en promediar un conjunto de árboles aleatorizados.

Idea general: bosque

Recordemos el contexto del problema, $Y = f(X) + \varepsilon$, el estimador RF de f para el caso de respuesta numérica, se puede expresar como:

$$\hat{f}_{rf}(x) = \frac{1}{B} \sum_{b} T(x, \Theta_b)$$

donde $T(x, \theta_b)$ es un árbol *aleatorizado*.

Θ_b representa **dos** fuentes de aleatoriedad:

- se entrena con una muestra bootstrap (igual que *bagging*)
- las variables para hacer particiones se seleccionan aleatoriamente

La idea es tener como base la reducción de la varianza lograda por bagging y obtener mejora a través de incorporar una fuente adicional de aleatoriedad.

Idea general: bosque

Consideremos la varianza del *promedio de B variables alearorias*

- Las variables i.i.d cada una con varianza σ^2, entonces varianza del promedio es $\frac{1}{B}\sigma^2$.
- Las variables son i.d (no independientes) con correlación ρ positiva, la varianza del promedio es

$$\rho\sigma^2 + \frac{(1-\rho)\sigma^2}{B}$$

Esto se vincula con las fuentes de aleatoriedad en RF:

- muestras bootstrap: aumentar B reduce el segundo componente
- selección de predictores: busca reducir ρ

1 Introducción

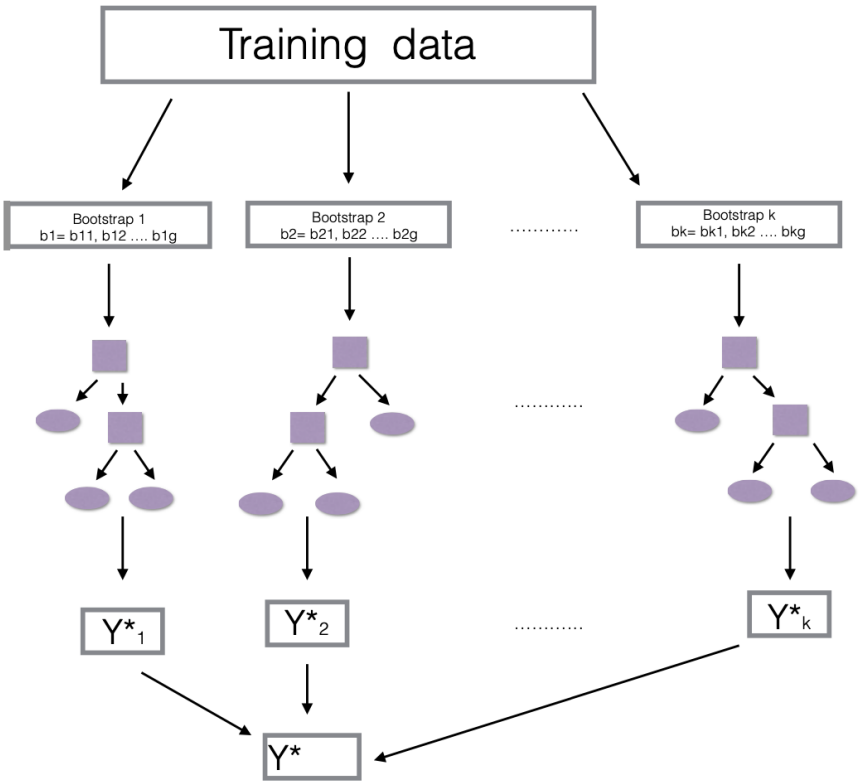
2 Bosque aleatorio: algoritmo

3 Importancia y selección de variables

4 Propiedades estadísticas

5 Variantes de Random Forest

Diagrama bosque



Algoritmo bosque

Obtengo un conjunto de árboles $\{T_b\}_1^B$. Para $b \in 1, 2, \ldots, B$

1. Selecciono muestra boostrap Z^b de tamaño N
2. Con los datos Z^b, estima árbol T_b, tipo CART, con dos modificaciones:
   - en cada nodo, solo se consideran m variables seleccionadas al azar
   - el árbol crece hasta el final sin podar

Luego:

Regresión $\quad \hat{f}_{rf} = \frac{1}{B} \sum_{b=1}^{B} T(x, \theta_b)$

Clasificación $\quad \hat{f}_{rf} = \text{argmax}_g \frac{1}{B} \sum_{b=1}^{B} I(T(x, \theta_b) = g)$

Parámetros auxiliares

Parámetros básicos: B, m y tamaño de las hojas.
Reducir m hace que se reduzca la correlación entre pares de árboles y reducir la varianza del promedio. Pero al mismo tiempo hace que cada árbol individual sea peor.

Valores por defecto:

- Clasificación, defecto $\sqrt{(p)}$ y los nodos tienen un tamaño minimo de 1.
- Regresión, defecto para m es $p/3$ y el mínimo tamaño de nodo es 5
- El valor de B depende de la implementación computacional.

Muestras OOB

- Cada muestra bootstrap tiene asociado un conjunto de datos *fuera de la muestra*, llamamos este conjunto OOB_b.
- Para cada observación $z_i = (x_i, y_i)$, se puede hacer una predicción utilizando **unicamente** los árboles T_b donde $z_i \in OOB_b$.

Definimos:

$$errOOB = \frac{1}{N}\sum_i (\hat{y}_i^{oob} - y_i)^2$$

La estimación del error OOB es casi idéntica al obtenido por un k-fold cross-validation.

Sobreajuste

- Cuando el número de variables es grande pero pocas son relevantes, RF es probable que no funcione bien si se selecciona un m chico.
- En cada partición sería pocas las chances de tener alguna variable relevante.
- RF se promociona como un método que no sobreajusta, incrementar B no causa un sobreajuste

1 Introducción

2 Bosque aleatorio: algoritmo

3 Importancia y selección de variables

4 Propiedades estadísticas

5 Variantes de Random Forest

Importancia de las variables

Medida que indica que tan importante es la variable para el método predictivo.
En cada partición de los árboles, la mejora en el criterio de partición es la medida de importancia de esa variable y se acumula en todos los árboles en el bosque.
RF también usa las observaciones OOB para otra medida de importancia que permite medir la fortaleza de cada variable para predecir (permuted importance variable).

Medida de importancia permutada

En cada árbol $T_b(x, \theta_b)$ se obtiene una medida del error individual con los datos OOB_b

$$VI_j = \frac{1}{B} \sum_b erOOB_b^* - erOOB_b$$

$erOOB_b^*$ es el error en $T_b(x, \theta_b)$ luego de permutar aleatoriamente los valores del predictor X_j.

- La importancia de X_j es mayor cuanto mayor sea el incremento en el error debido a permutar sus valores

Matriz de proximidad

- Contiene la información de que tan cercanas/similares son dos observaciones en el bosque.
- Para cada árbol, todo par de observaciones OOB que comparten el mismo nodo terminal incrementan su proximidad en uno.
- La matriz de proximidades puede ser utilizada para imputar datos faltantes.

La idea es poder visualizar usando la matriz de proximidades las observaciones que son cercanas según el bosque .

1 Introducción

2 Bosque aleatorio: algoritmo

3 Importancia y selección de variables

4 Propiedades estadísticas

5 Variantes de Random Forest

Propiedades estadísticas

¿Qué podemos decir del bosque estimado $\hat{f}(x)$ para hacer inferencia?

Más allá de tasa de error y su convergencia, hay muchos trabajos dedicados a estudiar propiedades estadísticas del método.

- Teoría asintótica (convergencia, normalidad)
- Varianzas y/o errores estandard
- Intervalos de predicción (cobertura)

Gérard Biau and Erwan Scornet. A random forest guided tour.
Test, 25(2):197–227, 2016

Convergencia y normalidad asintótica

- Difícil de trabajar con RF original.
- Modificación más común: hacer sub-muestras en vez de bootstrap.
- Wager et al. [2014], Mentch and Hooker [2016]

Algunos resultados se basan en

$$\hat{f}_{rf}(x) = \frac{1}{B}\sum_{b} T(x, \Theta_b) = \sum_{i=1}^{n} w_i(x) Y_i$$

RF vistos como promedios ponderados

Consideramos UN árbol aleatorizado

- $\ell = 1 \dots L$ las hojas del árbol (nodos terminales) y $R_\ell \subseteq S(X)$ la región que define ℓ.
- x es un nuevo dato, queremos predecir y
- La predicción de un solo árbol $T_b(x, \theta)$ para x es el promedio de las observaciones Y_i en la región $R_{\ell(x,\theta)}$.

Podemos expresar la predicción para x como

$$T_b(x,\theta) = \sum_{i=1}^{n} w_i(x,\theta) Y_i \qquad w_i(x,\theta) = \frac{1_{\{X_i \in R_{\ell(x,\theta)}\}}}{\#\{j : X_j \in R_{\ell(x,\theta)}\}}$$

RF vistos como promedios ponderados

En el **bosque** usamos el promedio de la pedicción de B árboles,

$$\begin{aligned}\hat{f}_{rf}(x) &= \frac{1}{B}\sum_b T(x,\Theta_b)\\ &= \frac{1}{B}\sum_b \sum_{i=1}^n w_i(x,\Theta_b) Y_i\\ &= \sum_{i=1}^n \left(\frac{1}{B}\sum_b w_i(x,\Theta_b)\right) Y_i\\ &= \sum_{i=1}^n w_i(x) Y_i\end{aligned}$$

donde $w_i(x) = \frac{1}{B}\sum_{t=1}^B w_i(x,\theta_t)$.

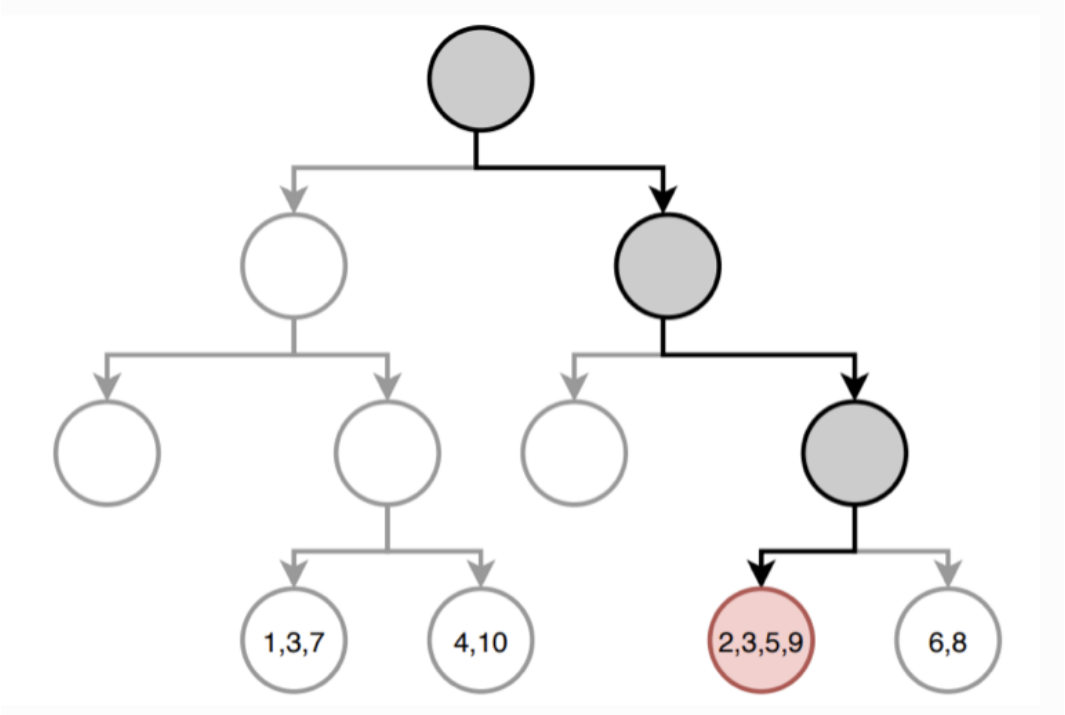
Notar que $w_i(x,\theta)$ suman 1 en cada árbol y $w_i(x)$ suma 1 en el bosque.

Ejemplo con 10 observaciones

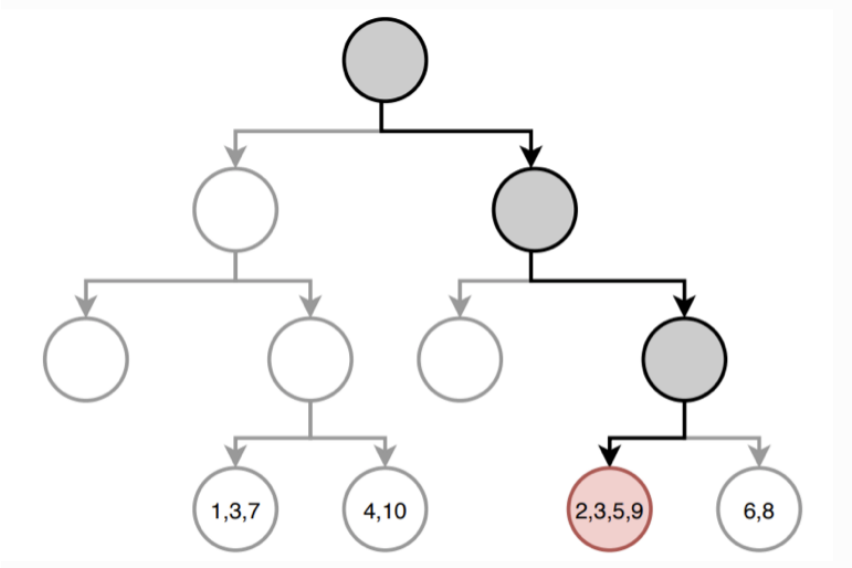
id	1	2	3	4	5	6	7	8	9	10
Y	10	18	24	8	2	9	16	10	20	14
X	...	...	...	...	...	...	...	...	...	...

Un solo árbol

Predicción con un árbol de regresión para x: el camino hasta llegar al nodo final está resaltado. En cada nodo terminal se detalla el índice de las observaciones de entrenamiento que forman parte.



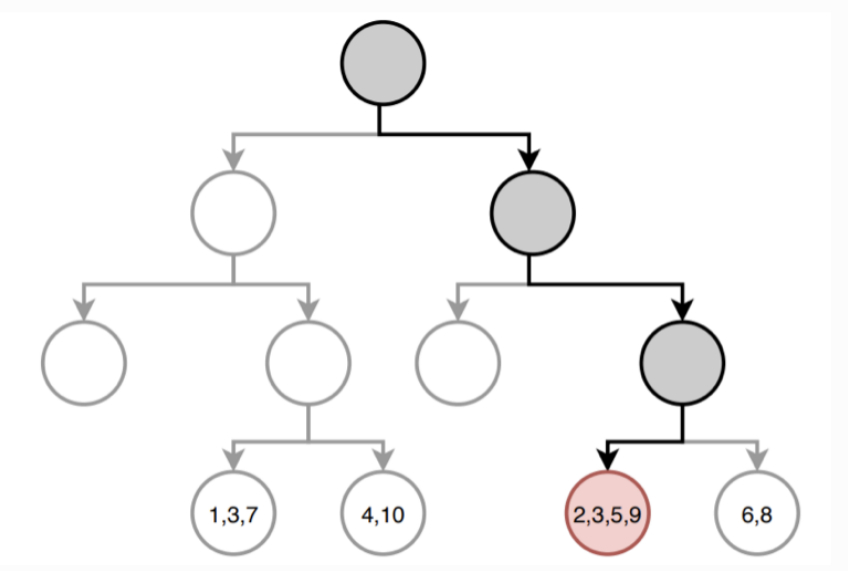
Predicción un árbol de regresión



- id: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
  Y: 10,18,24,8,2,9,16,10,20,14
- El valor predicho por el árbol es la media de la variable respuesta Y de las observaciones con id: 2, 3, 5, 9.
- $\hat{f}_{tr}(x) = \frac{18+24+2+20}{4} = 16$

Predicción un árbol de regresión

Usando la notación de ponderadores

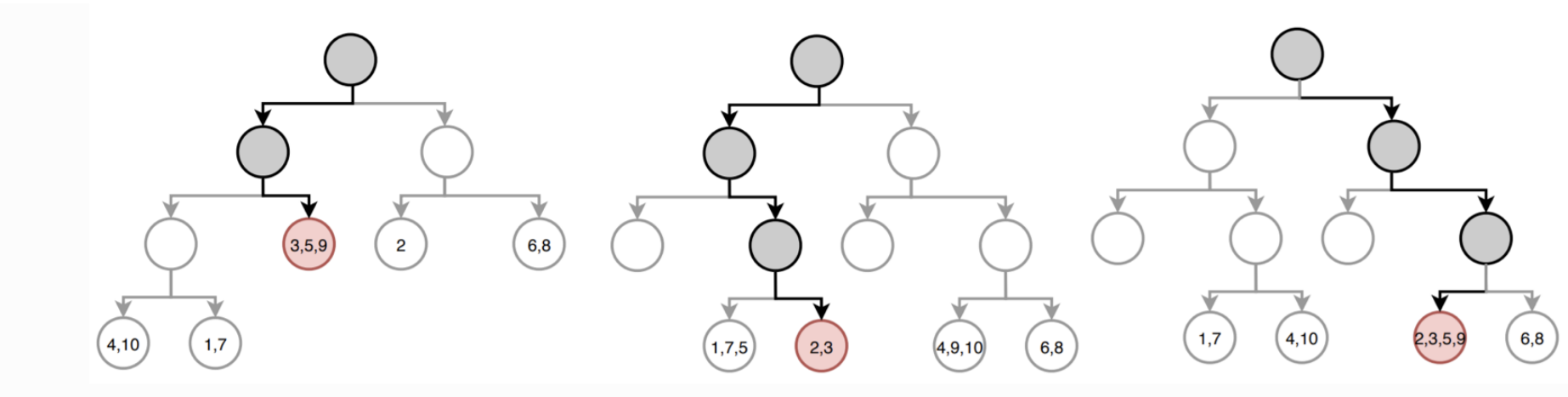


- id: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
  Y: 10,18,24,8,2,9,16,10,20,14
- El valor predicho por el árbol es la media de la variable respuesta Y de las observaciones con id: 2, 3, 5, 9.
- $\hat{f}_{tr}(x) = \sum_{i=1}^{10} w_i(x, \theta) Y_i$
- $w = (0, \frac{1}{4}, \frac{1}{4}, 0, \frac{1}{4}, 0, 0, 0, \frac{1}{4}, 0)$
- $\hat{f}_{tr}(x) = w * Y^T$

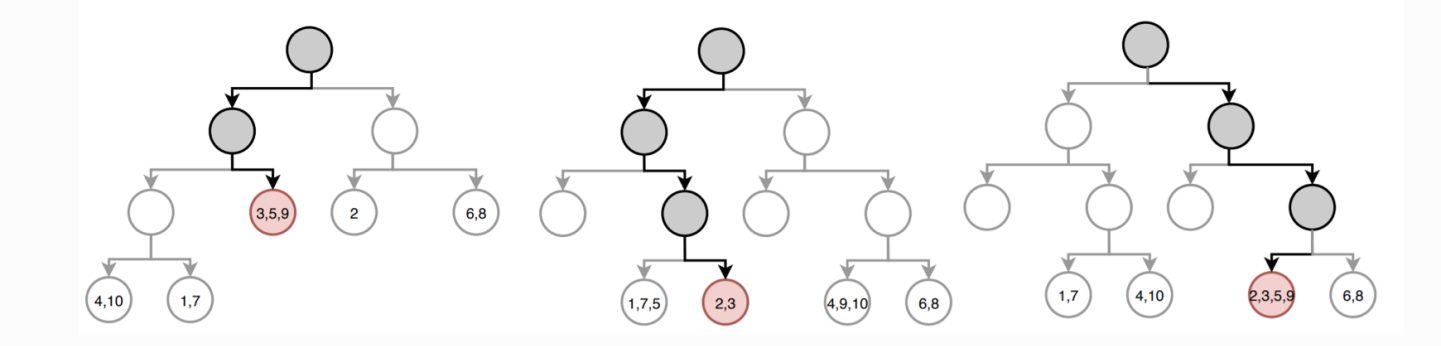
$$\hat{f}_{tr}(x) = w * Y^T = (0 \quad \tfrac{1}{4} \quad \tfrac{1}{4} \quad 0 \quad \tfrac{1}{4} \quad 0 \quad 0 \quad 0 \quad \tfrac{1}{4} \quad 0) \begin{pmatrix} 10 \\ 18 \\ 24 \\ 8 \\ 2 \\ 9 \\ 16 \\ 10 \\ 20 \\ 14 \end{pmatrix} \qquad (1)$$

Predicción del bosque

Predicción con random forest: en cada árbol, el camino hasta llegar al nodo final está resaltado. En cada nodo terminal se detalla el índice de las observaciones de entrenamiento que forman parte de él.



Bosque con tres árboles



- id: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
  Y: 10,18,24,8,2,9,16,10,20,14
- $w_{arbol1} = (0, 0, \frac{1}{3}, 0, \frac{1}{3}, 0, 0, 0, \frac{1}{3}, 0)$
- $w_{arbol2} = (0, \frac{1}{2}, \frac{1}{2}, 0, 0, 0, 0, 0, 0, 0)$
- $w_{arbol3} = (0, \frac{1}{4}, \frac{1}{4}, 0, \frac{1}{4}, 0, 0, 0, \frac{1}{4}, 0)$

- $\bar{w} = \frac{1}{3}(w_{arbol1} + w_{arbol2} + w_{arbol3})$
  $(0, \frac{1}{4}, \frac{13}{36}, 0, \frac{7}{36}, 0, 0, 0, \frac{7}{36}, 0)$

$$\hat{f}_{rf}(x) = \bar{w} * Y^T = (0 \quad \tfrac{1}{4} \quad \tfrac{13}{36} \quad 0 \quad \tfrac{7}{36} \quad 0 \quad 0 \quad 0 \quad \tfrac{7}{36} \quad 0) \begin{pmatrix} 10 \\ 18 \\ 24 \\ 8 \\ 2 \\ 9 \\ 16 \\ 10 \\ 20 \\ 14 \end{pmatrix} \qquad (2)$$

Varianzas e intervalos de predicción

- Sexton and Laake [2009] proponen usar Monte Carlo y variantes para evaluar incertidumbre en predicciones de RF
- Wager et al. [2014] usa las muestras OOB para estimar la varianza de $\hat{f}_{rf}(x)$
- Intervalos de predicción se pueden construir con los percentiles [Meinshausen, 2006]. Distribución empírica de los errores OOB [Zhang et al., 2019]

1 Introducción

2 Bosque aleatorio: algoritmo

3 Importancia y selección de variables

4 Propiedades estadísticas

5 Variantes de Random Forest

- Random survival forests [Ishwaran et al., 2008]
- Multivariate random forests [Segal and Xiao, 2011]
- Enriched random forests [Amaratunga et al., 2008]
- Quantile regression forests [Meinshausen, 2006]
- PPforest [da Silva et al., 2021]
- y más

RF con proyecciones

Natalia da Silva, Dianne Cook, and Eun-Kyung Lee. A projection pursuit forest algorithm for supervised classification.
Journal of Computational and Graphical Statistics, pages 1–21, 2021

Para problemas de **clasificación**, usa árboles de proyección (`PPtree`) en vez de CART.

- Usa bootstrap estratificado para atacar desbalance
- Usa combinaciones lineales para separar las clases
- Aprovecha la correlación entre las variables para separar grupos

Quantile RF

Nicolai Meinshausen. Quantile regression forests.
Journal of Machine Learning Research, 7(Jun):983–999, 2006 Para problemas

de regresión.

- En RF aproximamos $E(Y/X = x)$ como un promedio ponderado de las observaciones de Y.
- El promedio ponderado podría aproximar bien no solo la esperanza condicional sino toda la distribución condicional, en esto se enfoca QRF

$$F(y/X = x) = P(Y \leq y/X = x) = E(I_{\{Y \leq y\}}/X = x)$$

De forma similar a RF se puede aproximar $E(I_{\{Y \leq y\}}/X = x)$ como un promedio ponederado de las observciones $I_{\{Y \leq y\}}$

QRF propone estimar la distribución acumulada como:

$$\hat{F}(y|X=x) = \frac{1}{B}\sum_{i=1}^{B}\sum_{i=1}^{n} w_i(x,\theta) I(Y_i \leq y) = \sum_{i=1}^{n} w_i(x) I(Y_i \leq y)$$

Algoritmo:

- Mismo algoritmo que RF pero en cada hoja de cada árbol se queda con todas las observaciones no solo con el promedio
- Para cada $X = x$ se recorren todos los árboles y se calcula los pesos $w_i(x, \theta_t)$ y su promedio $w_i(x)$
- Estimo la funcion de distribución como se definió anteriormente para todo $y \in R$

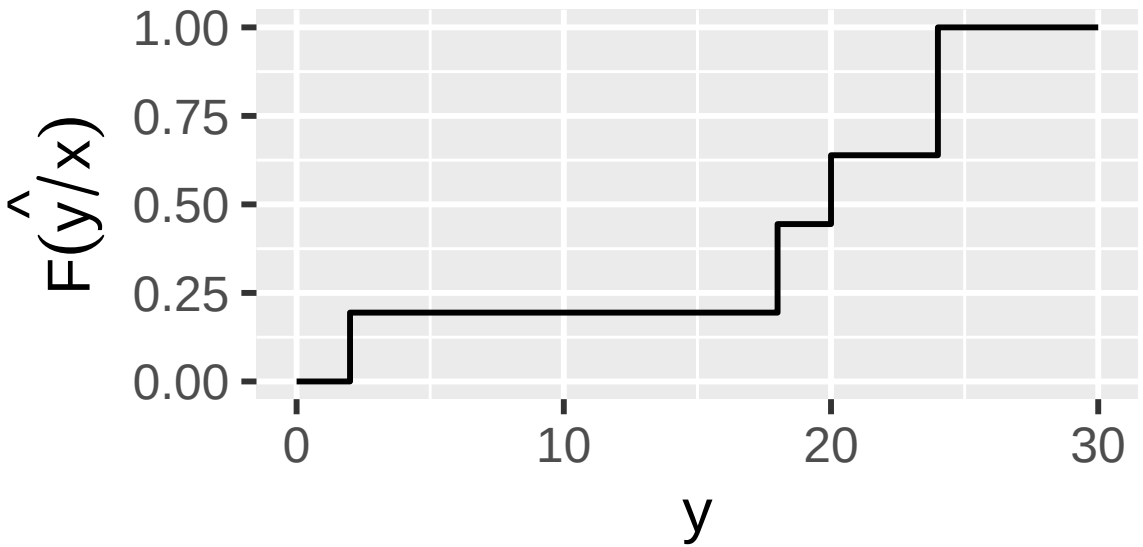
Y se estima el cuantil condicional Q_α como $\hat{Q}_\alpha = inf\{y : \hat{F}(y/X = x) \geq \alpha\}$

Bosque con tres árboles

$Y_{ordenado}$: 2, 8, 9, 10, 10, 14, 16, 18, 20, 24

$\bar{w}_{ordenado} = (\frac{7}{36}, 0, 0, 0, 0, 0, 0, \frac{1}{4}, \frac{7}{36}, \frac{13}{36})$

$$\hat{F}(y|X = x) = \sum_{i=1}^{n} w_i(x) I(Y_i \leq y)$$



- Intervalos de predicción
- regresión percentil no-parámetrica
- detección de atípicos

Dhammika Amaratunga, Javier Cabrera, and Yung-Seop Lee. Enriched random forests. *Bioinformatics*, 24(18):2010–2014, 2008.

Gérard Biau and Erwan Scornet. A random forest guided tour. *Test*, 25(2): 197–227, 2016.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Natalia da Silva, Dianne Cook, and Eun-Kyung Lee. A projection pursuit forest algorithm for supervised classification. *Journal of Computational and Graphical Statistics*, pages 1–21, 2021.

Hemant Ishwaran, Udaya B Kogalur, Eugene H Blackstone, and Michael S Lauer. Random survival forests. *The annals of applied statistics*, 2(3): 841–860, 2008.

Nicolai Meinshausen. Quantile regression forests. *Journal of Machine Learning Research*, 7(Jun):983–999, 2006.

Lucas Mentch and Giles Hooker. Quantifying uncertainty in random forests via confidence intervals and hypothesis tests. *The Journal of Machine Learning Research*, 17(1):841–881, 2016.

Mark Segal and Yuanyuan Xiao. Multivariate random forests. *Wiley interdisciplinary reviews: Data mining and knowledge discovery*, 1(1):80–87, 2011.

Joseph Sexton and Petter Laake. Standard errors for bagged and random forest estimators. *Computational Statistics & Data Analysis*, 53(3):801–811, 2009.

Stefan Wager, Trevor Hastie, and Bradley Efron. Confidence intervals for random forests: The jackknife and the infinitesimal jackknife. *The Journal of Machine Learning Research*, 15(1):1625–1651, 2014.

Haozhe Zhang, Joshua Zimmerman, Dan Nettleton, and Daniel J Nordman. Random forest prediction intervals. *The American Statistician*, pages 1–15, 2019.