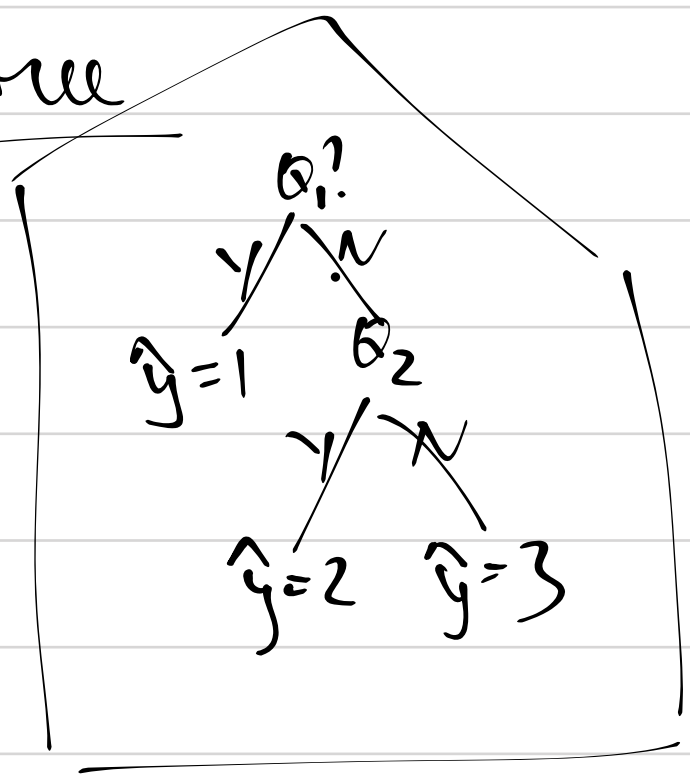
Random Forest + Data Spitting Strategies.

one tree



⊕ very interpretable
⊖ forecast quality is not very high

random forest = many trees.

Why many trees can be better?

Ex future value: Y_F ← indep of $\hat{Y}_1 \dots \hat{Y}_m$

Some independent forecasts that have the same bias; same ^ variance

$\hat{Y}_1, \hat{Y}_2, \hat{Y}_3 \dots Y_m$

a) $MSE(\hat{Y}_1) = ?$ $E((Y_F - \hat{Y}_1)^2)$

b) $MSE(\bar{Y}) = ?$ $= E((Y_F - \bar{Y})^2)$

$\bar{Y} = \frac{\hat{Y}_1 + \hat{Y}_2 + \dots + \hat{Y}_m}{m}$

$E(W^2) = Var(W) + (E(W))^2$ $Var(W) = E(W^2) - (E(W))^2$

$$MSE(\hat{Y}_1) = Var(Y_F - \hat{Y}_1) + \left(E(Y_F - \hat{Y}_1)\right)^2 =$$

$$= Var(Y_F) + Var(\hat{Y}_1) + \left(E(Y_F) - E(\hat{Y}_1)\right)^2$$

$$MSE(\bar{Y}) = Var(Y_F - \bar{Y}) + \left(E(Y_F - \bar{Y})\right)^2 =$$

$m \uparrow \Rightarrow \downarrow Var(\bar{Y})$

$$= Var(Y_F) + Var(\bar{Y}) + \left(E(Y_F) - E(\bar{Y})\right)^2$$

$$\bar{Y} = \frac{\hat{Y}_1 + \hat{Y}_2 + \dots + \hat{Y}_m}{m}$$

$$Var(\hat{Y}_1) > Var(\bar{Y}) = \frac{Var(\hat{Y}_1)}{m}$$

$$E(\bar{Y}) = E\left(\frac{\hat{Y}_1 + \dots + \hat{Y}_m}{m}\right) = E(\hat{Y}_1)$$

$$Var(\bar{Y}) = \frac{\sigma^2}{n}$$

It would be great to have m independent trees instead of one tree and just average the forecasts of all the trees!

↳ we need to introduce randomness in tree growth process.

randomness

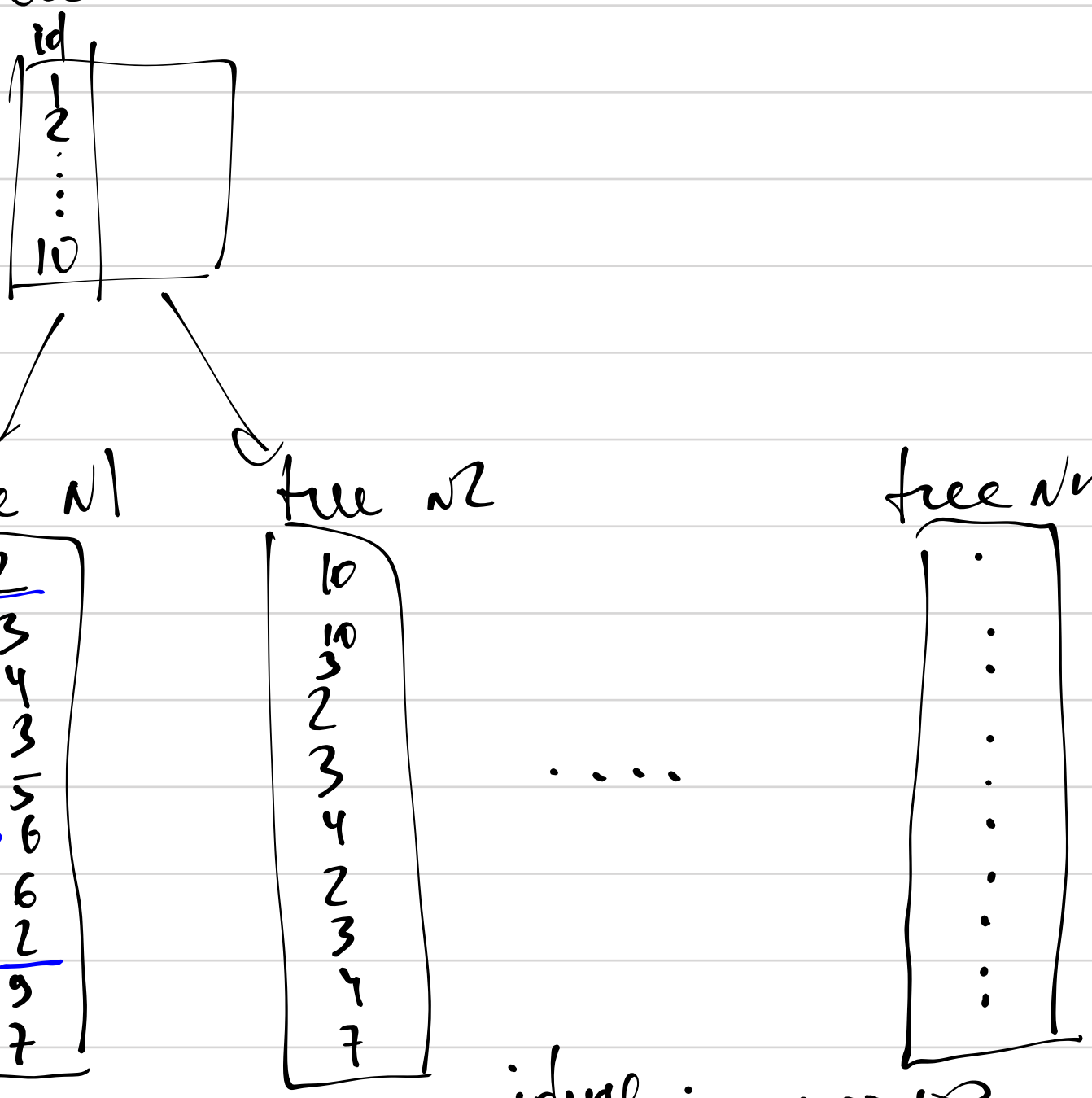
two sources of randomness

id	y_i	x_i	w_i
1	y_1	x_1	w_1
2	⋮	⋮	⋮
3	⋮	⋮	⋮
4	⋮	⋮	⋮
⋮	⋮	⋮	⋮
n	y_n	x_n	w_n

response variable

predictors / regressors / features

source N1: before growing the next tree take a sample (with replacement) of n observations out of n from original sample.



source N2

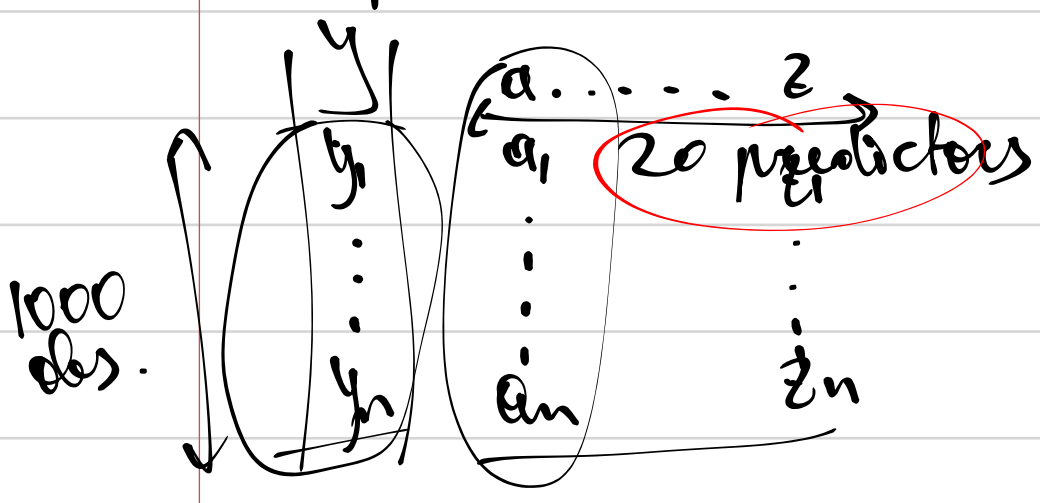
Source N2

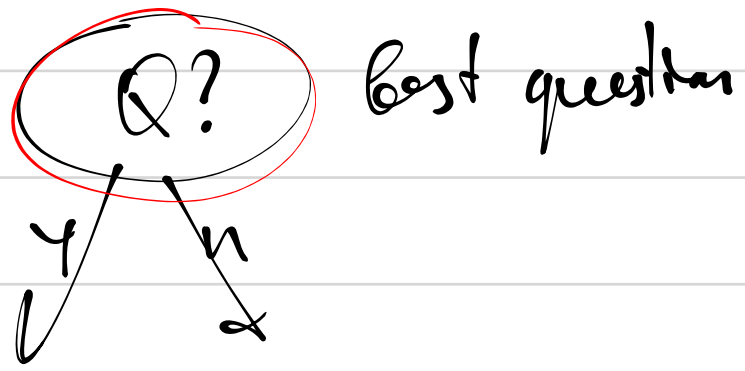
Random preselection of variables before each split.

CPU

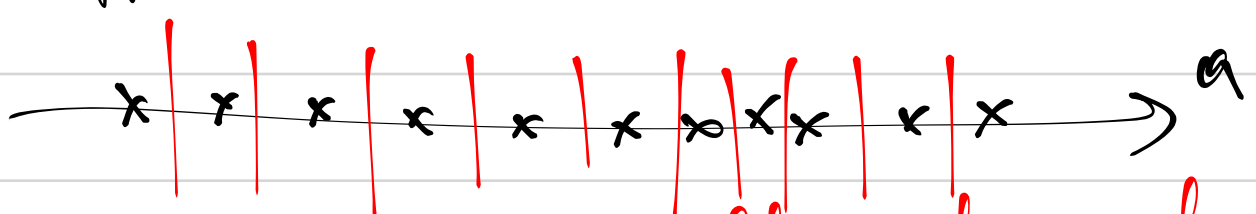
8 (16...) cores

Response





Continuous predictor => almost all values are different

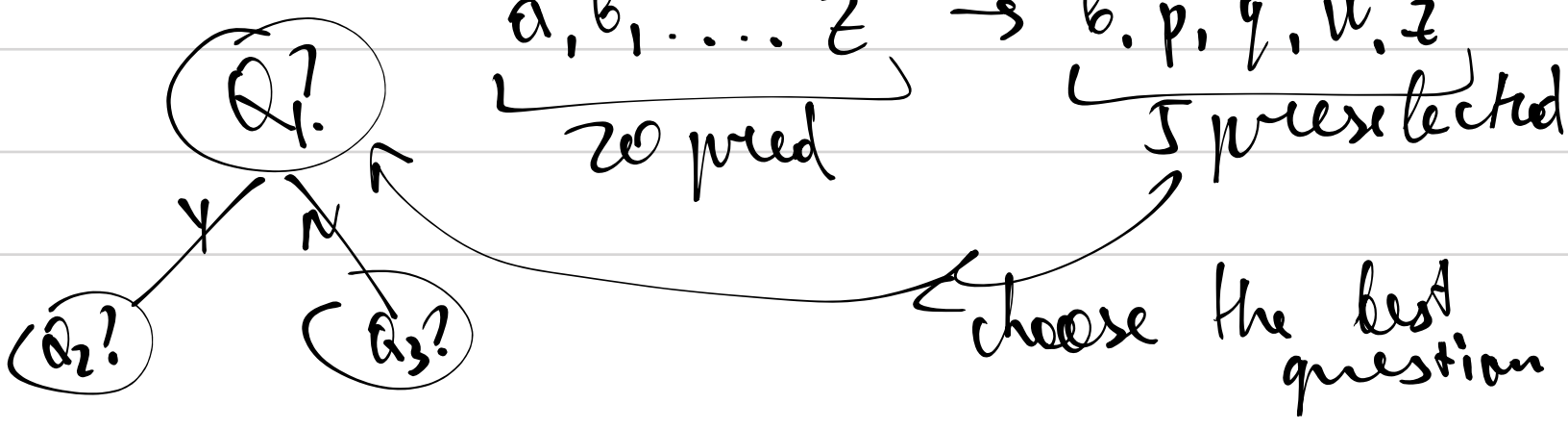


b > 7 a > 6 ... z > 20

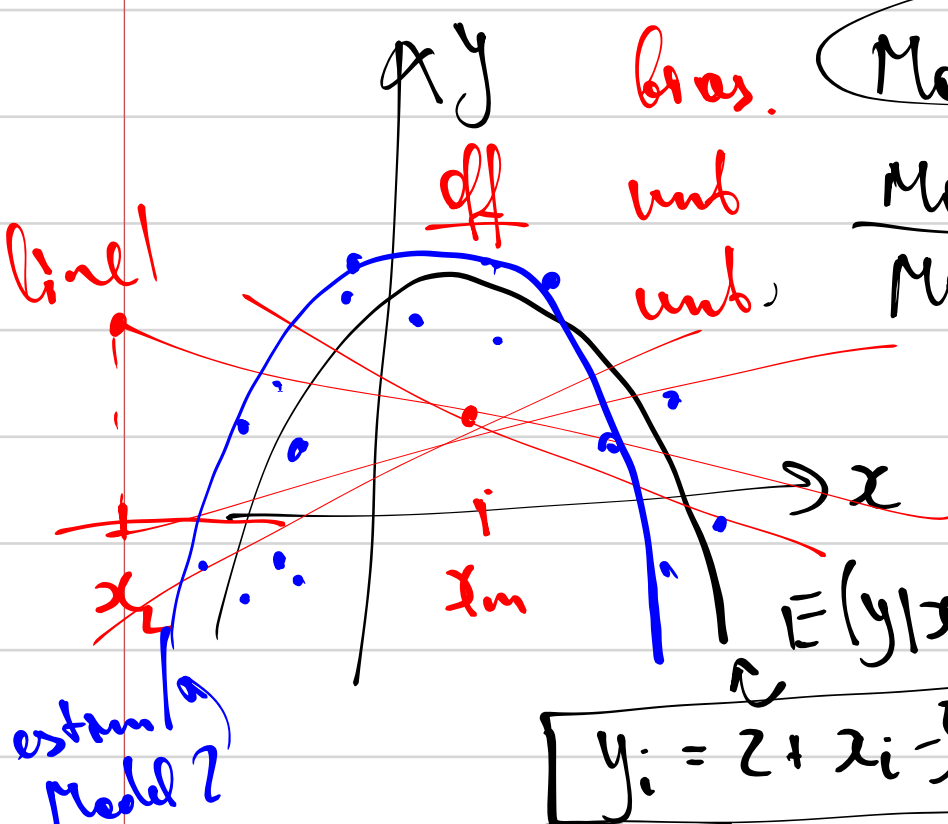
10000 trees × 100 most × 20 predictors × 1000 poss. splitting points

→ introduce randomness
→ decrease computation time

→ before each question we randomly select 5 predictors out of 20. We choose optimal question only by considering these 5 preselected predictors.



Can we do something to decrease bias?



Model 1 $\quad \hat{y}_i = \hat{\beta}_1 + \hat{\beta}_2 \cdot x_i$

Model 2 $\quad \hat{y}_i = \hat{\beta}_1 + \hat{\beta}_2 \cdot x_i + \hat{\beta}_3 \cdot x_i^2$

Model 3 $\quad \hat{y}_i = \hat{\beta}_1 + \hat{\beta}_2 \cdot x_i + \hat{\beta}_3 \cdot x_i^2 + \hat{\beta}_4 \cdot x_i^3$

$\approx$

$y_i = 2 + x_i - 3x_i^2 + u_i$

↑ TRUE dep.

with unbiased estimators?

a) which model will produce unbiased $\hat{\beta}_i$?

b) which model has lowest varrance of estimators?

---

moral: to decrease bias grow the biggest possible trees.

---

normally stopping criterion for random forest looks like: grow the tree until it is possible (until in each node there are no more than 5 observations).

$\downarrow \left(E(Y_F - \hat{Y})\right)^2 \Rightarrow$ long- long trees

$\downarrow Var(\hat{Y}) \Rightarrow$ big number of trees.

random forest

⊕ good quality of forecasts (even without tuning)

⊖ almost no interpretation

How to measure importance of variables?

permutation test.

idea: if y does not depend on a particular predictor x then changing the order of observations in x should not affect the forecast quality.

$$\begin{pmatrix} y_1 & x_1 \\ y_2 & \vdots \\ \vdots & \vdots \\ y_n & x_n \end{pmatrix} \longrightarrow \begin{pmatrix} y_1 & x_3 \\ \vdots & x_1 \\ \vdots & \vdots \\ y_n & x_j \end{pmatrix}$$

→ shuffle the predictor x 100 times
for each shuffle calculate the quality of the forecasts, calculate the average drop of the forecast quality.

drop of MSE due to shuffling.