

# Bagging and Random Forests

*K. Gibert<sup>(1)</sup>*

*(1)Department of Statistics and Operation Research*

*Knowledge Engineering and Machine Learning group*

*Universitat Politècnica de Catalunya, Barcelona*

*karina.gibert@upc.edu*

*<https://www.eio.upc.edu/homepages/karina>*

# Ensemble methods

Combine  $m$  *weak learners* to get a *strong learner*

Ensemble methods assume all  $n$  weak learners of  
a single type and different data or parameters

Higher accuracy in practice

Reduce variance

May reduce also bias

May help to reduce overfitting

Several combination schemes

(*voting, experts that abstain...*)



# Bagging

*bootstrap aggregation*

[Breiman, 1994]

Divide  $\mathcal{I}$  in Training and Test

For M times

Bootstrap Training getting a replica S

( $S=n$  random trials from Training with replacement)

Induce DT over S

Find performance of each DT

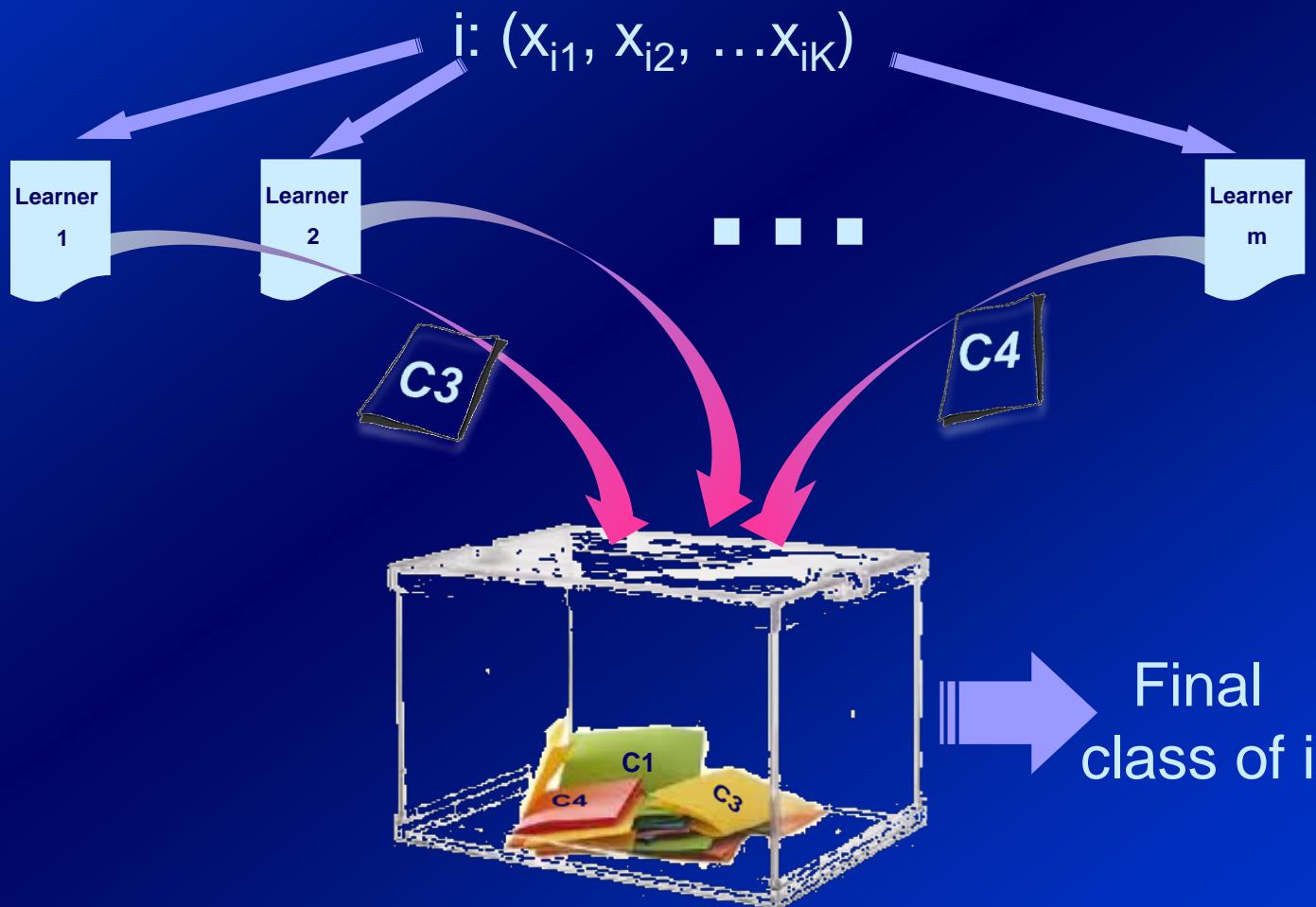
Average the M bootstrap predictions

Predict using majority vote of M learners



# Final assignment

Combine assignments proposed by  $m$  learners by voting



# Evaluate inter-learners agreement

Probability of agreement do not performs properly

Ex:  $P(C) = .95, P(C') = 0.05$

$P(\text{random agreement in two learners}) = .95^2 + .05^2 = 0.905!!$

Cohen's Kappa statistics performs better

$$\kappa = \frac{P(\text{observed agreement}) - P(\text{random agreement})}{1 - P(\text{random agreement})}$$

$$P(\text{random agreement}) = \sum_{C \in I} P(C)^2$$

$\kappa = 0$  disagreement,  $\kappa \in [0, 0.2]$  weak agreement,  $\kappa \in (0.2, 0.4]$  acceptable,  $\kappa \in (0.4, 0.6]$  moderate agg.,  $\kappa \in (0.6, 0.8]$  important agg.,  $\kappa \in (0.8, 1]$  perfect agg.

# Random Forest

[Breiman 2001]

Like Bagging, but use a *random subset* of  $K_{try}$  predictors at every split

Do not prune the  $M$  trees (*the bigger  $M$ , the better*)

Size of predictors subset: (quite robust to other sizes)

$K_{try} = \sqrt{K}$  for classification

$K_{try} = K/3$  for regression

Bagging: particular case of random forest with  $K_{try} = K$

Variance reduces more, even on smaller samples, increases accuracy

Increases efficiency as smaller sets of predictors are considered/iteration

Final prediction:

classification (maximum voted)

regression (mean predicted value)



Too small  $K_{try}$   
reduces accuracy



Too big  $K_{try}$   
Increases  
Overfitting