

Μετα-μάθηση

Meta-learning

Ensemble learning

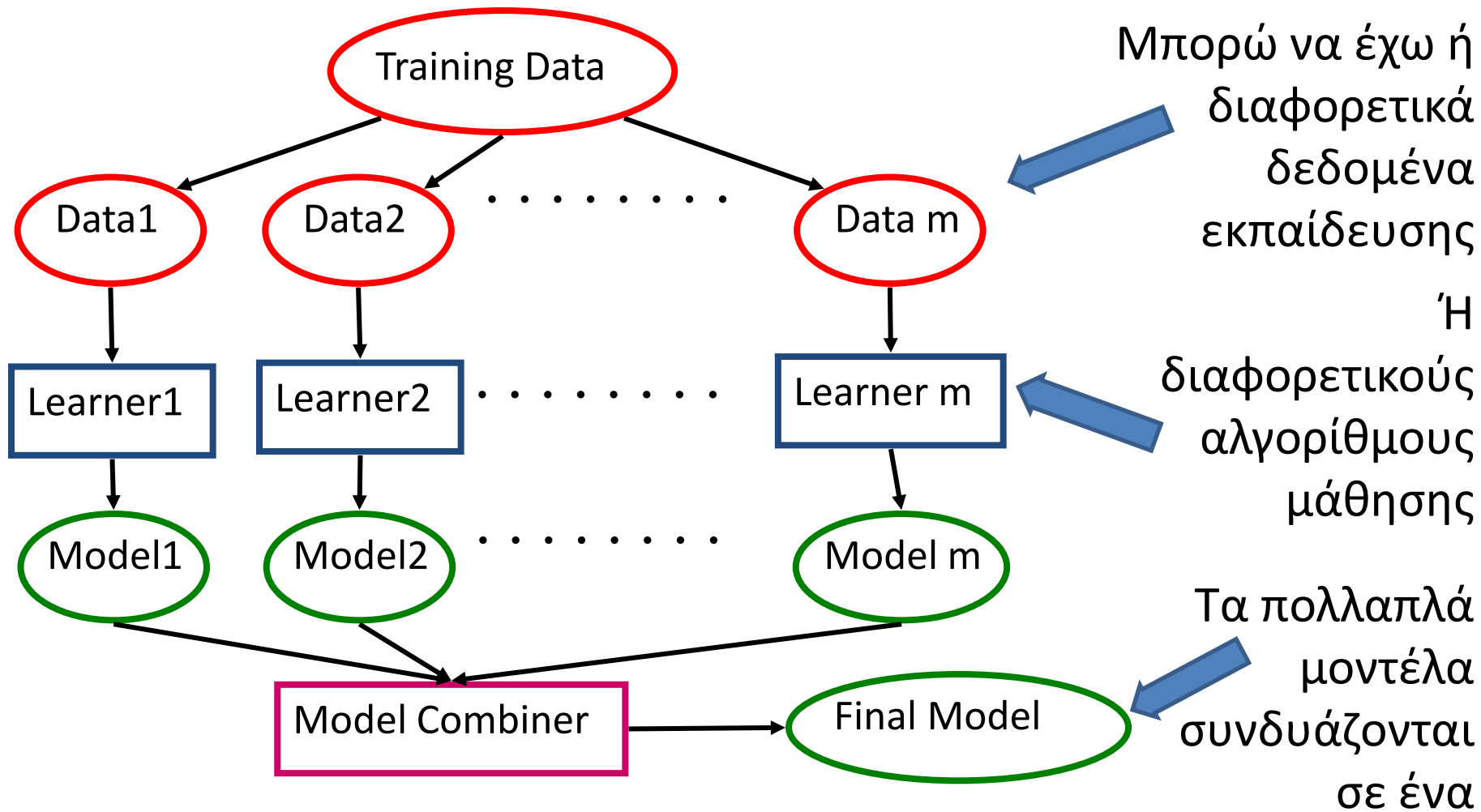
Συλλογική Μάθηση

Μετα-μάθηση

- Δημιουργία μιας συλλογής μοντέλων (ensemble of models) και συνδυασμός των προβλέψεών τους
- Παράδειγμα: Δημιουργία 100 διαφορετικών δέντρων αποφάσεων από διαφορετικά δεδομένα εκπαίδευσης και επιλογή της βέλτιστης ταξινόμησης ενός καινούριου παραδείγματος με ψήφο.
- Κίνητρο: Μείωση του σφάλματος ταξινόμησης
 - Ο συνδυασμός πολλών ανεξάρτητων αποφάσεων (όταν κάθε μια από αυτές είναι τουλάχιστον πιο ακριβής από τυχαία επιλογή) ακυρώνει πολλά λάθη και ενισχύει τις σωστές αποφάσεις
 - Ποιος θέλει να γίνει εκατομμυριούχος; (Ψήφος Κοινού)

Μετα-μάθηση

Meta-learning

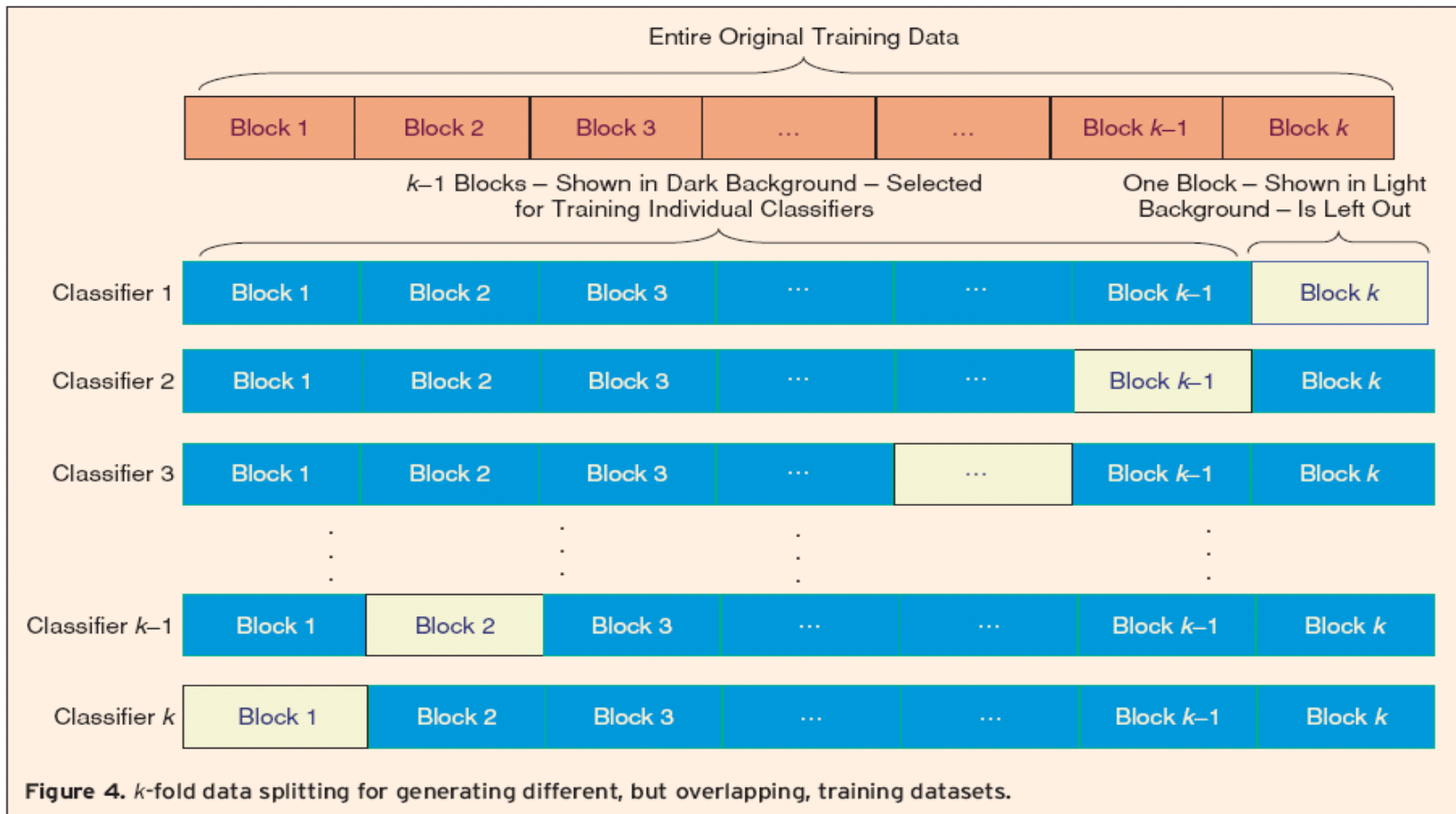


Διαφορετικά Δεδομένα – Ίδιος Αλγόριθμος

- Διασταυρωτική Αξιολόγηση
- Bootstrap Aggregating (Bagging)
- Random Forests
- Boosting

Διαφορετικά Δεδομένα – Ιδιος αλγόριθμος μάθησης

- Διασταυρωτική αξιολόγηση (cross validation)



Διαφορετικά Δεδομένα – Ιδιος αλγόριθμος μάθησης

- Bagging (Bootstrap Aggregating)
 - Εκπαίδευση
 - Έστω ότι έχω N παραδείγματα εκπαίδευσης
 - Κάνω τυχαία επαναληπτική δειγματοληψία K φορές, όπου κάθε φορά διαλέγω N παραδείγματα με αντικατάσταση (επανατοποθέτηση)
 - Δημιουργώ έτσι K σετ δεδομένων εκπαίδευσης από τα αρχικά δεδομένα εκπαίδευσης
 - Εκπαιδεύω τον αλγόριθμο μάθησης (base learner) K φορές, κάθε φορά με ένα από τα σετ εκπαίδευσης
 - Δημιουργούνται έτσι K μοντέλα
 - Αξιολόγηση
 - Τρέχω κάθε ένα από τα K μοντέλα σε κάθε παράδειγμα αξιολόγησης
 - Συνδυάζω τις K διαφορετικές εξόδους (πχ με πλειοψηφική ψήφο)
 - Σε ασταθείς αλγορίθμους μάθησης (πχ δέντρα απόφασης) που η απόδοσή τους μπορεί να αλλάζει δραματικά όταν αλλάζουν λίγο τα δεδομένα εκπαίδευσης μπορεί να μειώσει το λάθος ταξινόμησης

Διαφορετικά Δεδομένα – Ίδιος αλγόριθμος μάθησης

Bagging: Ο αλγόριθμος

model generation

Let n be the number of instances in the training data.

For each of t iterations:

 Sample n instances with replacement from training data.

 Apply the learning algorithm to the sample.

 Store the resulting model.

classification

For each of the t models:

 Predict class of instance using model.

Return class that has been predicted most often.

Figure 7.7 Algorithm for bagging.

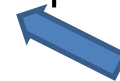
Διαφορετικά Δεδομένα – Ίδιος αλγόριθμος μάθησης

Random Forests

L. Breiman. Random forests. Machine Learning, 45(1):5-32, 2001.

- Επιλέγω τον αριθμό K των δέντρων που θέλω να σχηματίσω
- Επιλέγω τον αριθμό m των χαρακτηριστικών που θα χρησιμοποιηθούν για τον σχηματισμό των δέντρων
- Εκπαίδευση
 - Για κάθε δέντρο
 - κάνω δειγματοληψία με επανατοποθέτηση N παραδειγμάτων εκπαίδευσης (όπου N ο συνολικός αριθμός παραδειγμάτων εκπαίδευσης)
 - Σε κάθε κόμβο ρησιμοποιώ m τυχαία επιλεγμένα χαρακτηριστικά για να πραγματοποιήσω την βέλτιστη διακλάδωση
 - Δεν κλαδεύω το δέντρο που προκύπτει
- Αξιολόγηση
 - Τρέχω κάθε ένα από τα K δέντρα στο παράδειγμα αξιολόγησης
 - Εφαρμόζω πλειοψηφική ψήφο

Η διαφορά με τον
Bagging



Boosting

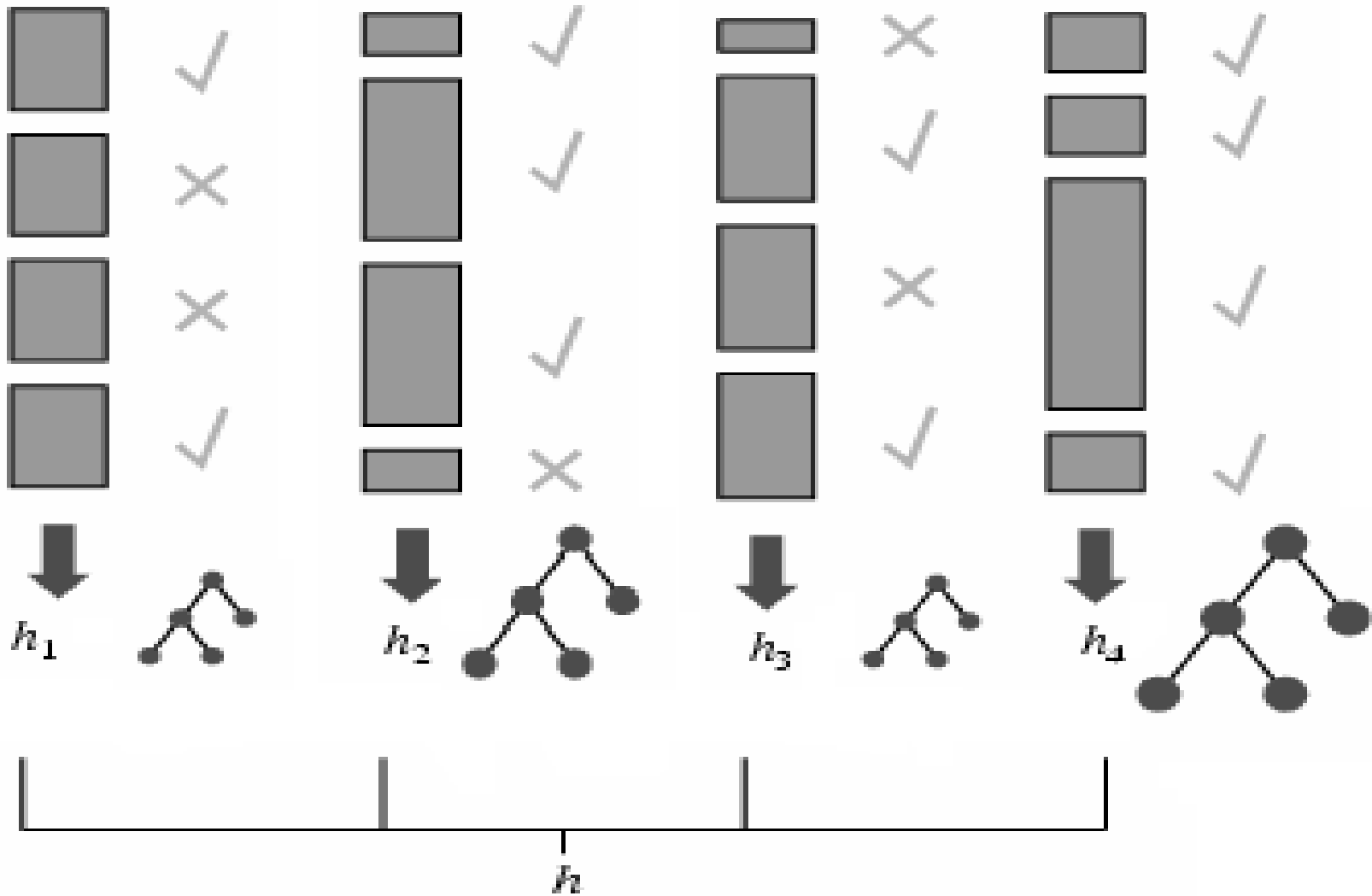
Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119-139, 1997.

- Αντί για δειγματοληψία των παραδειγμάτων, άλλαξε τα βάρη στα παραδείγματα!
- Ανάγκασε τον αλγόριθμο να εστιάσει σε παραδείγματα που στην τελευταία ταξινόμηση ταξινομήθηκαν λάθος

Boosting

- Εκπαίδευση
 - Δίνεται σε κάθε παράδειγμα ένα αρχικό βάρος, ίδιο σε όλα τα παραδείγματα
 - Σε κάθε επανάληψη
 - Δημιουργείται ένα δείγμα από τα παραδείγματα εκπαίδευσης που έχουν τα μεγαλύτερα βάρη
 - Εκπαιδεύεται ένας αλγόριθμος μάθησης με το παραπάνω δείγμα
 - Το μοντέλο που προκύπτει εφαρμόζεται στο αρχικό πλήρες σετ δεδομένων εκπαίδευσης
 - Υπολογίζεται το σφάλμα ταξινόμησης ϵ
 - Αν $\epsilon=0$ ή $\epsilon>50\%$, σταματάει ο σχηματισμός περαιτέρω μοντέλων
 - Τα παραδείγματα που ταξινομούνται λάθος θα πάρουν αυξημένο βάρος
 - Τα παραδείγματα που ταξινομούνται σωστά θα πάρουν μειωμένο βάρος
- Αξιολόγηση
 - Δώσε σε όλες τις τιμές της κλάσης ταξινόμησης βάρος 0.
 - Αν έχουν πραγματοποιηθεί t επαναλήψεις, για κάθε ένα από τα t μοντέλα προστίθεται $-\log(\epsilon/(1-\epsilon))$ στο βάρος της τιμής της κλάσης που προβλέπει το μοντέλο.
 - Η τελική τιμή της κλάσης ταξινόμησης είναι αυτή με το μεγαλύτερο βάρος.

Boosting (συνέχεια)



model generation

Assign equal weight to each training instance.

For each of t iterations:

 Apply learning algorithm to weighted dataset and store resulting model.

 Compute error e of model on weighted dataset and store error.

 If e equal to zero, or e greater or equal to 0.5:

 Terminate model generation.

 For each instance in dataset:

 If instance classified correctly by model:

 Multiply weight of instance by $e / (1 - e)$.

 Normalize weight of all instances.

classification

Assign weight of zero to all classes.

For each of the t (or less) models:

 Add $-\log(e / (1 - e))$ to weight of class predicted by model.

Return class with highest weight.

Figure 7.8 Algorithm for boosting.

Ίδια Δεδομένα – Διαφορετικοί αλγόριθμοι μάθησης

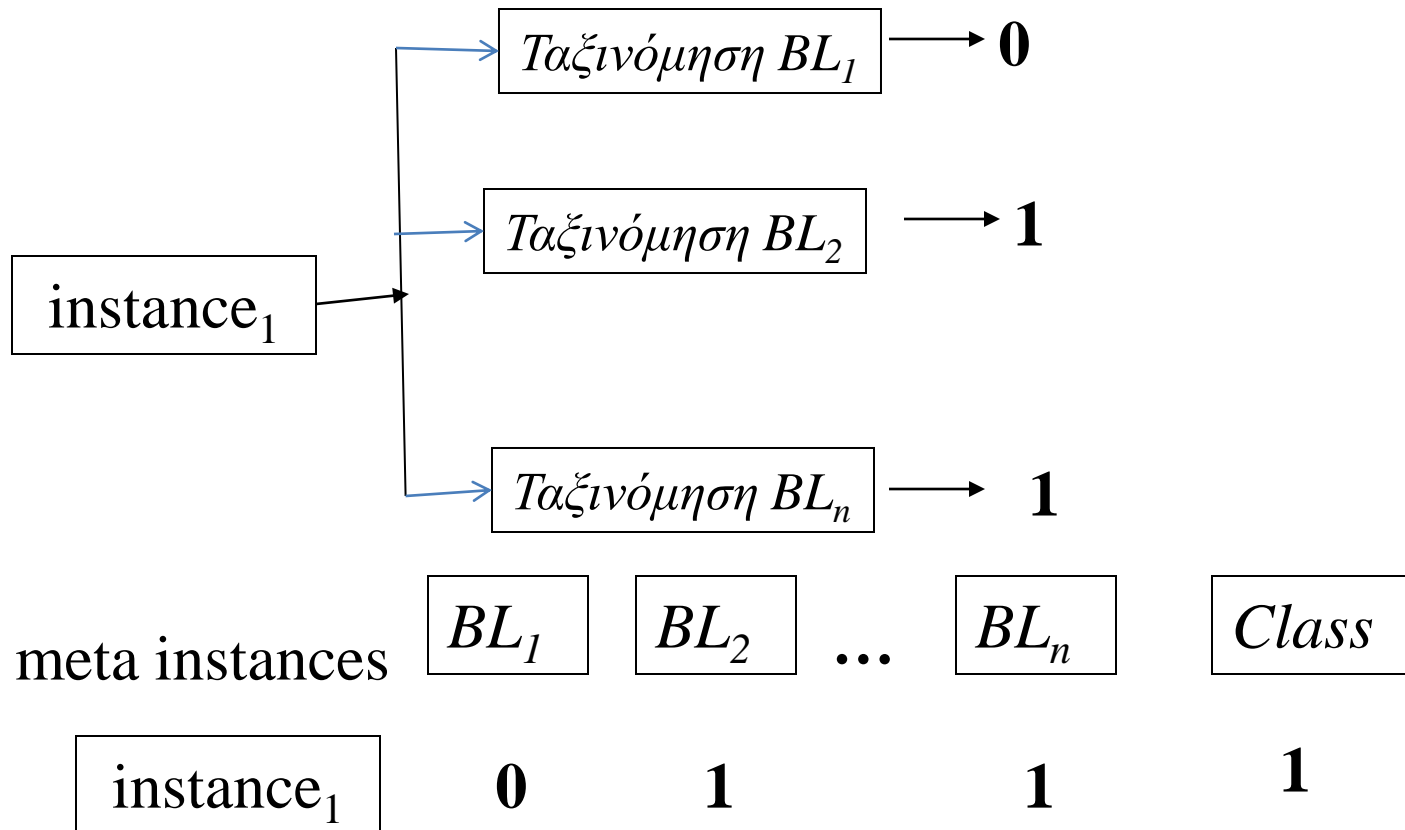
- Stacking (Stacked Generalization)

Ίδια Δεδομένα – Διαφορετικοί αλγόριθμοι μάθησης

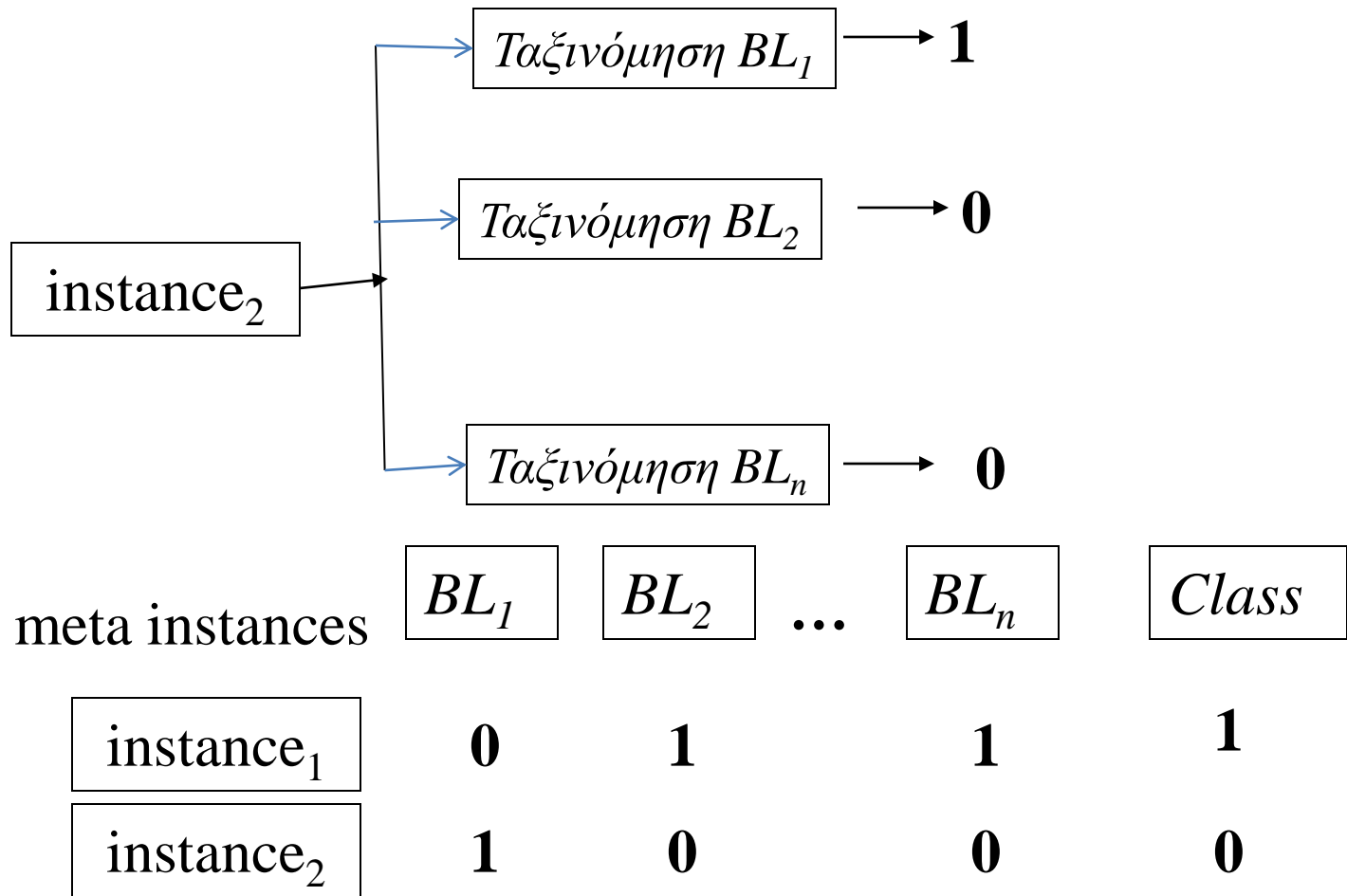
Stacking

- Όχι ψηφοφορία για την επιλογή της απόφασης
- Χρήση αλγορίθμου μάθησης για την λήψη της απόφασης
- Base learners: ταξινομητές επιπέδου 0
 - Διαφορετικοί
 - Ο καθένας βγάζει πρόβλεψη για το κάθε παράδειγμα ελέγχου
- Meta-learner: ταξινομητής επιπέδου 1
 - Κάθε παράδειγμα ελέγχου μετατρέπεται σε παράδειγμα εισόδου για τον meta-learner
 - Σαν χαρακτηριστικά έχει τις προβλέψεις των base learners για το συγκεκριμένο παράδειγμα (πολλές φορές μπορεί να συμπεριλαμβάνονται και τα αρχικά χαρακτηριστικά)

Stacking

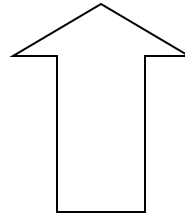


Stacking



Stacking

Meta Classifier



meta instances

BL_1

BL_2

...

BL_n

Class

instance₁

0

1

1

1

instance₂

1

0

0

0

Stacking

