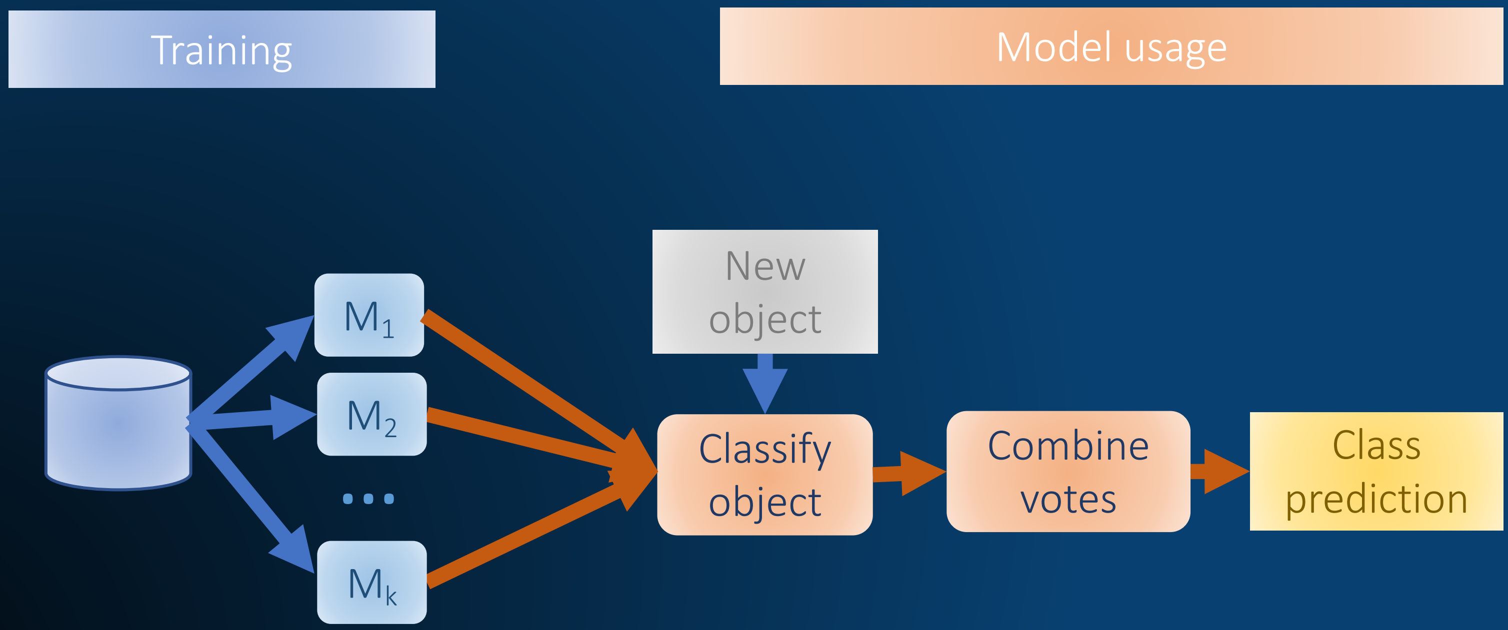
Gradient Boosting

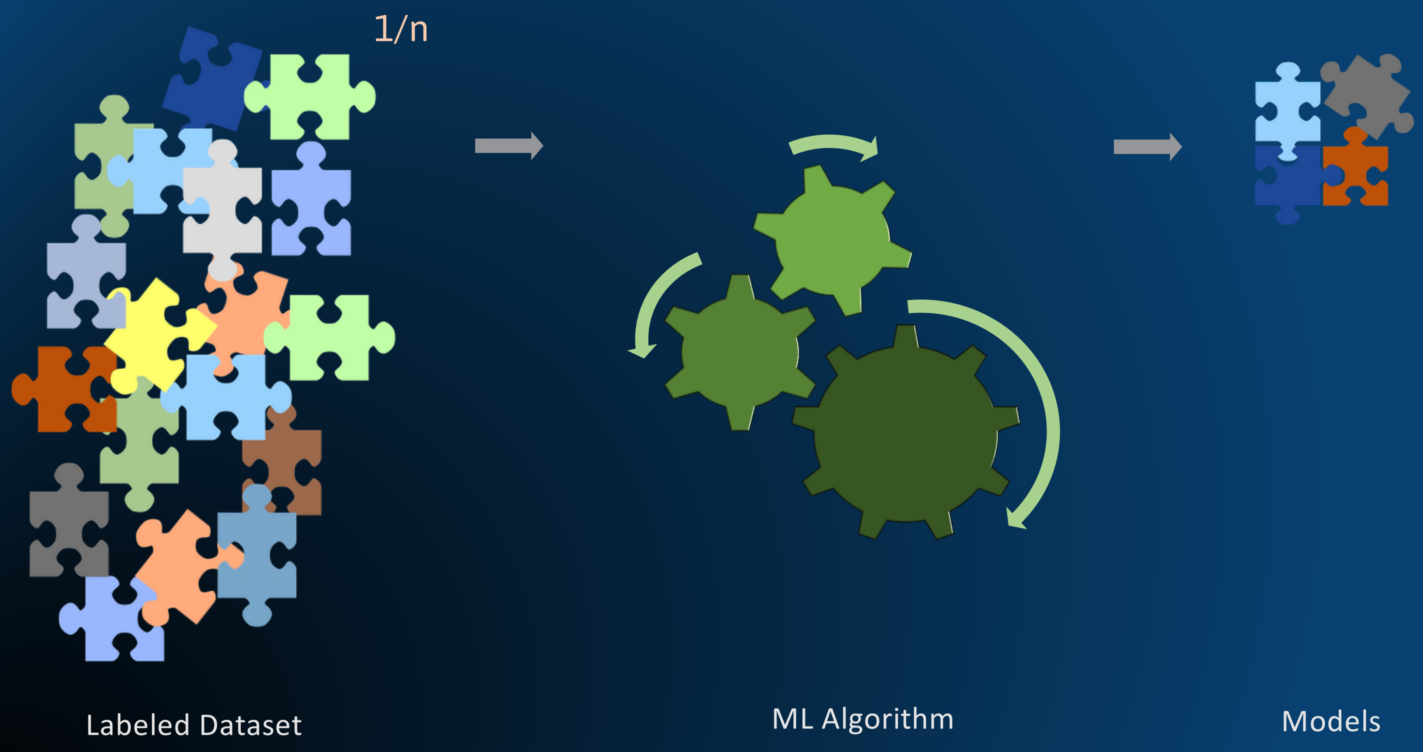
Cláudia Antunes

Instituto Superior Técnico – Universidade de Lisboa

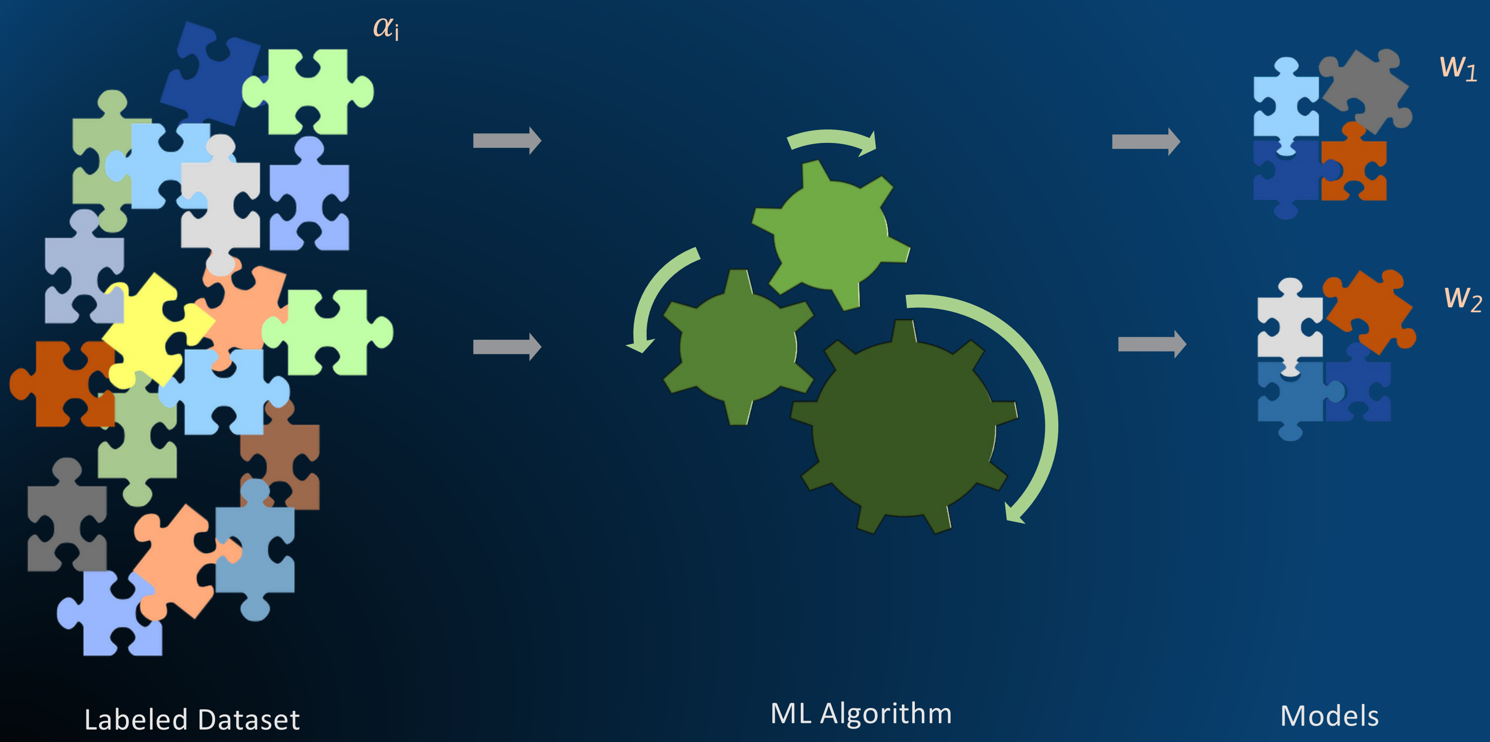
Ensemble classifier



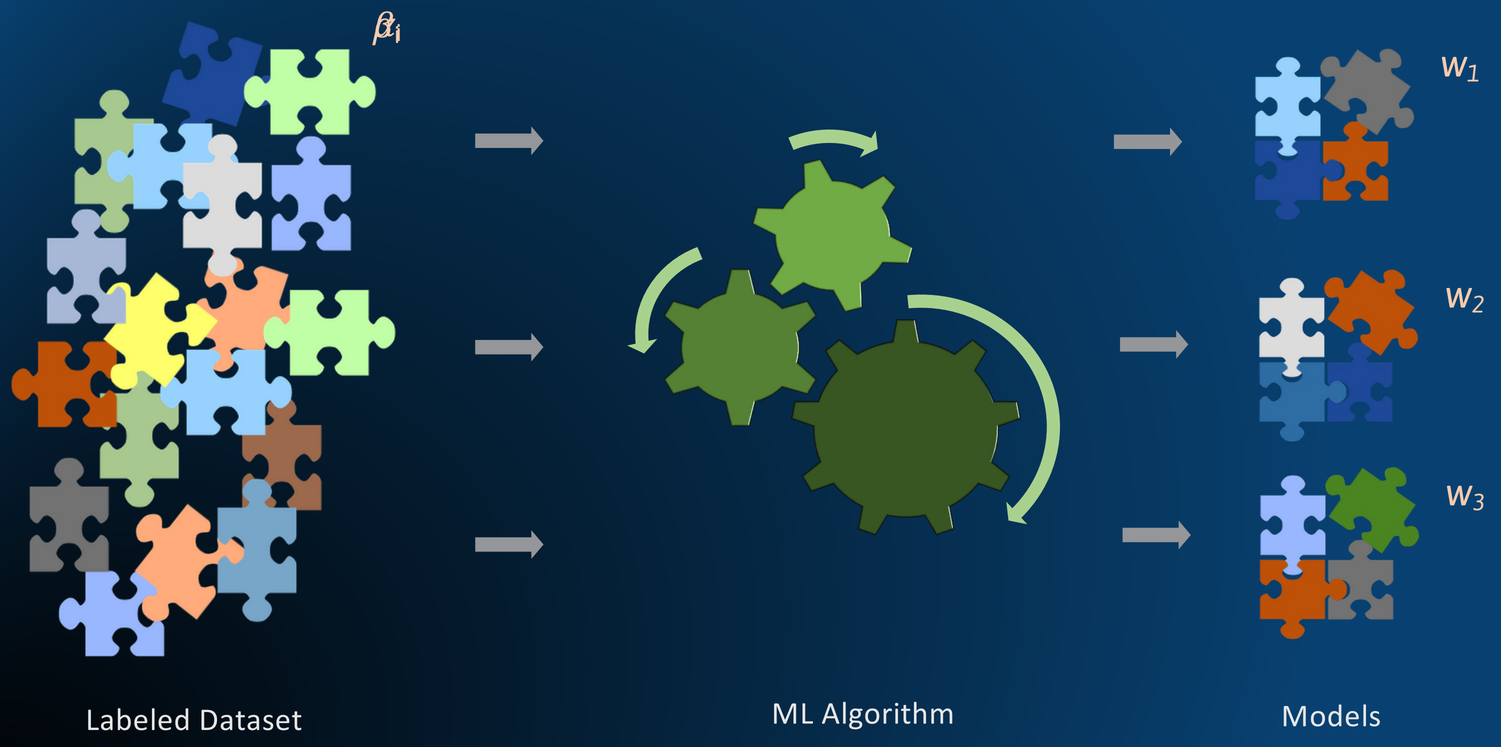
BOOSTING



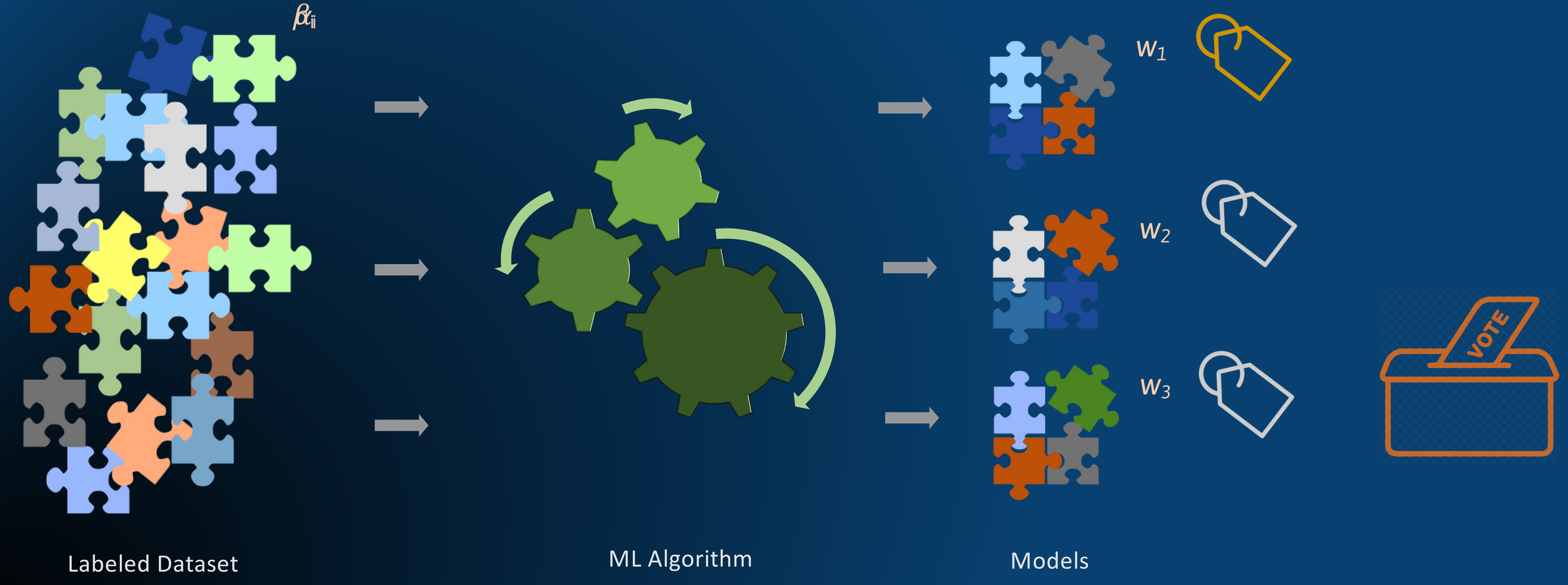
Boosting



BOOSTING



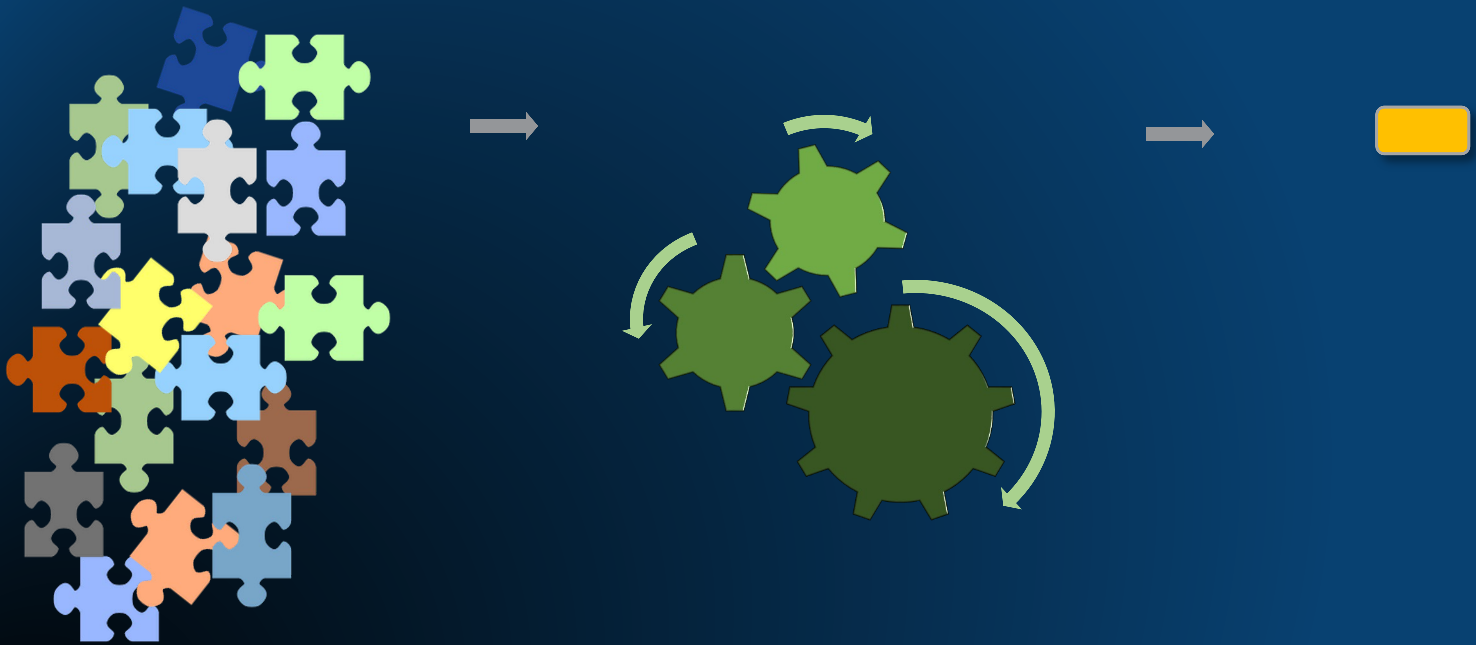
Boosting



Gradient Boosting

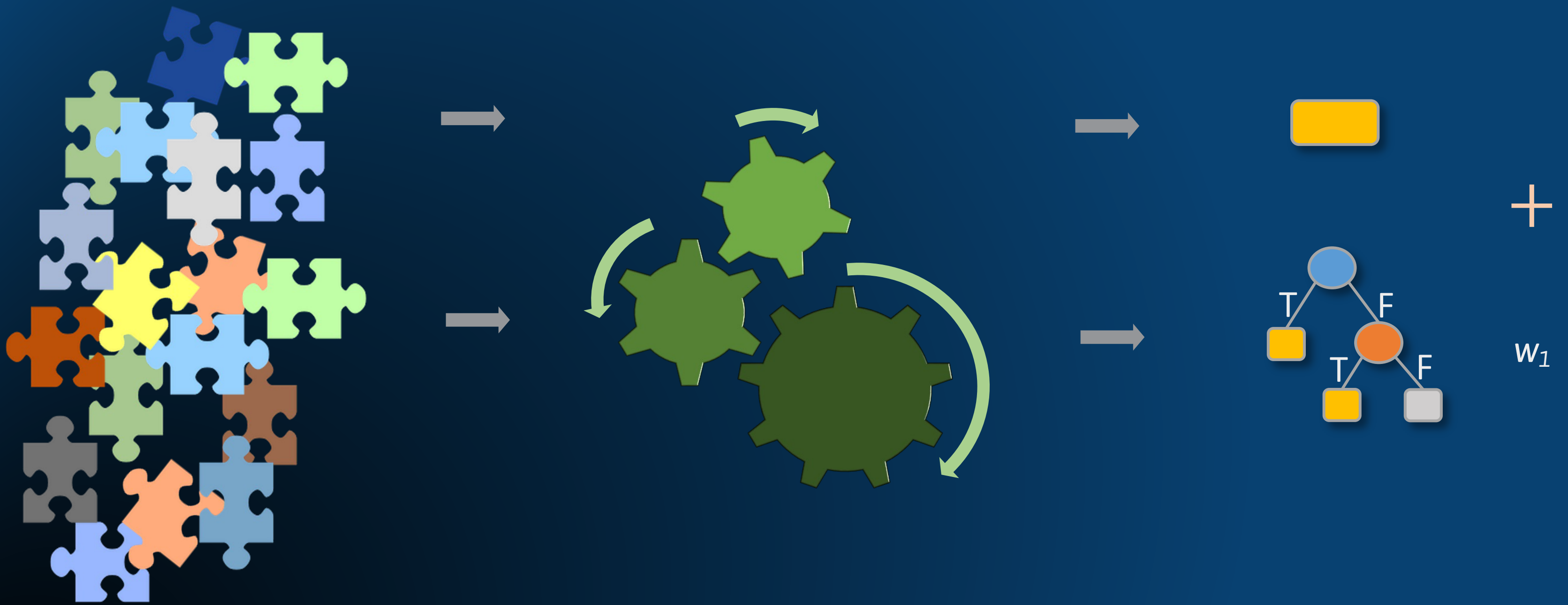
Gradient Boosting

Gradient Boosting



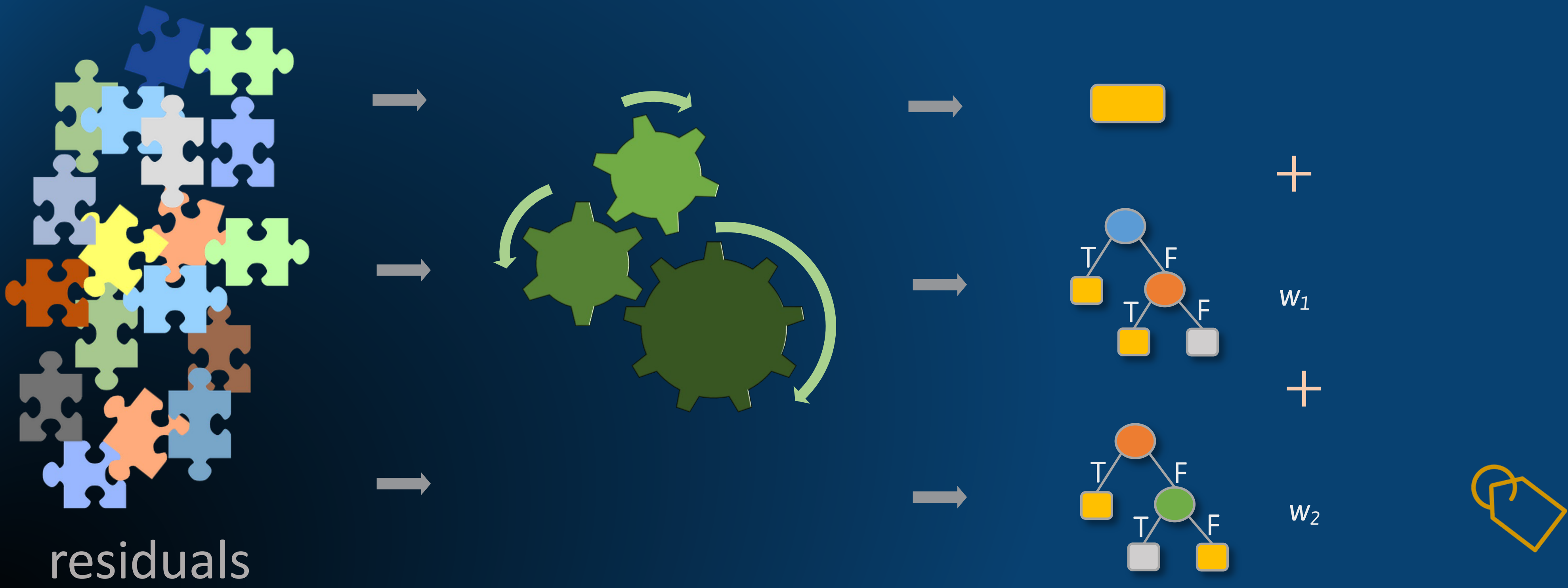
residuals

Gradient Boosting



residuals

GRADIENT BOOSTING



Diversity through gradient search

$$L(y, f(x))$$

Diversity through gradient search

$$\boxed{L(y, f(x))}$$

$$M_i = argmin_f \sum_{j=1}^{n} L(y_j, f(x_j))$$

$$\frac{\partial \sum_{j=1}^{n} L(y_j, M_i(x_j))}{\partial M_i(x_j)} = \frac{\partial L(y_j, M_j(j))}{\partial M_i(x_j)}$$

Diversity through gradient search

$$\boxed{L(y, f(x))}$$

$$\boxed{L_{MSE}\,(y, f(x)) = \frac{1}{2}(y - f\,(x))^2}$$

$$M_i = argmin_f \sum_{j=1}^{n} L(y_j, f(x_j))$$

$$\frac{\partial \sum_{j=1}^{n} L(y_j, M_i(x_j))}{\partial M_i(x_j)} = \frac{\partial L(y_j, M_j(j))}{\partial M_i(x_j)}$$

$$\frac{\partial L_{MSE}\,(y_j, M_i(x_j))}{\partial M_i(x_j)} = -(y_j - M_i(x_j))$$

$$y_j - M_i(x_j) = \begin{cases} 1, & y_j \neq M_i(x_j) \\ 0, & otherwise \end{cases}$$

residuals

Diversity through gradient search

$$\boxed{L(y, f(x))}$$

$$\boxed{L_{MSE}\,(y, f(x)) = \frac{1}{2}(y - f\,(x))^2}$$

$$M_i = argmin_f \sum_{j=1}^{n} L(y_j, f(x_j))$$

$$\frac{\partial \sum_{j=1}^{n} L(y_j, M_i(x_j))}{\partial M_i(x_j)} = \frac{\partial L(y_j, M_j(j))}{\partial M_i(x_j)}$$

$$\frac{\partial L_{MSE}\,(y_j, M_i(x_j))}{\partial M_i(x_j)} = -(y_j - M_i(x_j))$$

$$r_{ij} = -\frac{\partial L(y_j, M_i(x_j))}{\partial M_i(x_j)}$$

$$y_j - M_i(x_j) = \begin{cases} 1, & y_j \neq M_i(x_j) \\ 0, & otherwise \end{cases}$$

residuals

Models' composition

$$M_1 = argmin_f \sum_{j=1}^{n} L(y_j, f(x_j))$$

$$M_{i+1} = M_i + \eta \; argmin_f \sum_{j=1}^{n} L(y_j, M_i(x_j) + f(x_j))$$

MODELS' COMPOSITION

$$M_1 = argmin_f \sum_{j=1}^{n} L(y_j, f(x_j))$$

$$M_{i+1} = M_i + \boldsymbol{\eta}\, argmin_f \sum_{j=1}^{n} L(y_j, M_i(x_j) + f(x_j))$$

learning rate

Input

$$D = \{(x_j, y_j): 1 \leq j \leq n\},\ k, \mathrm{c},\ \eta, L(y, \mathrm{M}(x))$$

Output M_k

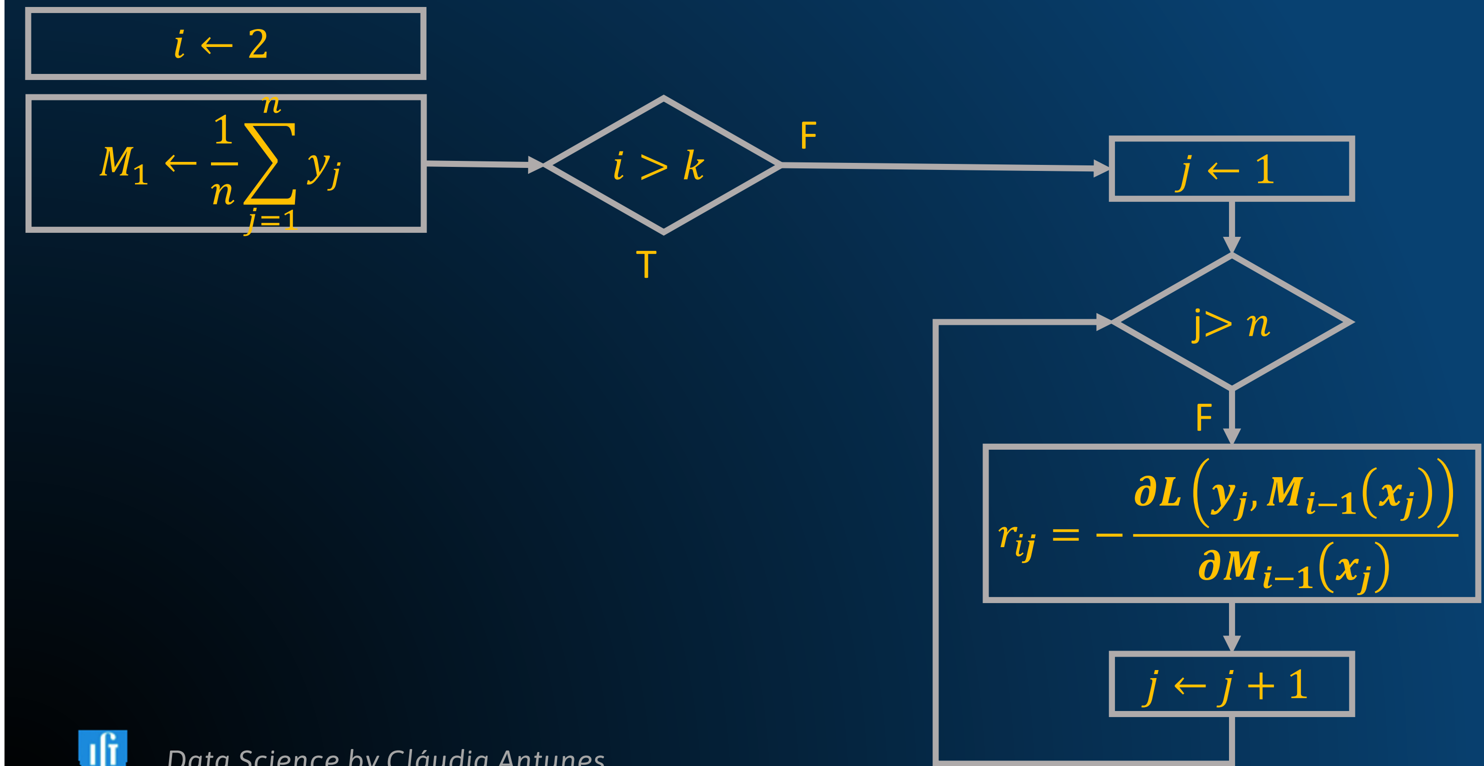
$i \leftarrow 2$

$M_1 \leftarrow \frac{1}{n}\sum_{j=1}^{n} y_j$

Input

$$D = \{(x_j, y_j): 1 \le j \le n\},\ k,\ c, \eta, L(y, M(x))$$

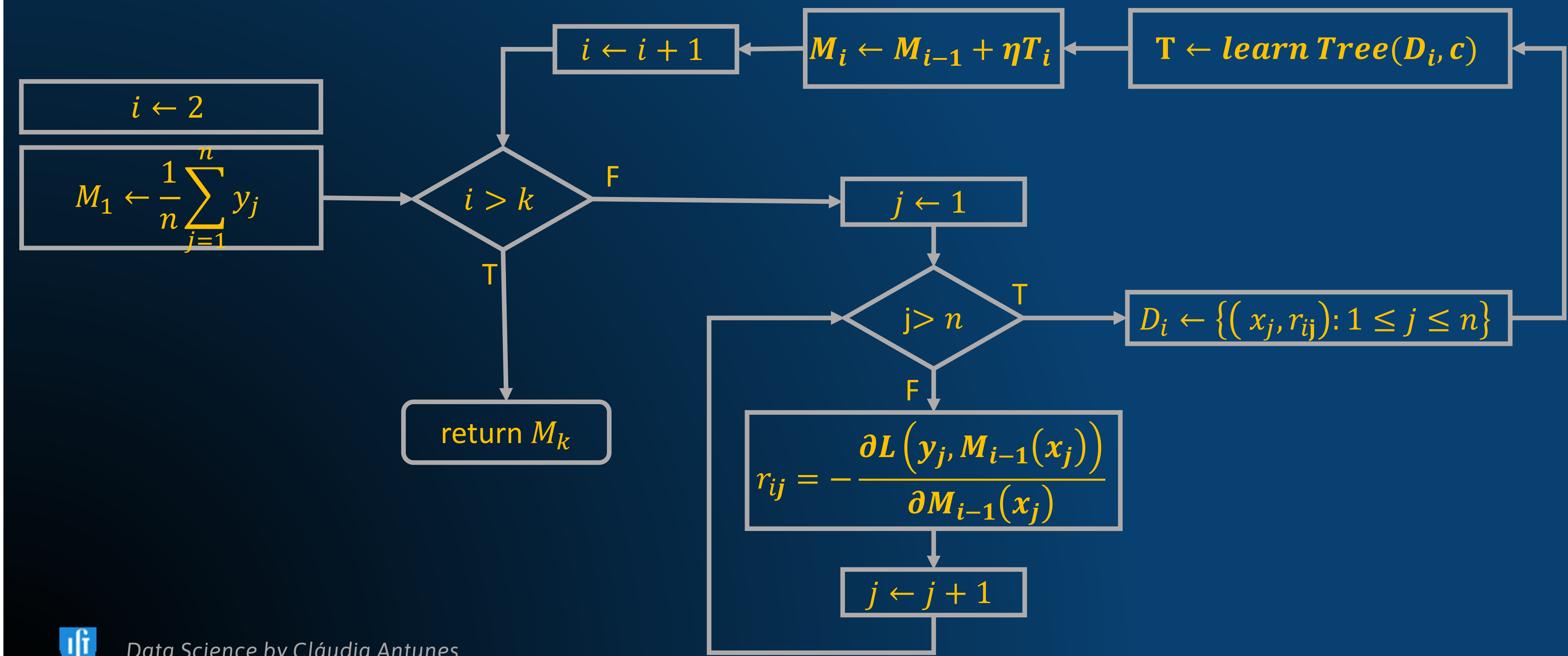
Output M_k



Input

$$D = \{(x_j, y_j): 1 \le j \le n\},\ k,\ \mathrm{c}, \eta, L(y, \mathrm{M}(x))$$

Output M_k



Gradient Boosting

Records as **conjunctions of propositions**

Training algorithm

- Learn **k** decision trees from re-labeling the dataset, according to the gradient of the loss function
- Create a model from the composition of the k learnt trees

Classification algorithm

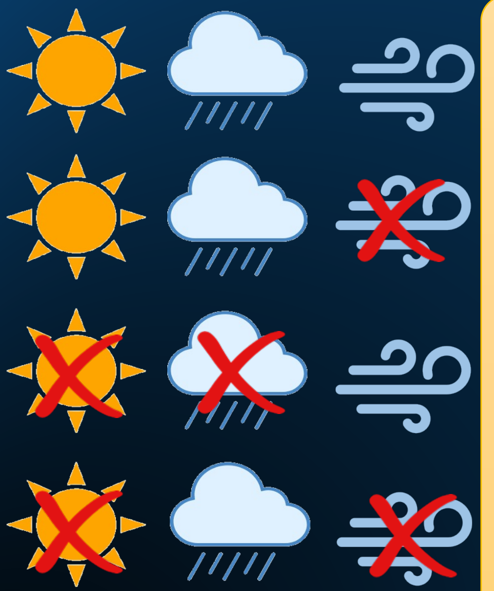
For each *Z* to be classified

- Classify *Z* according to the leaf reached by the final model

Prediction Example

Learning algorithm

Dataset



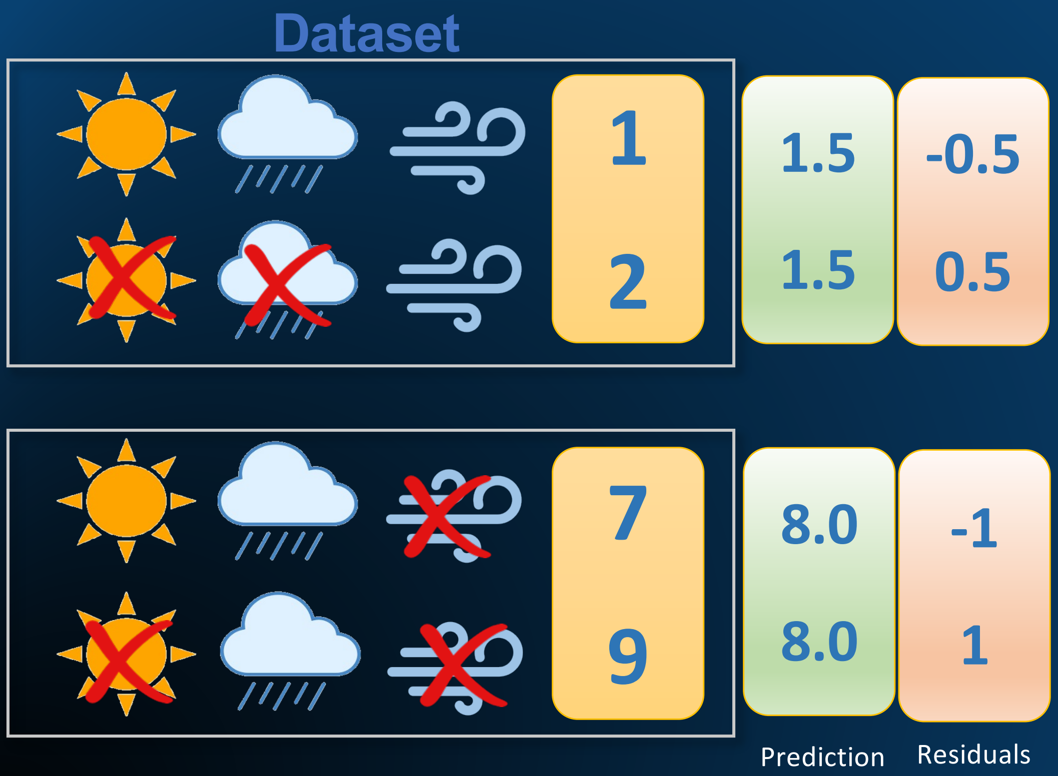
1	-3.75
7	2.25
2	-2.75
9	4.25

$\boxed{4.75}$

$$MAE = 3.25$$

Learning algorithm

LEARNING ALGORITHM



$$MAE = 0.75$$