Presentation title

Presentation subtitle

Name Surname Name Surname Name Surname

May 21, 2024

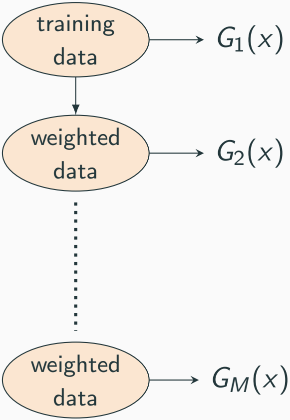
MSc in AI, University of Florence

The idea behind AdaBoost i

What are we predicting?

$\mathcal{Y} \in \{-1, 1\}$

The idea behind AdaBoost ii



$$G(x) = \operatorname{sign}\left(\sum_{m=1}^{M} \beta_m G_m(x)\right)$$

Journey to the final classifier:

- Linear combination of weak learners
- Adaptively build up complexity
- Early stopping to achieve regularization
- Re-weighting of training data

Loss function:
$L(y, f(x)) = \exp(-yf(x))$

Forward stagewise additive modeling

The general framework for boosting

Input: M, $\{(x_i, y_i)\}_1^N$

Start with $f_0(\boldsymbol{x}) = 0$;

for $m = 1$ **to** M **do**

$\quad (\beta_m, \gamma_m) = \arg\min_{\beta,\gamma} \sum_{i=1}^N L(y_i, f_{m-1}(x_i) + \beta b(x_i; \gamma))$;

$\quad f_m(\boldsymbol{x}) = f_{m-1}(\boldsymbol{x}) + \beta_m b(\boldsymbol{x}; \gamma_m)$;

end

Where $b(\boldsymbol{x}; \gamma_m) \in \mathbb{R}$ is a basis function depending on parameter γ_m

AdaBoost algorithm

Input: M, $\{(x_i, y_i)\}_1^N$

Start with $f_0(\boldsymbol{x}) = 0$;

for $m = 1$ **to** M **do**

  Compute weights $w_i^{(m)} = \exp(-y_i f_{m-1}(x_i))$;

  $G_m = \arg\min_G \sum_{i=1}^N w_i^{(m)} \mathbb{I}(y_i \neq G(x_i))$;

  Compute $\beta_m = \frac{1}{2} \log\left(\frac{1 - \mathrm{err}_m}{\mathrm{err}_m}\right)$;

  Update $f_m(\boldsymbol{x}) = f_{m-1}(\boldsymbol{x}) + \beta_m G_m(\boldsymbol{x})$;

end

Output: $G(\boldsymbol{x}) = \mathrm{sign}(f_M(\boldsymbol{x}))$

Where the weak learner $G_m \in \{-1, 1\}$ is a CART

ModelX and cross-validation

text. . .

Metrics

Model	Metric1	Metric2
Model1	0.816	0.668
Model2	0.607	0.667
Model2	0.667	0.465

T. Hastie, R. Tibshirani, and J. H. Friedman
The Elements of Statistical Learning
Springer, 6:191–199, 2009.