Supervised Learning

Boosting - Adaboost

Johannes Jurgovsky

Technische Hochschule Rosenheim

Overview

Technische Hochschule Rosenheim

Ensemble Learning

- Exploit wisdom of a crowd

Paradigms

- Stacking
- Bagging
- **Boosting**

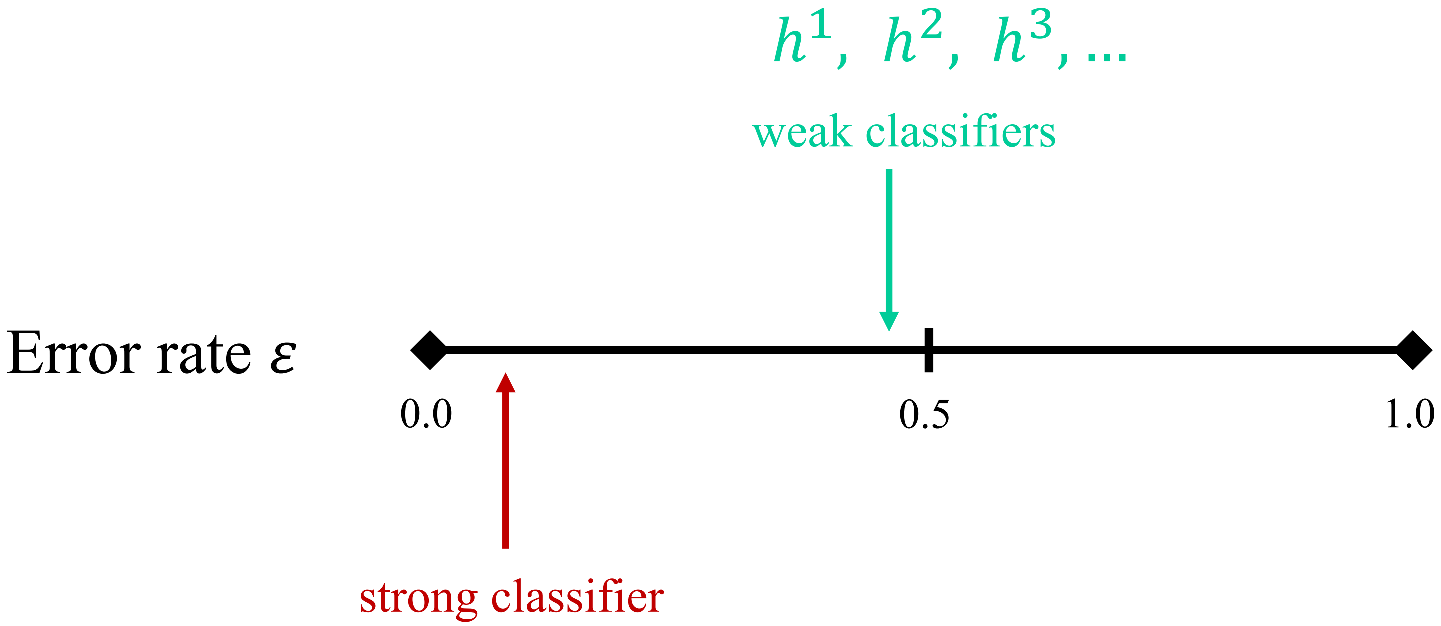
Key ideas in Boosting:

- (Arbitrary) *weak* learners
- Combine learners *sequentially*
- Assign *voting power* to each learner

- Algorithm:
  - Adaboost (*Ada*ptive *Boost*ing) : Binary classification

Weak classifier

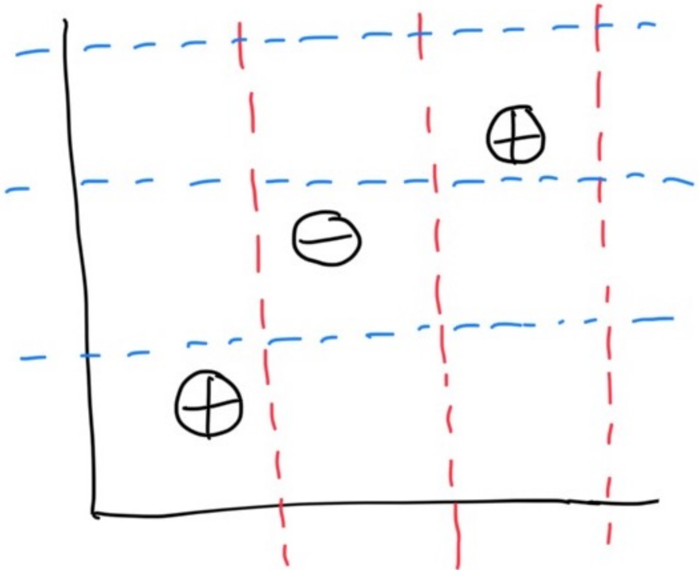
- Build strong classifier from weak classifiers



$$H(\vec{x}) = sign(\ \alpha^1 h^1(\vec{x}) + \ \alpha^2 h^2(\vec{x}) + \ \alpha^3 h^3(\vec{x}) + \cdots)$$

Weak classifier

- Decision Tree Stump
  - Decision tree with only a single level
  - One split on one feature
  - Example:
    - *N = 3* samples: {+, +, -}
    - 2 features
    - We could create 12 decision tree stumps as classifiers
    - What is the error rate of the stump corresponding to the middle red line? (assuming it predicts everything to its left as positive)



$$\varepsilon = \frac{1}{N}\#(wrong\ decisions) = \sum_{(wrong\ decisions)} \frac{1}{N}$$

weight of a sample

- Boosting works with any classifier
  - Decision Tree Stumps used for purpose of illustration

Adaboost

- Weight of a sample:
  - Emphasize previously incorrectly classified samples
  - Guide selection of classifier in the next round

  Error rate: $\varepsilon^t = \sum_{i \in (wrong)} w_i^t$

  Enforce distribution over samples: $\sum_i^N w_i^t = 1, \quad \forall t$

- Ensemble classifier: Each learner gets to vote on the final classification

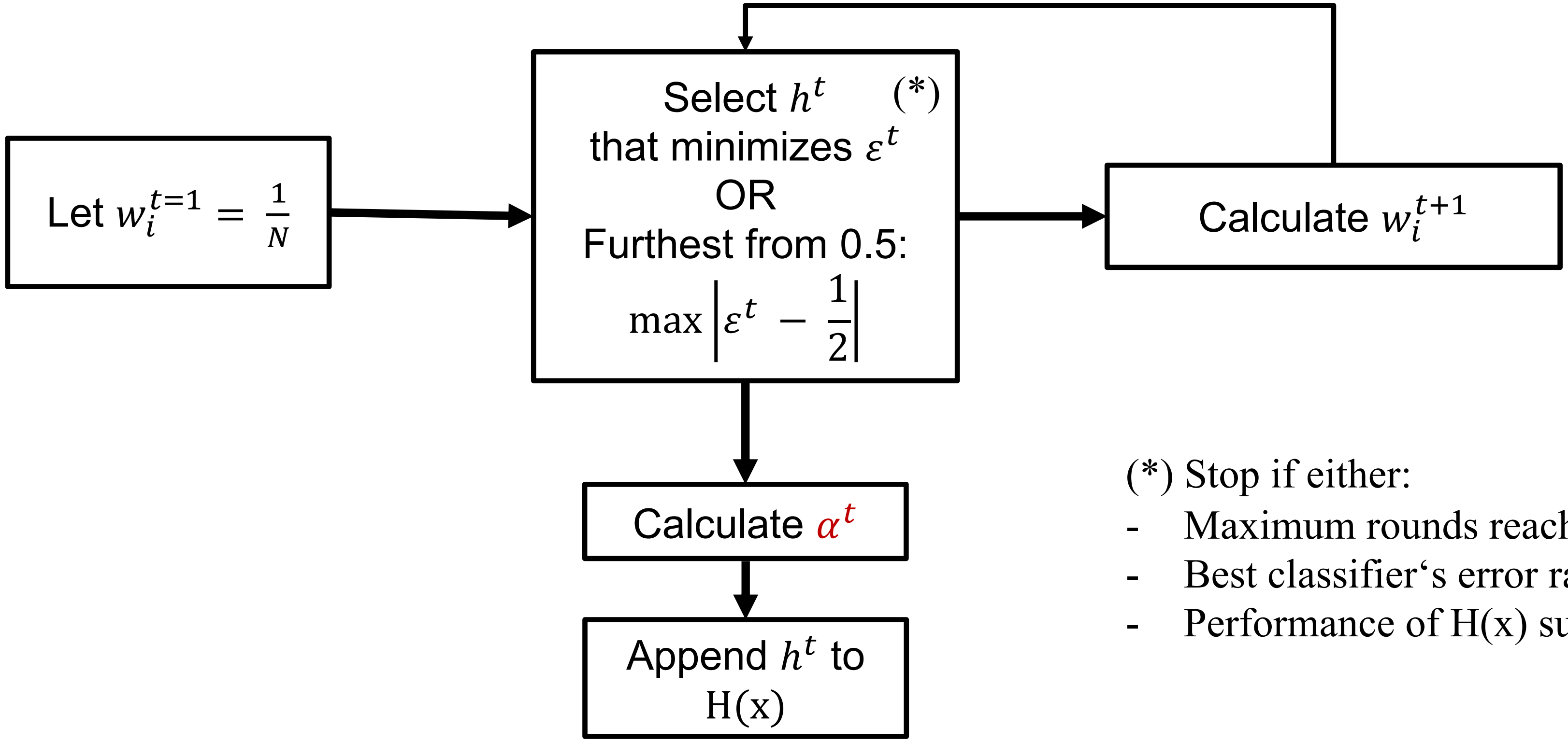
  Each classifier $h(\vec{x})$ classifies a sample as either +1 or -1

  The *voting power* α controls a classifier‘s contribution to $H(\vec{x})$

$$H(\vec{x}) = sign(\alpha^1 h^1(\vec{x}) + \alpha^2 h^2(\vec{x}) + \alpha^3 h^3(\vec{x}) + \cdots)$$

Adaboost - Process

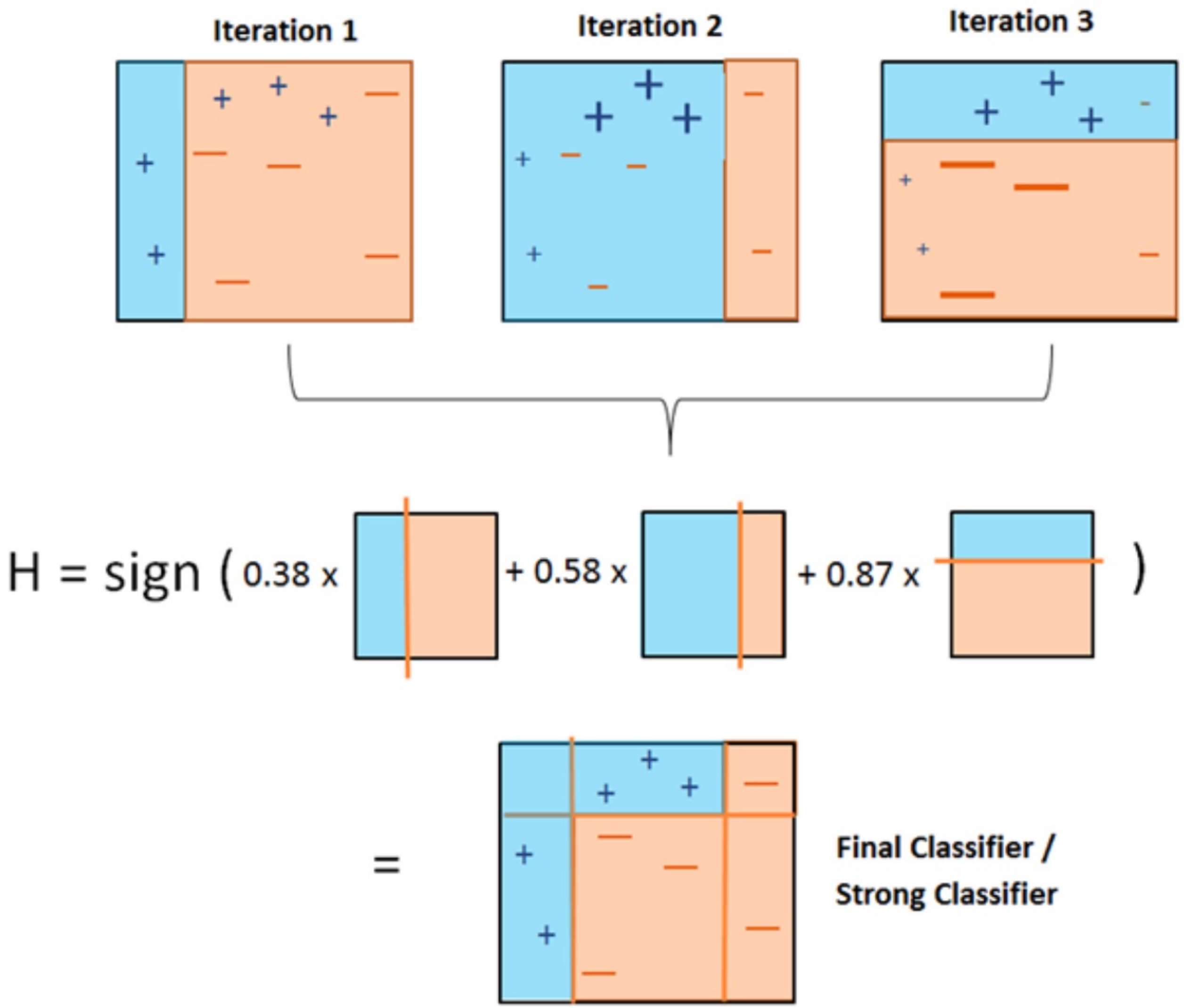
- The strong classifier $H(\vec{x})$ is assembled in a series of rounds



(*) Stop if either:

- Maximum rounds reached
- Best classifier‘s error rate is 0.5
- Performance of H(x) sufficient

Adaboost - Illustration



Source

Adaboost - Updates

- ## Weight update:

  If sample *i* was classified incorrectly, increase weight w_i for the next round.
  Otherwise, decrease weight.

  - $w_i^{t+1} = \frac{1}{Z} w_i^t \, e^{-\alpha^t h^t(\vec{x}_i) y(\vec{x}_i)}$

    Z is the normalizer s.t. $\sum_i w_i^{t+1} = 1$

- ## Voting power:

  The smaller the error rate of classifier h^t, the more (positive) voting power is assigned to classifier h^t
  At $\varepsilon^t = \frac{1}{2}$, the voting power α^t is Zero. (-> a random classifier as no voting power in the ensemble)

  - $\alpha^t = \frac{1}{2} \ln \frac{1-\varepsilon^t}{\varepsilon^t}$

- These formulas are not derived from something - but carefully constructed.
  They are mathematically convenient and have nice properties.

Adaboost – Simplification

- This particular choice of weight updates and voting power has two implications:

  - The new weights are scaled versions of the old weights with the property

$$\sum_{i\,\in(correct)} w_i^{t+1} = \frac{1}{2} \qquad\qquad \sum_{j\,\in(incorrect)} w_j^{t+1} = \frac{1}{2}$$

  - The error rate of $H(\vec{x})$ is bounded by a negative exponential.
    i.e. the error rate eventually approaches Zero as we add more classifiers.

Adaboost

Exercise

Summary

- Boosting is
  - strong
    - if the classifiers make non-overlapping errors, Boosting is guaranteed to create a perfect classifier
    - even if the classifiers do not make non-overlapping errors, Boosting may create a perfect classifier
  - versatile
    - arbitrary types of classifiers can be used
    - …weak learners are sufficient
  - robust against overfitting
  - efficient to implement

- Other Boosting algorithms:
  - Gradient Boosting: Applicable to Regression and Classification
  - XGBoost (e*X*treme Gradient *Boost*ing): Off-the-shelf highly parallelized implemention

References

- Y. Freund and E. Schapire; „A Decision-Theroetic Generalization of On-Line Learning and an Application to Boosting“, Journal of Computer and System Sciences, 55, 119-139; 1997.
- T. Hastie, R. Tibshirani and J. Friedman; „Elements of Statistical Learning“, Second Edition; Springer, 2008.

- More Links:
  - https://www.youtube.com/watch?v=UHBmv7qCey4&t=2759s