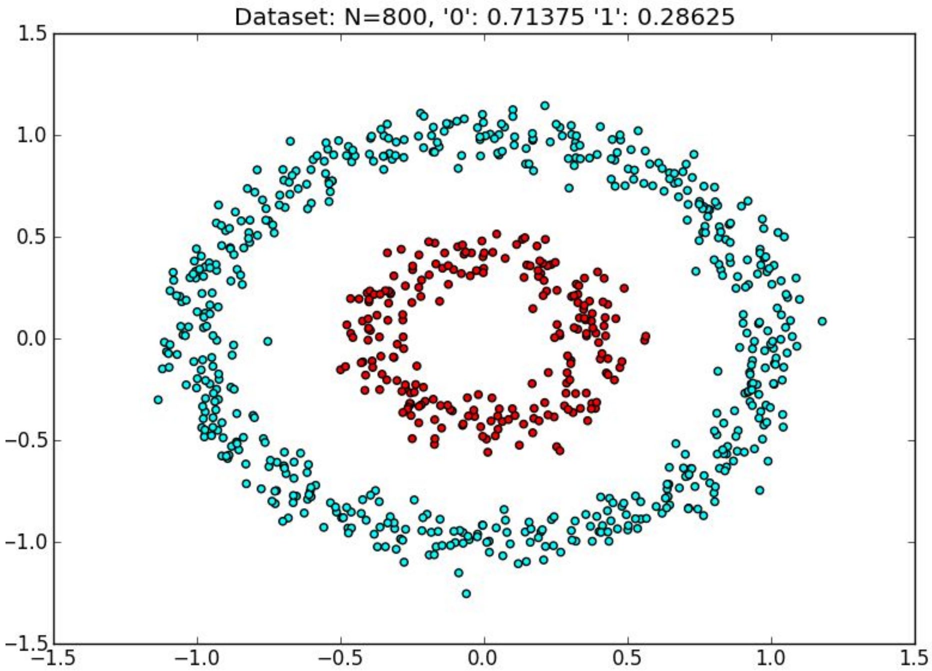
An Exploration of Boosting for Classification

N. El-Helou, K. Khalil, J. Neal

Introduction

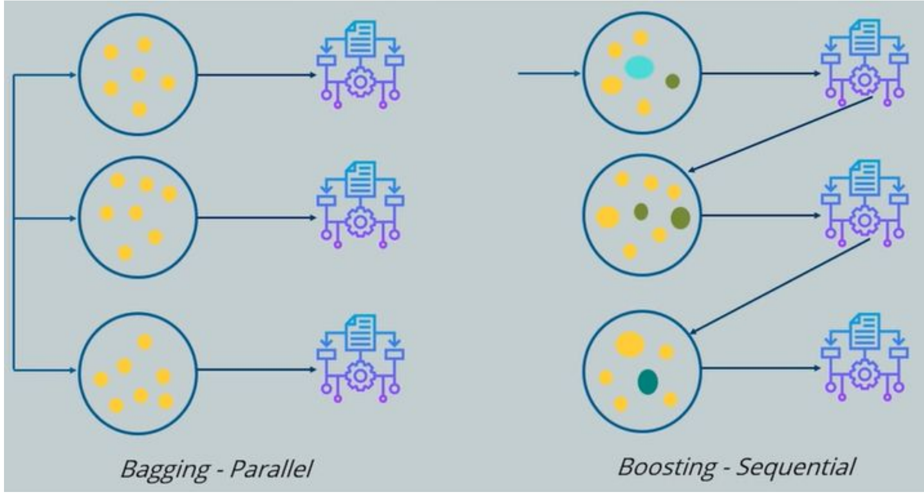
- Interested in SVM “kernel trick” and other methods that that can classify datasets which are not linearly separable.
- AdaBoost is a simple Boosting Algorithm which does the above.



https://www.eric-kim.net/eric-kim-net/posts/1/kernel_trick.html

Problem Formulation

- Can you build a strong classifier out of weak classifiers?
    - **Ensemble Learning**
        - **Bagging** - training *in parallel* a number of weak learners on a bootstrapped data set and taking majority vote
        - **Boosting** - *iteratively* training weak classifiers and summing them
- Will the resulting classifier be **computationally efficient**, **accurate,** and **robust**?



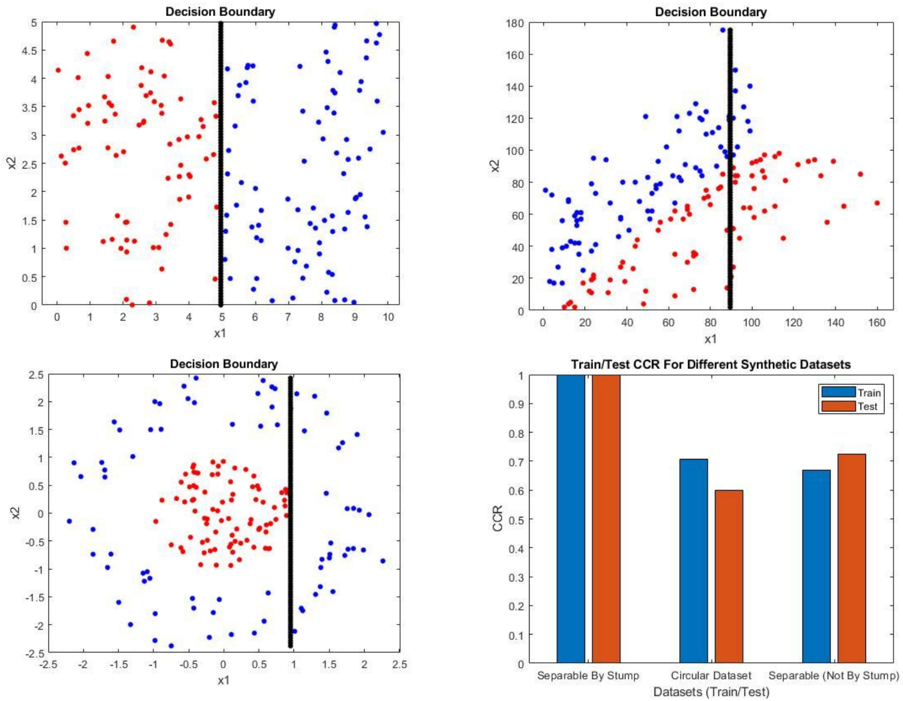
https://www.youtube.com/watch?v=kho6oANGu_A

Solution Approaches

- **Weak Classifier:**
    - Decision Stump
- **Boosting Techniques:**
    - XGBoost
    - CatBoost
    - LogitBoost
    - LPBoost
    - **AdaBoost (Adaptive Boosting)**

http://gallery.wacom.com/gallery/55276859/Yandex-CatBoost

Decision Stump Classifiers



AdaBoost Algorithm

Given: $D = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N)\}$

Initialization: $Weights\ w_0(i) = \frac{1}{n}\ \forall i\ [1, n]\ (equal\ weights\ for\ each\ data\ point)$

$Classifier\ F_0(x) = 0$

Calculate all possible weak classifiers $g \in G$

for t = 1, 2, …, until stopping criterion T

- **Choose classifier that minimizes error:**

Choose classifier $g_t \in G$ such that $g_t = \underset{g \in G}{argmin} \sum_{i=1}^{n} w_t(i)\, 1(g(\boldsymbol{x}_i) \neq y_i) = \underset{g \in G}{argmin}\ \varepsilon_t$

- **Update sum of weak classifiers:**

$F_t = F_{t-1} + \alpha_t * g_t\ where\ \alpha_t = \frac{1}{2} \log\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right)$

- **Update weights on all points:**

$\forall i \in [1{:}n]: w_{t+1}(i) = \frac{w_t(i)}{z_t}\ x \begin{cases} e^{\alpha_t}\ if\ g_t(\boldsymbol{x}_i) \neq y_i \\ e^{-\alpha_t}\ g_t(\boldsymbol{x}_i) = y_i \end{cases}\ where\ z_t = 2\sqrt{\varepsilon_t\,(1-\varepsilon_t)}\ (for\ normalization)$

end for

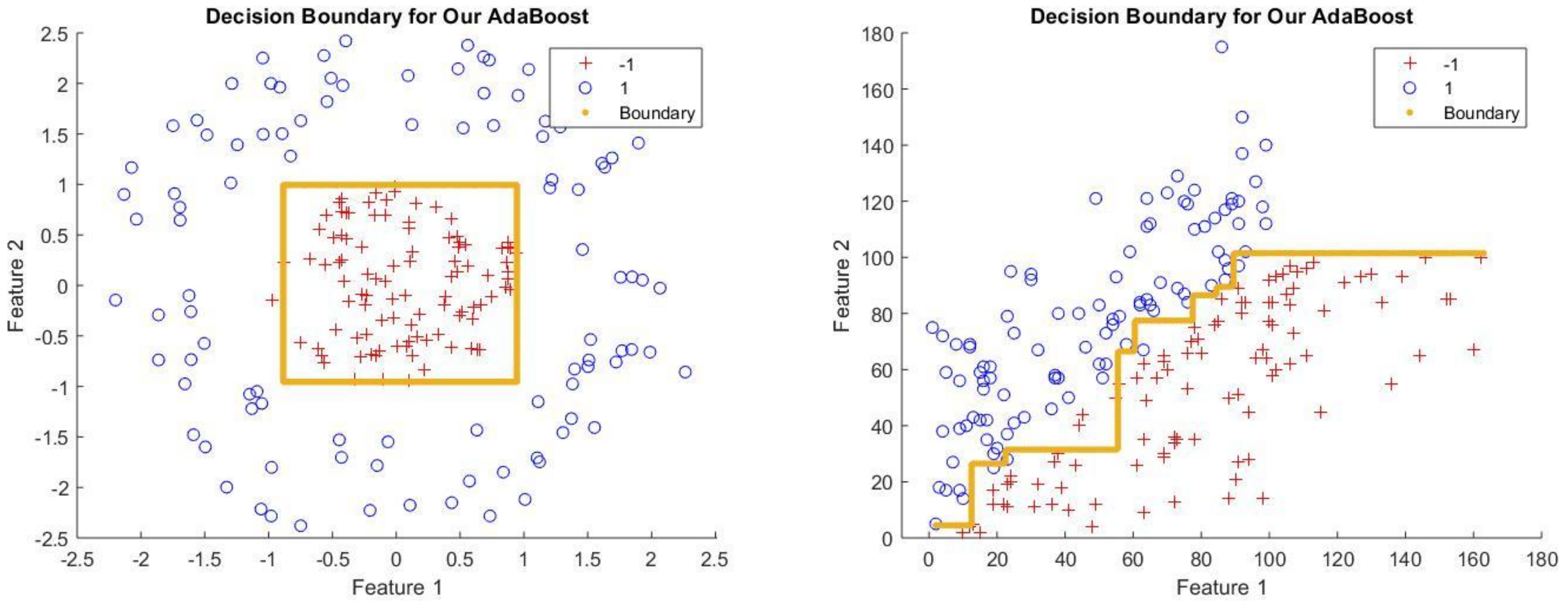
Our AdaBoost Implementation

Initialize: W(nx1), α(Tx1), g(Tx3)
j = 1, ..., n $W_j = 1/n$
all_gs = calculate_gs(data)

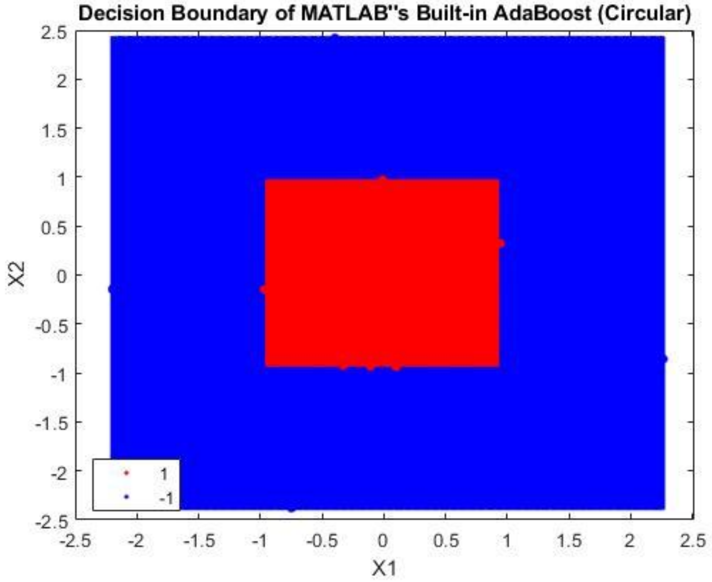
for t = 1, 2, …, T
    best_classifier = calculate_best_g(data_train, W, all_gs)
    store weight of current classifier α_t
    Store current classifier as follows g_t = [feature, threshold, smaller_is]
    W = update_weigths(data_train, W, best_classifier)
end for

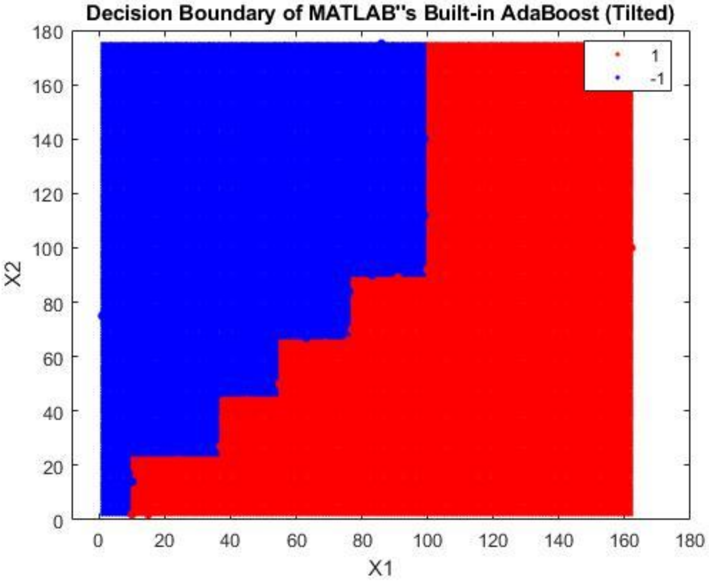
$F(x_{test}) = \text{sign}(\ \alpha_1 * g_1(x_{test}) + \alpha_2 * g_2(x_{test}) + \ldots + \alpha_T * g_T(x_{test})\ )$

AdaBoost Decision Boundaries

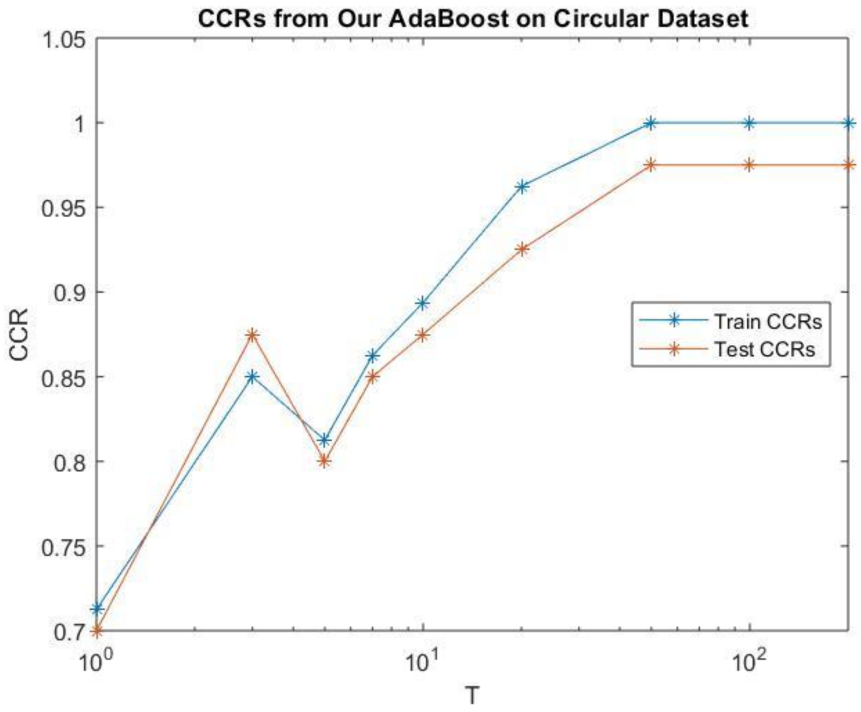


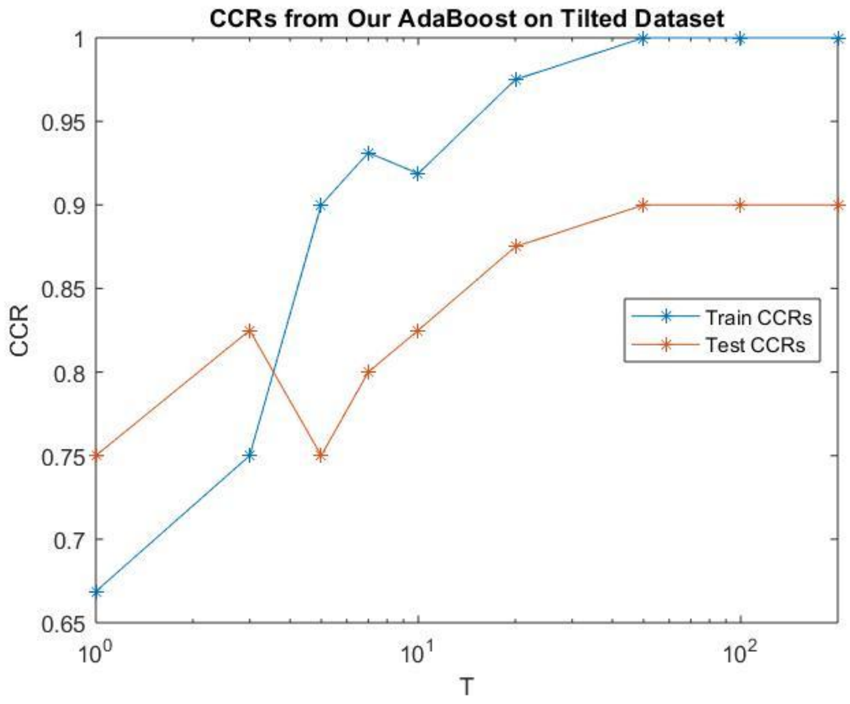
MATLAB’s AdaBoost (using fitcensemble)



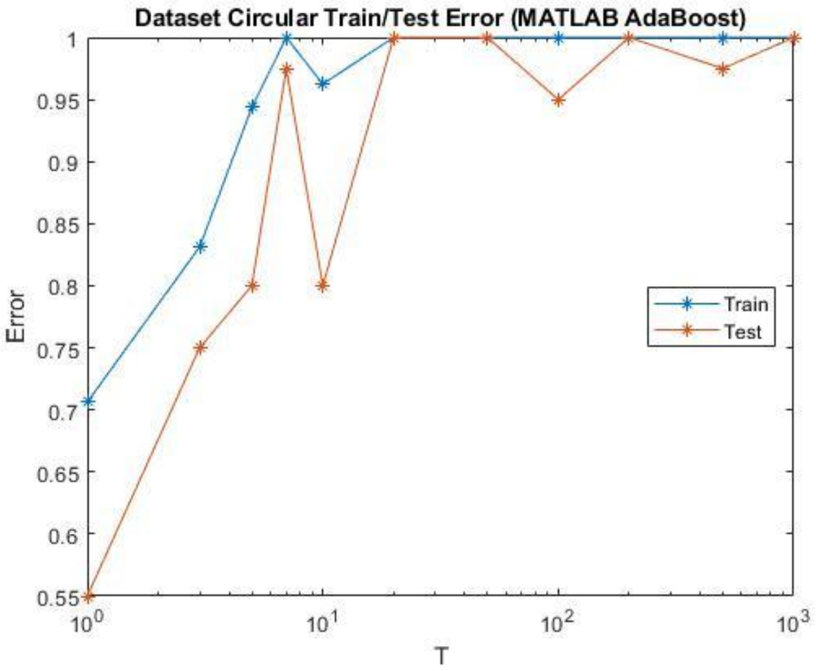


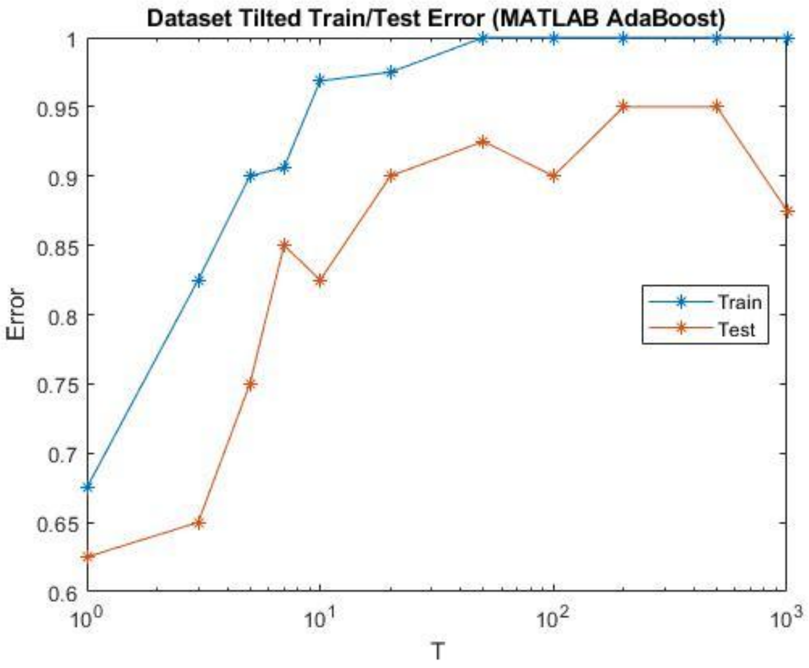
CCR vs Number of Iterations





MATLAB’s AdaBoost (using fitcensemble)

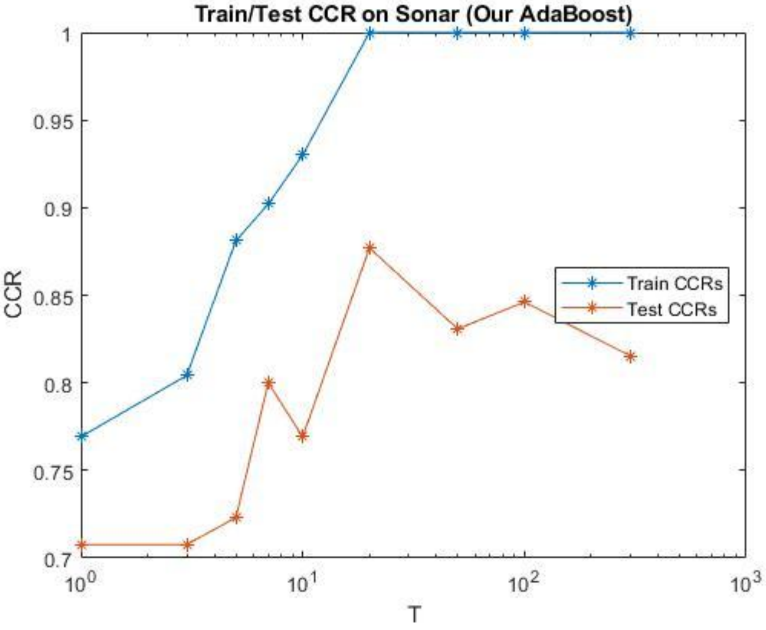


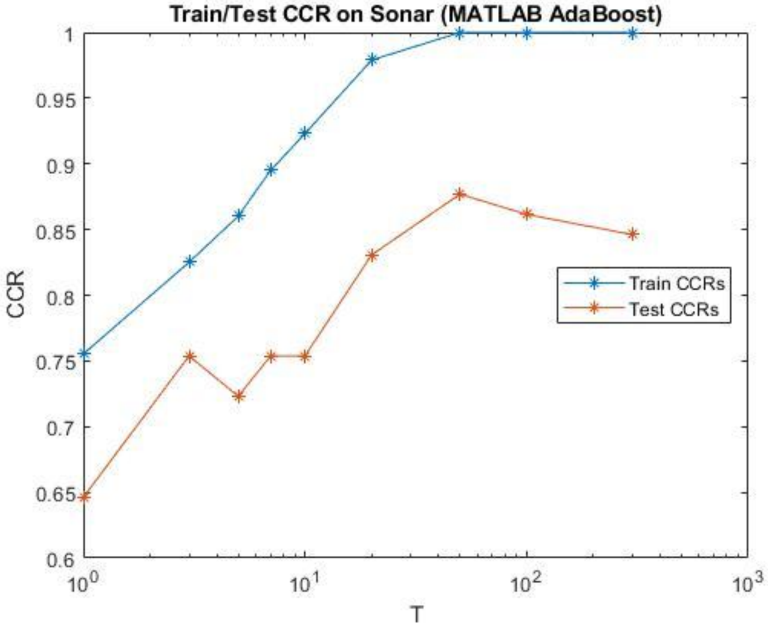


Sonar Dataset

- Small-scale dataset: Sonar Dataset- https://www.openml.org/d/40
- *Binary classification Problem*
- 60 numerical (Float) features, 208 Samples, 86 kB
- Labels: (Rock, Mine)

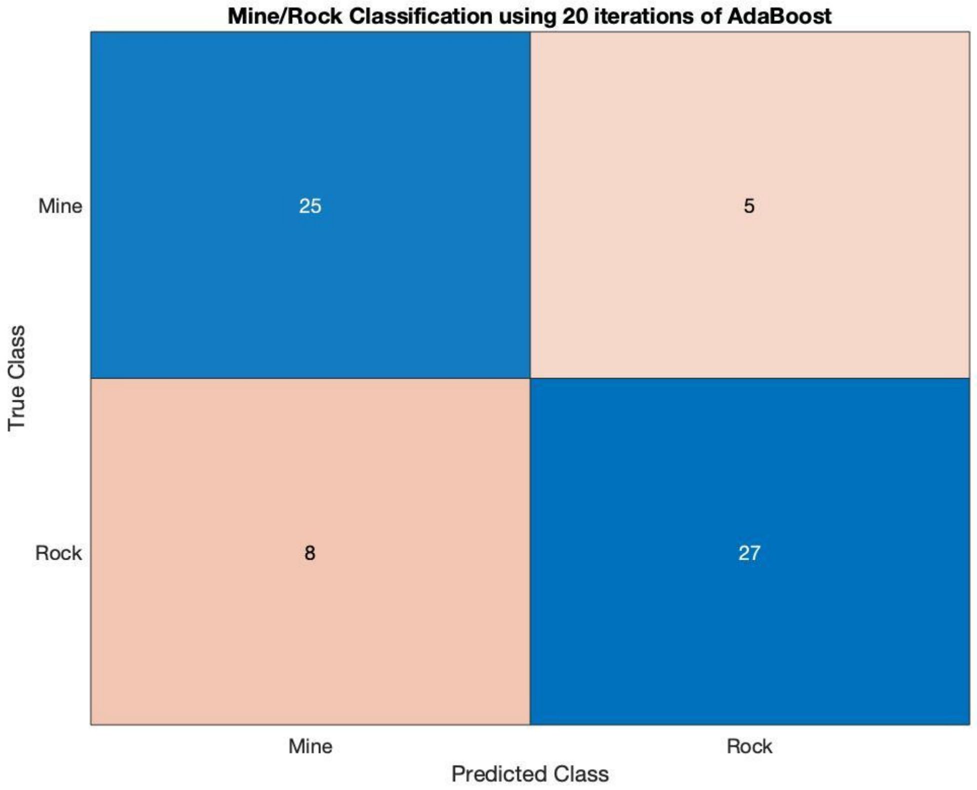
Train/Test CCRs for AdaBoost on Sonar





Sonar Dataset Confusion Chart

- 80 % CCR
- 5 mines (~ 8 %) wrongly classified as rocks
- AdaBoost had weaker results on higher dimensional dataset
- Not safe for military application...



Future Work

- Determine ideal boosting iteration T (tuning parameter) through cross validation
- Test our AdaBoost algorithm on a large scale dataset
- Compare AdaBoost on stumps to SVM kernel trick

Our Conclusions (Based on our implementation and our dataset)

- AdaBoost performs relatively well for 2 dimensional data (test CCR > 90 %)
- AdaBoost didn't perform as well on high dimensional data (test CCR < 90 %)
- AdaBoost performs marginally better than a weak classifier (30 % CCR increase on circular dataset)

Works Cited

Ben-David, Shai, and Shai Shalev-Shwartz. Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press, 2014.

edureka! "Boosting Machine Learning Tutorial | Adaptive Boosting, Gradient Boosting, XGBoost | Edureka." Youtube, 27 Jun. 2019.

Jiao, Jiantao, et al. “Lecture 19: AdaBoost and Course Review.” *EE 378A: Statistical Signal Processing*, 2016, web.stanford.edu/class/ee378a/lecture-notes/lecture19.pdf.

Murphy, Kevin P. Machine Learning: a Probabilistic Perspective. MIT Press, 2013.

SauceCat. “Boosting Algorithm: AdaBoost.” *Medium*, Towards Data Science, 30 Apr. 2017, towardsdatascience.com/boosting-algorithm-adaboost-b6737a9ee60c.

Starmer, Josh. [StatQuest With Josh Starmer]. (2019, January 14). AdaBoost, Clearly Explained [Video file]. https://www.youtube.com/watch?v=LsK-xG1cLYA

https://www.ncbi.nlm.nih.gov/pubmed/26390865