

# ECE 569 HW 3

Brandyn Tucknott

Due 5 December 2025

All code has been submitted on canvas.

1. Consider the classification problem in HW 2. Implement the following using CVX.

(a) Apply the C-Hull formulation to train a classifier:

$$\begin{aligned} & \underset{u,v}{\text{minimize}} \quad \|Au - Bv\|_2^2 \\ & \text{subject to} \quad 1^T u = 1, u \succeq 0 \\ & \quad \quad \quad 1^T v = 1, v \succeq 0. \end{aligned}$$

Visualize the training data together with a classifier. Also visualize the testing data and the classifier in another figure, and report the classification error on the testing data using the true labels provided in `test_separable.mat`.

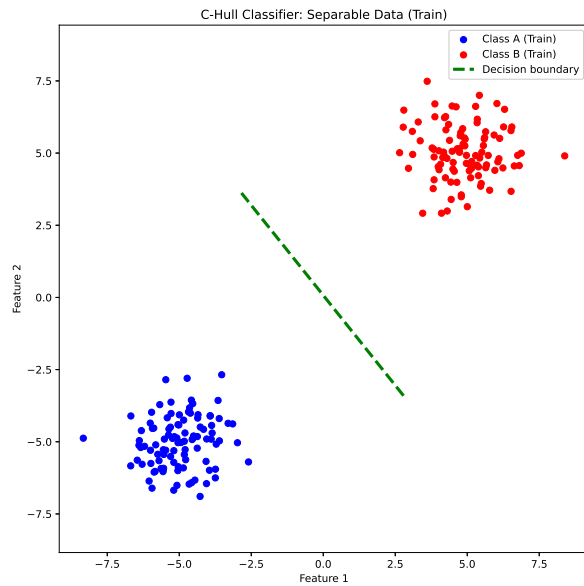


Figure 1: C-Hull Classifier, Separable Training Data

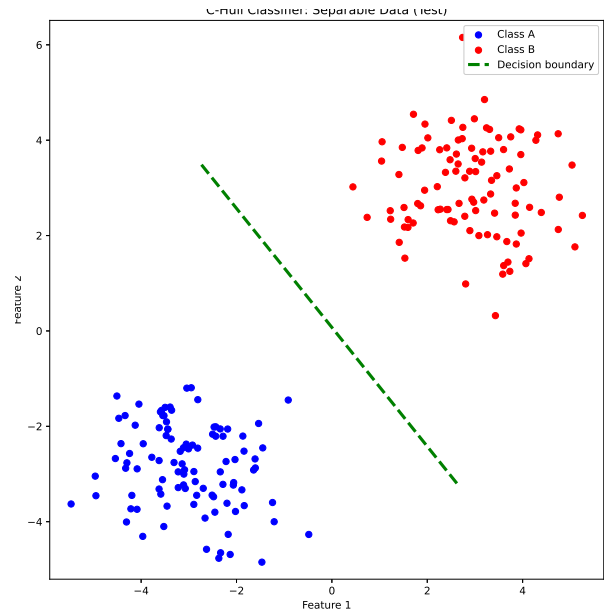


Figure 2: C-Hull Classifier, Separable Testing Data

After solving for a C-Hull classifier, the given test data is classified with 100% accuracy.

(b) repeat the above for `train_overlap.mat` and `test_overlap.mat` using the reduced C-Hull:

$$\begin{aligned} & \underset{u,v}{\text{minimize}} \quad \|Au - Bv\|_2^2 \\ & \text{subject to} \quad 1^T u = 1, d1 \succeq u \succeq 0 \\ & \quad \quad \quad 1^T v = 1, d1 \succeq v \succeq 0. \end{aligned}$$

Report the classification error on the testing data using an appropriate  $d$ .

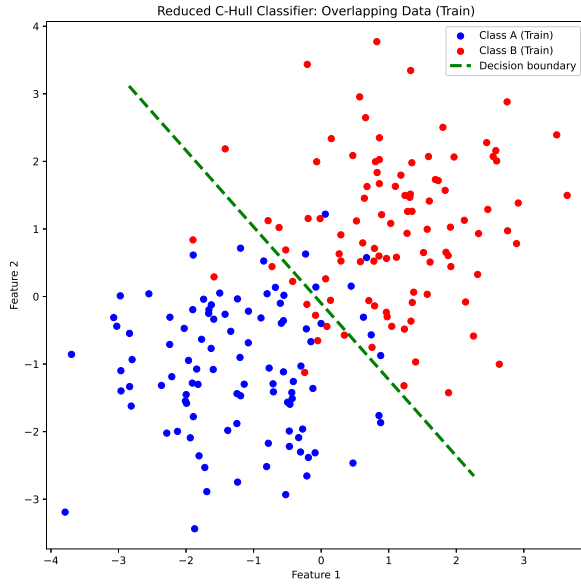


Figure 3: Reduced C-Hull Classifier, Overlapping Training Data

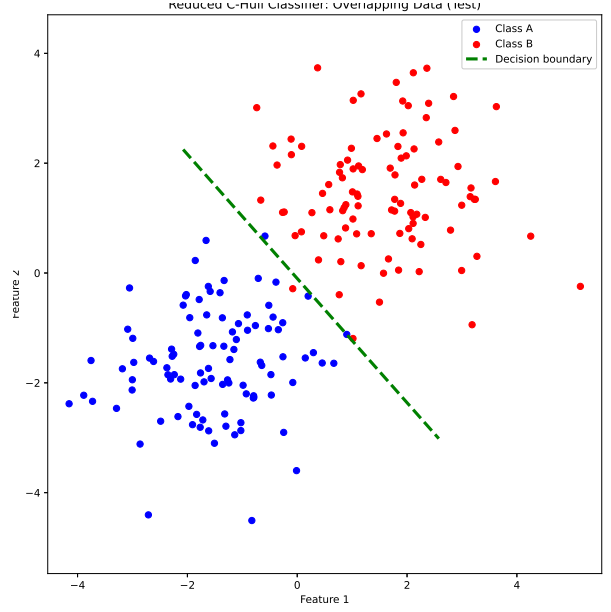


Figure 4: Reduced C-Hull Classifier, Overlapping Testing Data

After solving for a Reduced C-Hull classifier with  $d = 0.75$  (admittedly arbitrary), an accuracy of 98.5% is achieved on the test data.

2. Under the same setting of Question 1, do the following:

- (a) Implement C-Hull and Reduced C-Hull using the projected gradient. Do the same visualization as in Question 1.

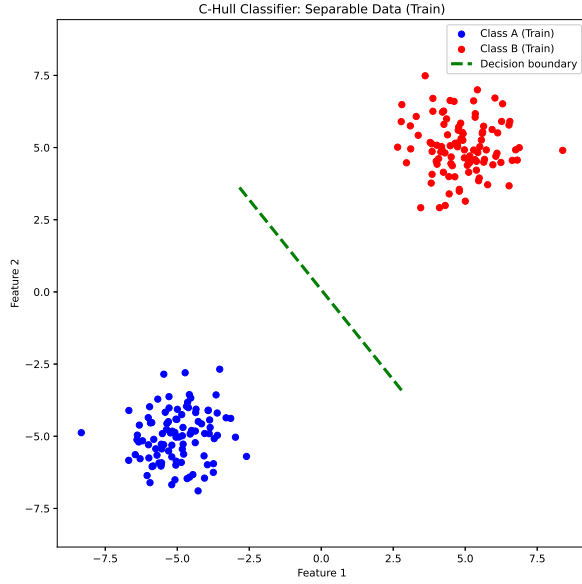


Figure 5: Projected Gradient (C-Hull), Separable Training Data

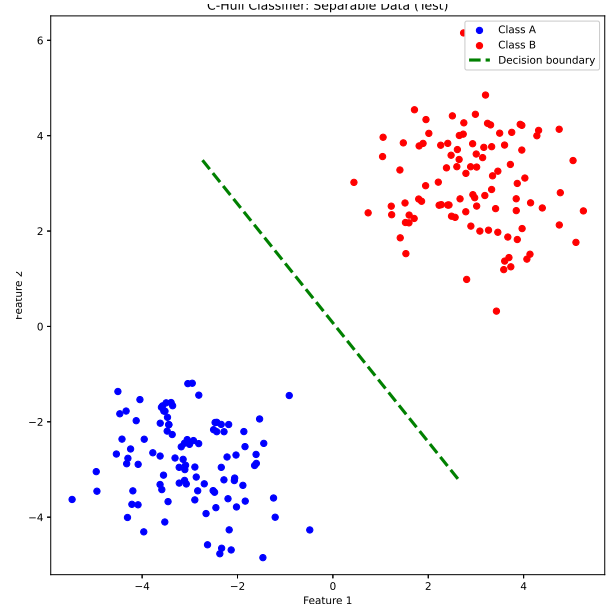


Figure 6: Projected Gradient (C-Hull), Separable Testing Data

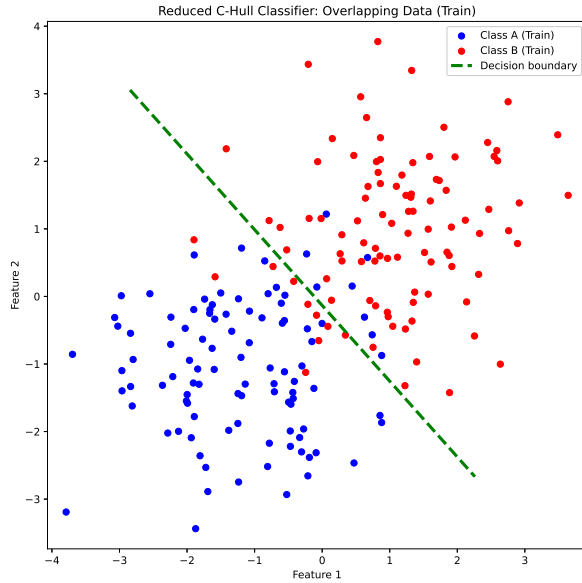


Figure 7: Projected Gradient (Reduced C-Hull), Overlapping Training Data

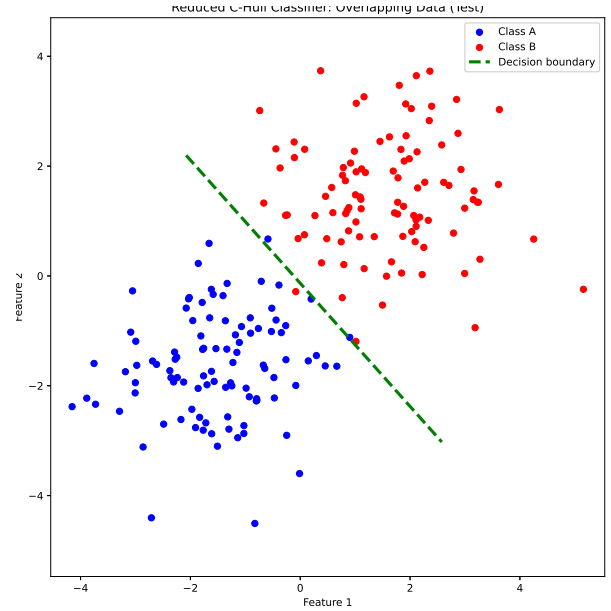


Figure 8: Projected Gradient (Reduced C-Hull), Overlapping Testing Data

Implementing projected gradient descent for the C-Hull yielded an accuracy of 100% on the separable test set. Projected gradient descent on reduced C-Hull yielded the same outcome as in question 1, with an accuracy of 98.5% ( $d = 0.75$ ) on the overlapping test set. The pseudo-code

for the implementation for C-Hull and reduced C-Hull is provided in `q2_pgd_chull.py` and `q2_pgd_rchull.py` respectively.

(b) Repeat the above using Nesterov acceleration.

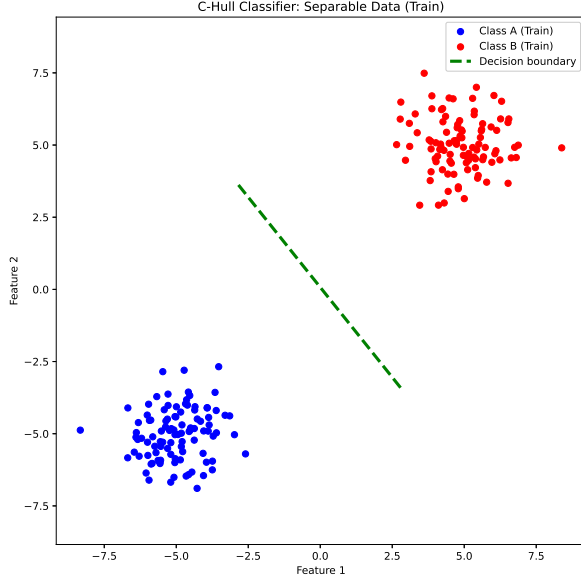


Figure 9: Nesterov Acceleration (C-Hull), Separable Training Data

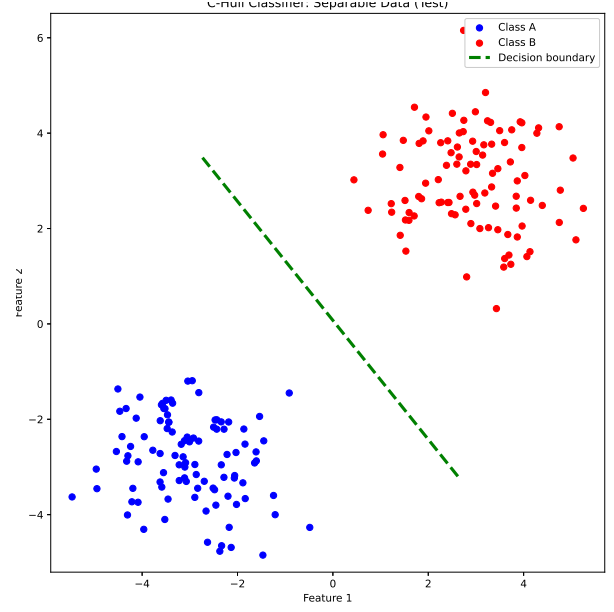


Figure 10: Nesterov Acceleration (C-Hull), Separable Testing Data

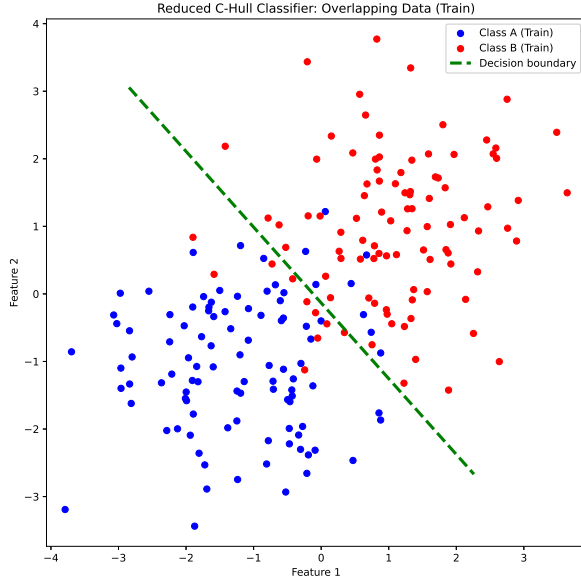


Figure 11: Nesterov Acceleration (Reduced C-Hull), Overlapping Training Data

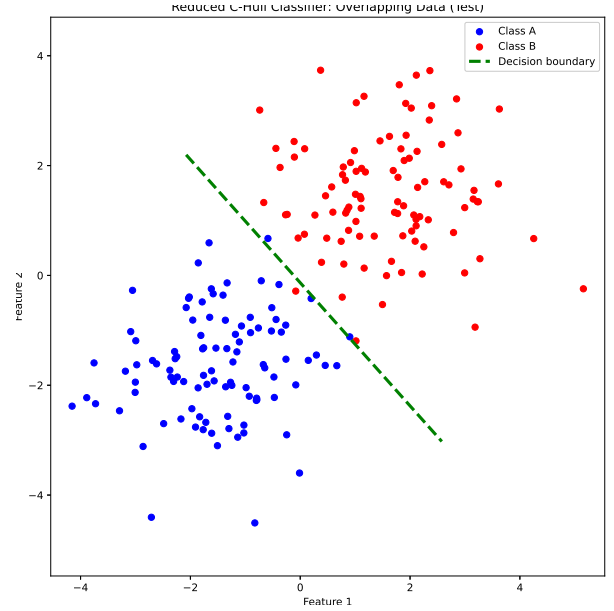


Figure 12: Nesterov Acceleration (Reduced C-Hull), Overlapping Testing Data

Implementing Nesterov acceleration for C-Hull gave a 100% accuracy on the separable test data, and again the reduced C-Hull gave an accuracy of 98.5% with  $d = 0.75$ . The pseudo-code for C-Hull and reduced C-Hull are in `q2_nagd_chull.py` and `q2_nagd_rchull.py` respectively.

3. Under the same setting of Question 1, implement C-Hull and Reduced C-Hull using ADMM. Do the same visualization as in Question 1.

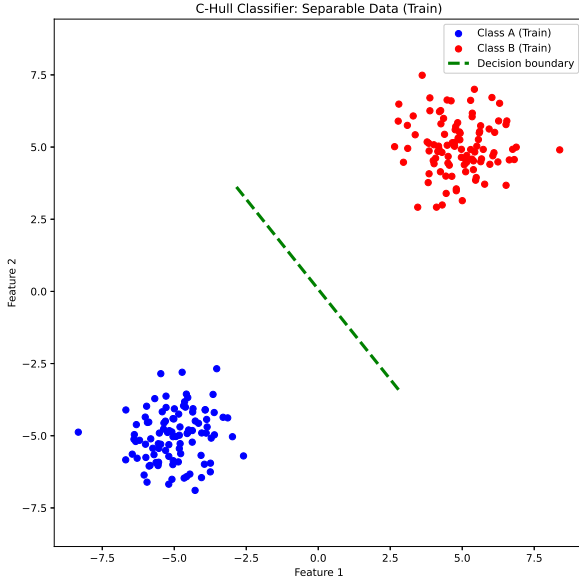


Figure 13: ADMM (C-Hull), Separable Training Data

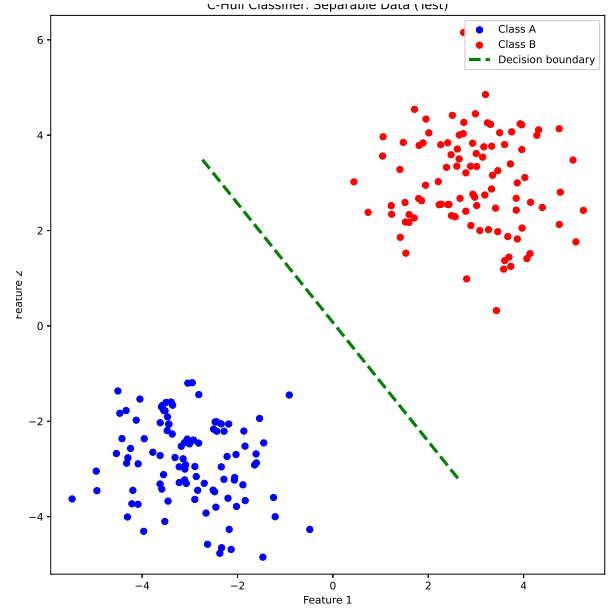


Figure 14: ADMM (C-Hull), Separable Testing Data

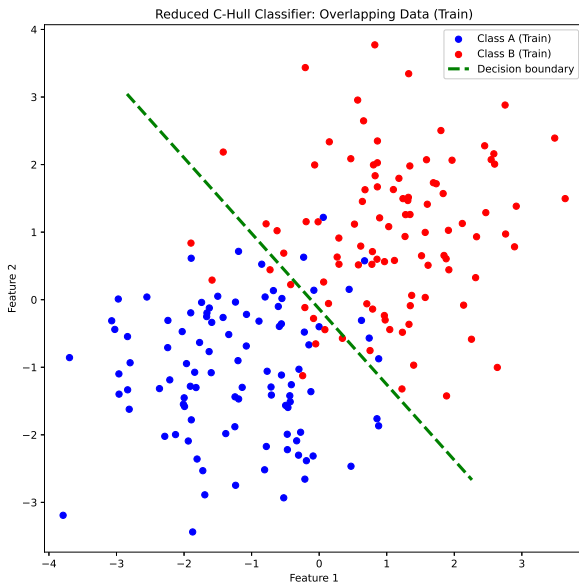


Figure 15: ADMM (Reduced C-Hull), Overlapping Training Data

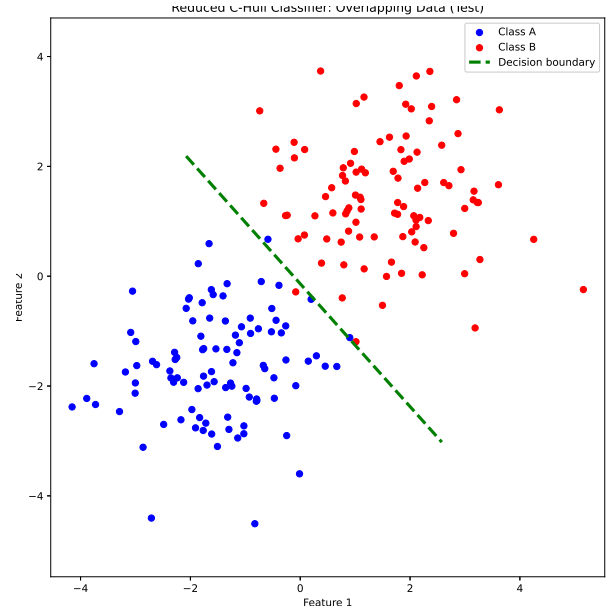


Figure 16: ADMM (Reduced C-Hull), Overlapping Testing Data

Using ADMM, implementing C-Hull gave a 100% accuracy on the separable test data, and the reduced C-Hull gave an accuracy of 98.5% with  $d = 0.75$ . The pseudo-code for C-Hull and reduced C-Hull are in `q3_admm_chull.py` and `q3_admm_rchull.py` respectively.

4. Plot an Iteration vs Objective Value figure for the training process. Compare all the algorithms that you implemented in this figure. In addition, plot a Time vs Objective Value figure using all the algorithms.
- DISCLAIMER: I am fairly confident the graphs are wrong, but I couldn't figure out how to fix them.

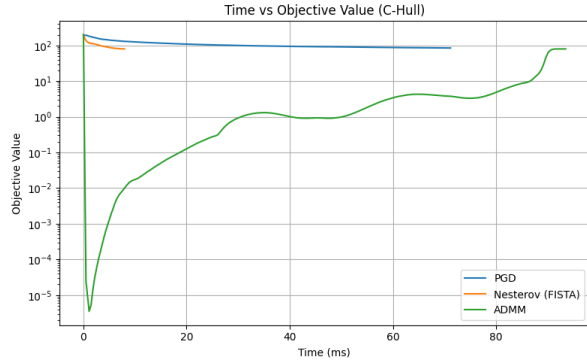


Figure 17: Time vs Objective Value (C-Hull)

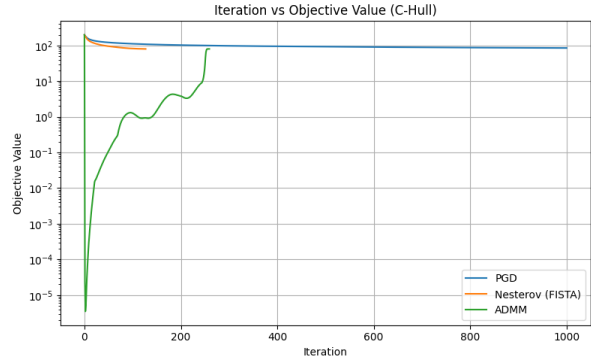


Figure 18: Iteration (ms) vs Objective Value (C-Hull)

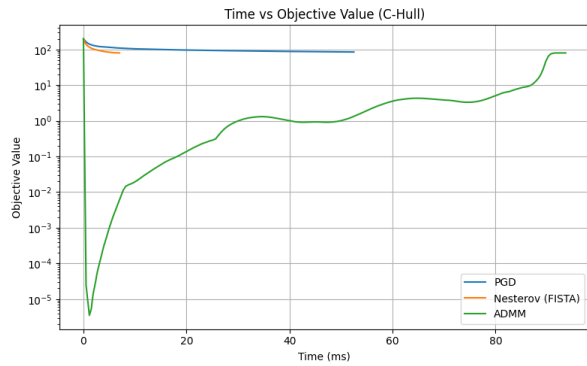


Figure 19: Time vs Objective Value (Reduced C-Hull)

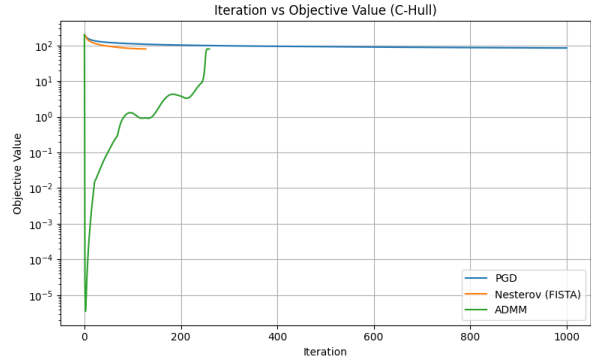


Figure 20: Iteration (ms) vs Objective Value (Reduced C-Hull)