Assignment 4 SML Ritvik Shekhar 2023440

Steps of AdaBoost Algorithm

1. Initialize sample weights:

$$w_i = \frac{1}{n}, \quad \text{for } i = 1, 2, \ldots, n$$

2. For each round $j = 1$ to T:

   (a) For each feature:

   - Generate 100 thresholds between the feature's min and max.
   - For each threshold and each polarity ($+1$ and -1):
     - Predict $h_j(x) \in \{-1, +1\}$
     - Compute the weighted loss:

$$L_j = \sum_{i=1}^{n} w_i \cdot \mathbb{I}(y_i \neq h_j(x_i)) / \sum w_i$$

       where $\mathbb{I}$ is the indicator function.
     - Choose the feature, threshold, and polarity that minimizes L_j

   (b) Compute the weight of the weak classifier h_j:

$$\alpha_j = \frac{1}{2} \log \left(\frac{1 - L_j + \varepsilon}{L_j + \varepsilon} \right), \quad \text{(e.g., } \varepsilon = 10^{-10} \text{ to avoid division by zero)}$$

   (c) Update the sample weights:

$$w_i \leftarrow w_i \cdot \exp(-2\alpha_j y_i h_j(x_i))$$

   (d) If correctly classified wt will decrease and if wrongly classified wt will increase

   (e) Normalize the sample weights so that $\sum_i w_i = 1$

   (f) Add the classifier h_j with weight α_j to the ensemble array.

3. Final prediction after T rounds:

$$H(x) = \text{sign} \left(\sum_{j=1}^{T} \alpha_j h_j(x) \right)$$

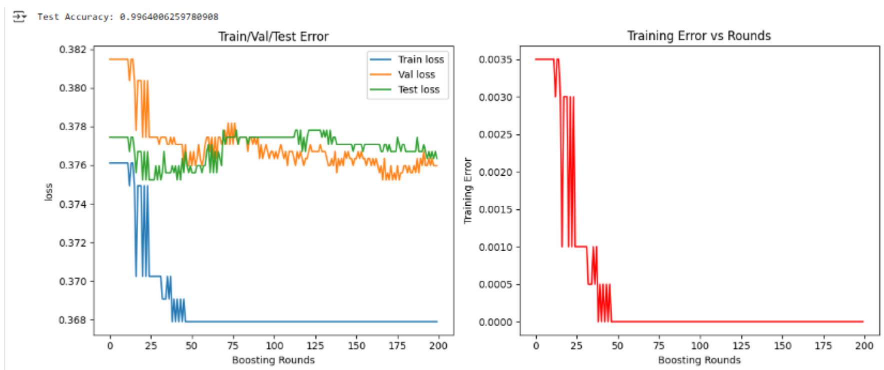

Figure 1: Test/validation/train loss and train error with iterations

Gradient Boosting for Regression (with Decision Stumps)

1. **Initialize the model** with a constant prediction:

$$F_0(x) = \bar{y}$$

2. **For $m = 1$ to M (number of boosting rounds):**

   (a) Compute the residuals (negative gradient):

   (b) In case of squared error:

   $$r_i^{(m)} = y_i - F_{m-1}(x_i)$$

   (c) In case of absolute error:

   $$r_i^{(m)} = sign(y_i - F_{m-1}(x_i))$$

   (d) Fit a **decision stump** $h_m(x)$ to the residuals:

   - Try all possible split points (midpoints between sorted feature values).
   - For each split, compute the left and right means of residuals.
   - Choose the split that minimizes:

   $$\text{RSS} = \sum_{i \in \text{left}} (r_i - \bar{r}_{\text{left}})^2 + \sum_{i \in \text{right}} (r_i - \bar{r}_{\text{right}})^2$$

   (e) Update the model:

   $$F_m(x) = F_{m-1}(x) + \nu h_m(x)$$

   where ν is the learning rate.

3. **Final prediction is :**

$$\hat{y} = F_M(x)$$

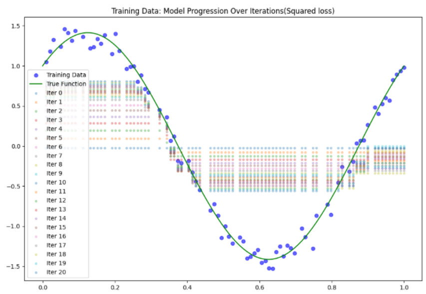

Figure 2: Plot predictions for training data and ground truth vs. iterations for squared loss

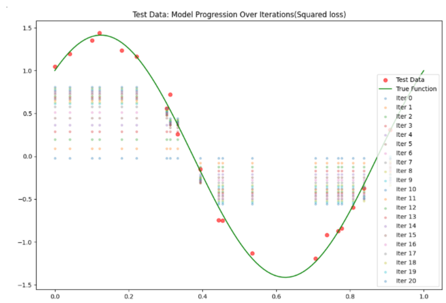

Figure 3: Plot predictions for test data and ground truth vs. iterations for squared loss

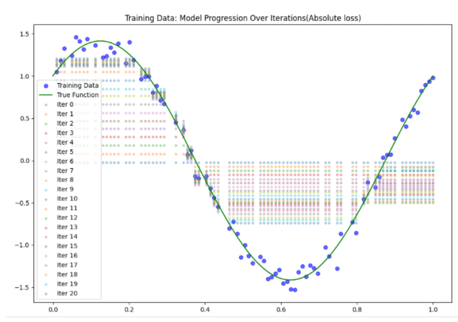

Figure 4: Plot predictions for train data and ground truth vs. iterations for absolute loss

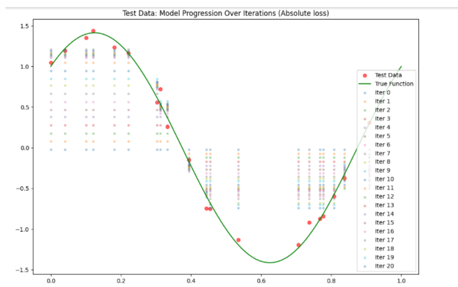

Figure 5: Plot predictions for test data and ground truth vs. iterations for absolute loss

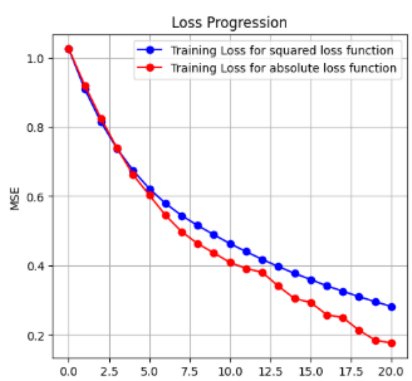

Figure 6: Plot training loss over iterations

```
X_train: (10, 2)
y_train: [0 0 1 0 1 0 1 1 1 0]
X_test: (10, 2)
y_test: [1 0 1 0 0 1 1 1 0 0]
original :1 , y_pred :1.0090968033570897
original :0 , y_pred :1.1095189842020597e-38
original :1 , y_pred :1.0090968028000118
original :0 , y_pred :2.599280760911318e-27
original :0 , y_pred :3.158331264865705e-27
original :1 , y_pred :0.2680681744629958
original :1 , y_pred :0.129844046768557
original :1 , y_pred :1.0081379999838893
original :0 , y_pred :8.839579055137779e-30
original :0 , y_pred :1.0434327514628875e-36
Mean Squared Error on test set: 0.12931273108742902
```

Figure 7: Mean Squared error

1 Neural network