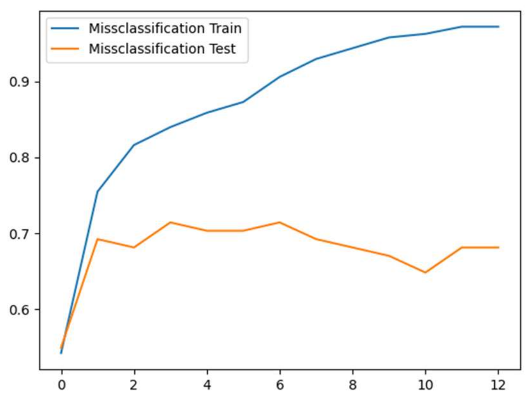
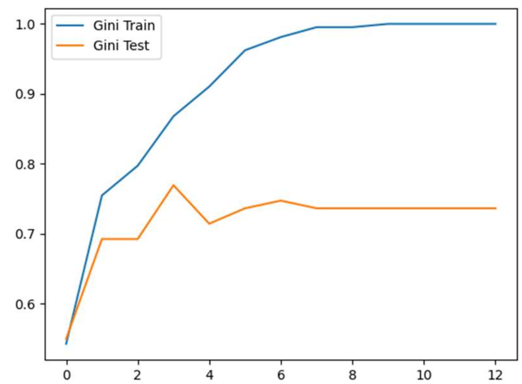
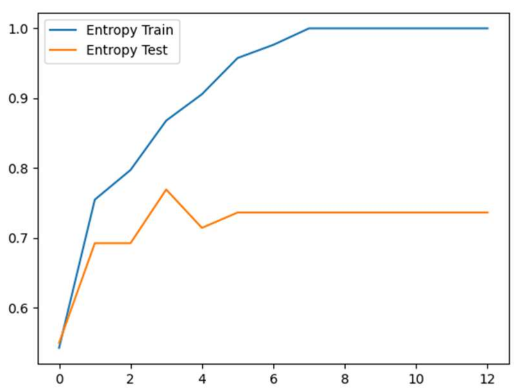
Exercise 1

a)







x-axis: max depth, y-axis: accuracy

In general, all the loss functions increase in test accuracy as they increase in training accuracy up until a point where further increases in training accuracy is correlated with decreased testing accuracy. As max depth increases, training accuracy increases and test accuracy also increases to a certain point, then decreases. The Gini and Entropy loss functions seem to perform similarly in test and training accuracy.

b)

Without Random Forests

Median: 0.8351648351648352, Minimum: 0.8241758241758241, Maximum: 0.8351648351648352

By using Bagging without Random Forests, the accuracy is greater than that of the non-ensemble methods, even at worst.

With Random Forests

Median: 0.7692307692307693, Minimum: 0.7692307692307693, Maximum: 0.8131868131868132

By using Bagging with Random Forests, the accuracy can potentially be better than that of the non-ensemble methods and on average and at worst perform about as well as the non-ensemble methods.

Comparing the use of random forest, it seems that not using random forests provides a better accuracy across the board.

Exercise 2

b/ $E[e^{-yH} \mid X=x]$

$$= \sum_{y \in \{-1,1\}} e^{-H} p(y \mid X=x)$$

$$= e^{-H} p(y=1 \mid X=x) + e^{H} p(y=-1 \mid X=x)$$

$$\Rightarrow \frac{\partial}{\partial H} E[e^{-yH} \mid X=x] = \frac{\partial}{\partial H}\left(e^{-H} p(y=1 \mid X=x) + e^{H} p(y=-1 \mid X=x)\right)$$

set to 0 $\Rightarrow$ $0 = -e^{-H} p(y=1 \mid X=x) + e^{H} p(y=-1 \mid X=x)$

$$e^{-H} p(y=1 \mid X=x) = e^{H} p(y=-1 \mid X=x)$$

$$e^{2H} = \frac{P(y=1 \mid X=x)}{P(y=-1 \mid X=x)}$$

$$2H = \log \frac{P(y=1 \mid X=x)}{P(y=-1 \mid X=x)}$$

$$H = \frac{1}{2} \log \frac{P(y=1 \mid X=x)}{P(y=-1 \mid X=x)}$$

Thus the minizer of the exponential loss is the log odds ratio.

c/ let $\bar{\epsilon}_t = \epsilon_t(\bar{h}_t)$

since $\epsilon_t > 1/2 \Rightarrow \bar{\epsilon}_t < \frac{1}{2} \Leftarrow \bar{\epsilon}_t = 1 - \epsilon_t$

$$\bar{\beta}_t = \frac{1}{2} \log \frac{1-\bar{\epsilon}_t}{\bar{\epsilon}_t}$$

$$= \frac{1}{2} \log \frac{1-(1-\epsilon_t)}{1-\epsilon_t}$$

$$= \frac{1}{2} \log \frac{\epsilon_t}{1-\epsilon_t}$$

$$= -\frac{1}{2} \log \frac{1-\epsilon_t}{\epsilon_t}$$

$$= -\bar{\beta}_t$$

$$\bar{\epsilon}_t = \epsilon_t(\bar{h}_t) = \epsilon(-\bar{h}_t) = \sum_{i=1}^{n} p_i^t [\![-h_t(x_i) \neq y_i]\!]$$

consider the update rule

$$\exp(-y_i \bar{B}_t \bar{h}_t(x_i)) = \exp(-y_i(-B_t)(-h(x_i))) = \exp(-y_i B_t h(x_i))$$

$\therefore$ we have the same update for w in (4) d/

$$E_t \exp[-y\beta h_t(x)] \qquad (x,y) \sim p^t$$

$$= \sum_{i=1}^{n} \exp[-y_i\beta h_t(x)]\, p^t(y_i)$$

$$= \sum_{i:h_t(x_i)=y_i}^{n} \exp[-y_i\beta h_t(x)]\, p^t(y_i) + \sum_{i:h_t(x_i)\neq y_i}^{n} \exp[-y_i \beta h_t(x)]\, p^t(y_i)$$

$$= \sum_{i:h_t(x_i)=y_i}^{n} e^{-\beta} p_i^t + \sum_{i:h_t(x_i)\neq y_i}^{n} e^{\beta} p_i^t$$

Partial derivative wrt β and set to 0

$$0 = \frac{\partial}{\partial \beta}\left(\sum_{i:h_t(x_i)=y_i}^{n} e^{-\beta} p_i^t + \sum_{i:h_t(x_i)\neq y_i}^{n} e^{\beta} p_i^t\right)$$

$$0 = \sum_{i:h_t(x_i)=y_i}^{n} -e^{\beta} p_i^t + \sum_{i:h_t(x_i)\neq y_i}^{n} e^{\beta} p_i^t \Rightarrow \sum_{i:h_t(x_i)=y_i}^{n} e^{-\beta} p_i^t = \sum_{i:h_t(x_i)\neq y_i}^{n} e^{\beta} p_i^t .$$

$$\log\left(\sum_{i:h_t(x_i)=y_i}^{n} e^{-\beta} p_i^t\right) = \log\left(\sum_{i:h_t(x_i)\neq y_i}^{n} e^{\beta} p_i^t\right)$$

$$-\beta + \log\left(\sum_{i:h_t(x_i)=y_i} p_i^t\right) = \beta + \log\left(\sum_{i:h(x_i)\neq y_i} p_i^t\right)$$

ε_t by eqn (2)

$$-\beta + \log(1-\varepsilon_t) = \beta + \log(\varepsilon_t)$$

$$2\beta = \log(1-\varepsilon_t) - \log(\varepsilon_t)$$

$$\beta = \frac{1}{2} \log \frac{1-\varepsilon_t}{\varepsilon_t}$$

$\therefore$ the optimal β in (12) is given in (3) as desired e/ $\frac{\partial}{\partial \beta} \mathbb{E}_t \exp[-y\beta h_t(x_i)] = \exp(-y\beta h_t(x_i))(-y h_t(x_i)) p_i^t$

by eqn (1) $\quad = \exp(-y\beta h_t(x_i))(-y h_t(x_i)) \frac{w_i^t}{\sum_{j=1}^{n} w_j^t}$

set to 0 $\Rightarrow \quad 0 = \frac{1}{\sum_{j=1}^{n} w_j^t} \sum_{i=1}^{n} w_i^{t+1}(-y h_t(x_i))$

$$0 = \sum_{i=1}^{n} w_i^{t+1}(-y h_t(x_i))$$

divide by $-\sum_{j=1}^{n} w_j^{t+1} \Rightarrow \quad 0 = \sum_{i=1}^{n} p_i^{t+1}(y h_t(x_i))$

$$0 = -\sum_{i: y_i \neq h_t(x_i)} p_i^{t+1} + \sum_{i: y_i = h_t(x_i)} p_i^{t+1}$$

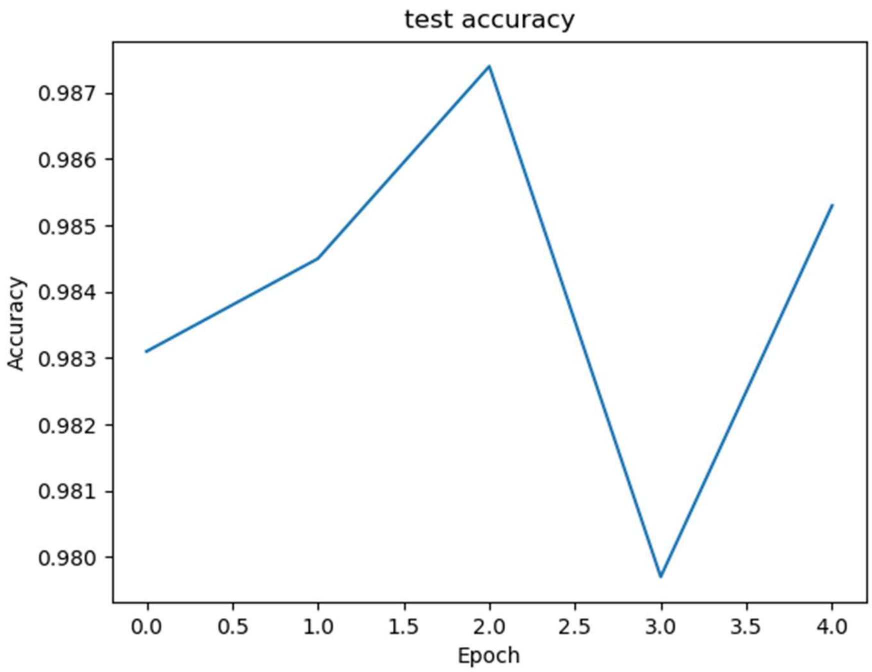
$$0 = -\varepsilon_{t+1} + 1 - \varepsilon_{t+1}$$

$$\varepsilon_{t+1} = \frac{1}{2}$$

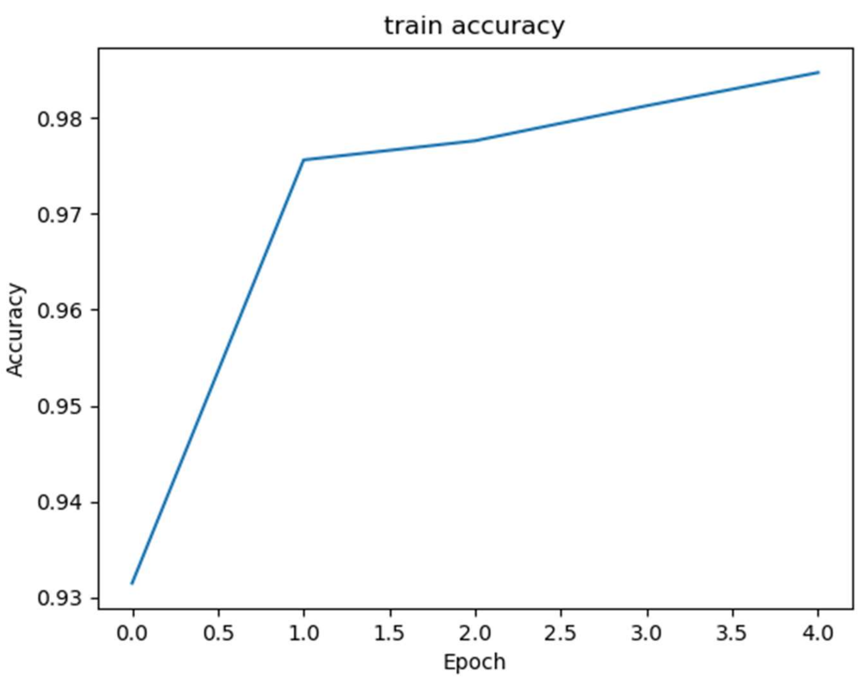
Exercise 3

b) 5 epochs

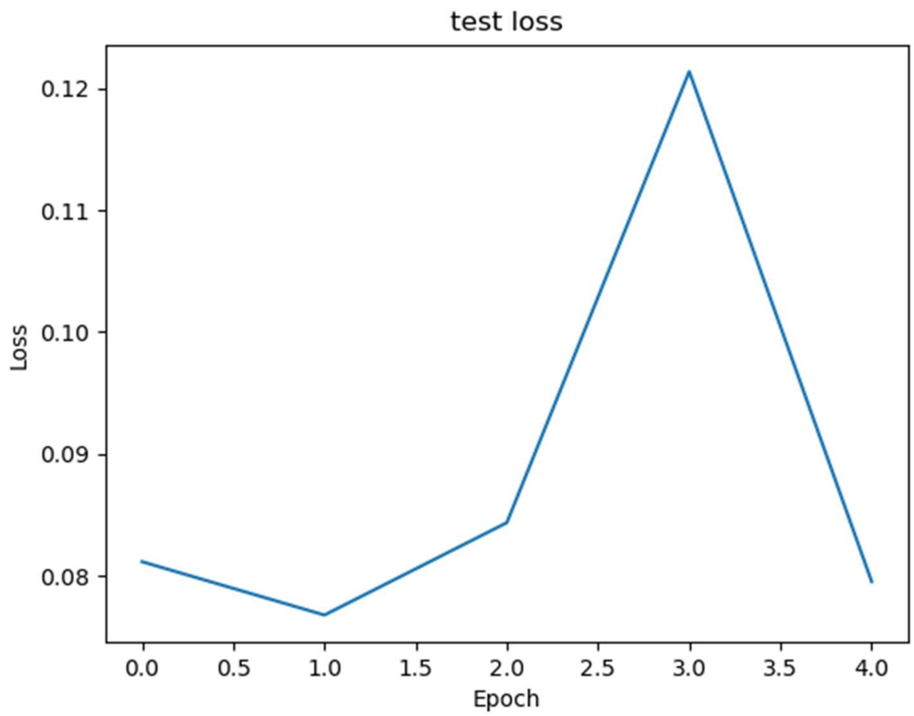
i.



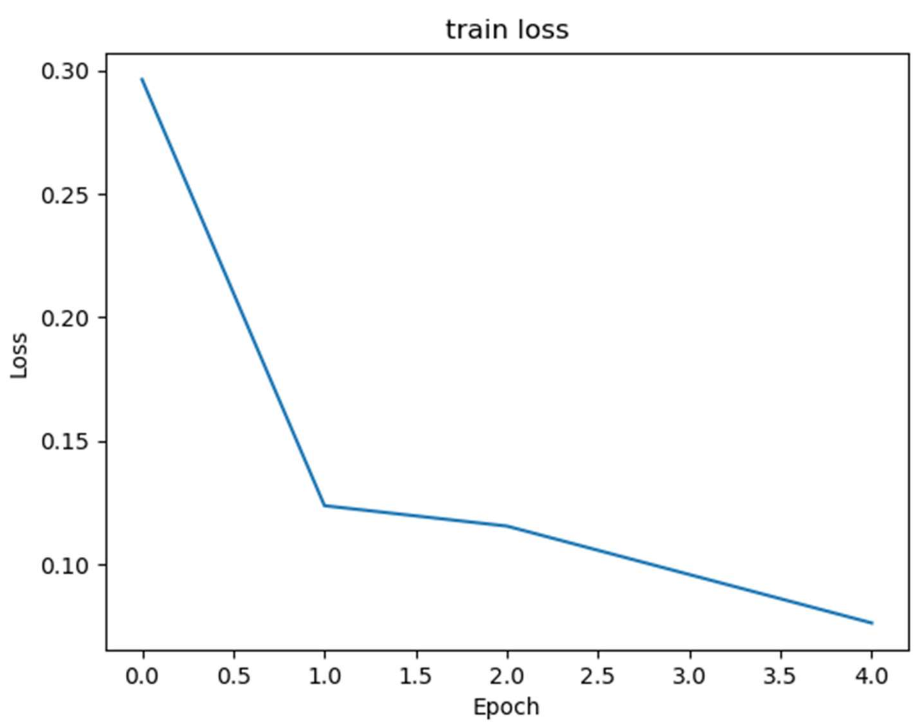
ii.



iii.


test loss
Loss
0.12
0.11
0.10
0.09
0.08
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Epoch

iv.


train loss
Loss
0.30
0.25
0.20
0.15
0.10
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Epoch

c)

Results after flips:

test_horizontal loss, test_ horizontal acc: [5.478106498718262, 0.392300009727478]

test_vertical loss, test_vertical acc: [4.790810585021973, 0.4199999868697815]

There is a large amount of loss and low accuracy across both horizontal and vertical flips.

d)

Results after Gaussian noise:

test_gn0.01 loss, test_gn0.01 acc: [0.08179637789726257, 0.9830999970436096]

test_gn0.1 loss, test_gn0.1 acc: [0.6923056840896606, 0.8141000270843506]

test_gn1 loss, test_gn1 acc: [2.7780394554138184, 0.2563999891281128]

As more noise is added to the test datasets, the loss increases and accuracy decreases.

e)

test loss, test acc: [0.04784698039293289, 0.988099992275238]

test_h loss, test_h acc: [0.04738935828208923, 0.9873999953269958]

test_v loss, test_v acc: [0.06509146094322205, 0.9815999865531921]

test_gn0.01 loss, test_gn0.01 acc: [0.023918500170111656, 0.993399977684021]

test_gn0.1 loss, test_gn0.1 acc: [0.03706265985965729, 0.9900000095367432]

test_gn1 loss, test_gn1 acc: [0.3881167769432068, 0.879800021648407]

I retrained the model by making training sets with each of the modifications made in part 3 and 4, and added them into a final dataset including the original dataset, ie. a dataset of 6 combined data sets. The resulting test accuracies are as you can see above. The datasets saw a decrease in test loss and increase in test accuracy across all modifications.