Exercise 1: prove by induction:

Base case: $t=1$. By the algorithm of Adaboost: $w_1=[1/n, 1/n, \cdots] \in R^n$, $\tilde{w}_1=w_1$ for initialization, $p_1^t=\frac{w_1^t}{\sum_{j=1}^n w_j^t}=\frac{\tilde{w}_1^t}{\sum_{j=1}^n \tilde{w}_1^t}=\tilde{p}_1^t$

Induction step: If $p_i^t=\tilde{p}_i^t$ for all i, the $p_i^{t+1}=\tilde{p}_i^{t+1}$ for all i.

claim: $h_t(x_i)\neq y_i$ iff $\tilde{h}_t(x_i)\neq \tilde{y}_i$: If $h_t(x_i)\neq y_i$, then $\frac{h_t(x_i)+1}{2}\neq\frac{y_i+1}{2}\Rightarrow \tilde{h}_t(x_i)\neq\tilde{y}_i$; If $\tilde{h}_t(x_i)\neq\tilde{y}_i\Rightarrow\frac{h_t(x_i)+1}{2}\neq\frac{y_i+1}{2}$ $h_t(x_i)\neq y_i$, we proved the claim. So:

$\varepsilon_t=\sum_{i=1}^n p_i^t[h_t(x_i)\neq y_i]=\sum_{i=1}^n \tilde{p}_i^t[\tilde{h}_t(x_i)\neq\tilde{y}_i]=\tilde{\varepsilon}_t$;

Two Cases: ① If $h_t(x_i)=y_i$, $\tilde{h}_t(x_i)=\tilde{y}_i$, then $w_i^{t+1}=w_i^t e^{-y_i\beta_t h_t(x_i)}=w_i^t e^{-y_i^2\beta_t}$ $(y_i\in\{\pm1\}, y_i^2=1)=w_i^t e^{\frac{1}{2}\times\log\frac{\varepsilon_t}{1-\varepsilon_t}}=w_i^t\times\sqrt{\frac{\varepsilon_t}{1-\varepsilon_t}}$.

$\tilde{w}_i^{t+1}=\tilde{w}_i^t\left(\frac{\tilde{\varepsilon}_t}{1-\tilde{\varepsilon}_t}\right)^{|-|\tilde{h}_t(x_i)-\tilde{y}_i|}=\tilde{w}_i^t\left(\frac{\varepsilon_t}{1-\varepsilon_t}\right)$,

② If $h_t(x_i)\neq y_i$, $\tilde{h}_t(x_i)\neq\tilde{y}_i$, then $w_i^{t+1}=w_i^t e^{-y_i\beta_t h_t(x_i)}=w_i^t e^{\beta_t}=w_i^t e^{\frac{1}{2}\log\frac{1-\varepsilon_t}{\varepsilon_t}}=w_i^t\cdot\sqrt{\frac{1-\varepsilon_t}{\varepsilon_t}}$

$\tilde{w}_i^{t+1}=\tilde{w}_i^t\left(\frac{\tilde{\varepsilon}_t}{1-\tilde{\varepsilon}_t}\right)^{|-|h_t(x_i)-\tilde{y}_i|}=\tilde{w}_i^t$, so that

So: $\frac{w_i^{t+1}}{\tilde{w}_i^{t+1}}=\frac{w_i^t}{\tilde{w}_i^t}\cdot\frac{\varepsilon_t}{1-\varepsilon_t}$ and. $p_i^t=\tilde{p}_i^t:\frac{p_i^t}{\tilde{p}_i^t}=\frac{\frac{w_i^t}{\sum_{j=1}^n w_j^t}}{\frac{\tilde{w}_i^t}{\sum_{j=1}^n \tilde{w}_j^t}}=1$ $\Longrightarrow$ So. $\frac{w_i^t}{\tilde{w}_i^t}=\frac{\sum_{j=1}^n w_j^t}{\sum_{j=1}^n \tilde{w}_j^t}=a$

$\frac{\sum_{j=1}^n \tilde{w}_j^{t+1}}{\sum_{j=1}^n w_j^{t+1}}=\frac{\sum_{j=1}^n \tilde{w}_j}{\cdot\frac{\varepsilon_t}{1-\varepsilon_t}\cdot\sum_{j=1}^n\tilde{w}_j^t\frac{w_j^t}{\tilde{w}_j^t}}=\frac{\sum_{j=1}^n\tilde{w}_j}{\frac{\varepsilon_t}{1-\varepsilon_t}\cdot a\sum_{j=1}^n w_j^t}=a^{-1}\frac{1-\varepsilon_t}{\varepsilon_t}$

$\frac{\sum_{j=1}^n \tilde{w}_j^t}{\sum_{j=1}^n w_j^t}$ ys a constant

$\frac{p_i^{t+1}}{\tilde{p}_i^{t+1}}=\frac{\frac{w_i^{t+1}}{\sum_{j=1}^n w_j^{t+1}}}{\frac{\tilde{w}_i^{t+1}}{\sum_{j=1}^n \tilde{w}_j^{t+1}}}=\frac{w_i^t}{\tilde{w}_i^t}\cdot\frac{\varepsilon_t}{1-\varepsilon_t}\cdot a^{-1}\cdot\frac{1-\varepsilon_t}{\varepsilon_t}=a\cdot a^{-1}=1$, so

$\frac{p_i^{t+1}}{\tilde{p}_i^{t+1}}=1$, then $p_i^{t+1}=\tilde{p}_i^{t+1}$

By induction, we proved $p_i^t=\tilde{p}_i^t$, then the updates Gustel are equivalent.

(b) for $y\in\{\pm1\}$, then: $\min_{H\in\mathbb{R}} E[\exp(-yH)|X=x] = \min_{H\in\mathbb{R}} P(y=1|X=x)\cdot e^{-H} + P(y=-1|X=x)\cdot e^{H}$

The minimizer $H' = \operatorname{argmin}_{H\in\mathbb{R}} P(y=1|X=x)\cdot e^{-H} + P(y=-1|X=x)\cdot e^{H}$, we find the derivative of such a function.

Let $a = P(y=1|X=x)$, so: $f(H) = a\cdot e^{-H} + (1-a)\cdot e^{H}$: $f'(H) = -ae^{-H} + (1-a)e^{H}$, $f''(H) = ae^{-H} + (1-a)e^{H} > 0$. so: $f'(H)$ keeps increasing, when $f'(H)$ reaches 0, $f(H)$ reaches minimum:

$$-a\cdot e^{-H'} + (1-a)e^{H'} = 0 \Rightarrow (1-a)e^{H'} = a\cdot e^{-H'} \quad e^{2H'} = \frac{a}{1-a}:$$

$$H' = \frac{1}{2}\cdot C\cdot \log\frac{P(y=1|X=x)}{P(y=-1|X=x)} \quad (C \text{ is a constant for ln-log interchange})$$

So $H' \propto \log\frac{P(y=1|X=x)}{P(y=-1|X=x)}$ as required.

(c) Yes, we get the same result in 4):

If $\widehat{h_t}(x) = -h_t(x)$, then $\varepsilon_t(\widehat{h_t}) = \sum_{i=1}^{n} p_i^t\cdot[\widehat{h_t}(x_i)\neq y_i] = \sum_{i=1}^{n} p_i^t\cdot[h_t(x_i)=y_i] = \sum_{i=1}^{n} p_i^t\cdot(1-[h_t(x_i)\neq y_i]) = \sum_{i=1}^{n} p_i^t - \sum_{i=1}^{n} p_i^t[h(x_i)\neq y_i]$

$p_i^t = \frac{w_i^t}{\sum_{j=1}^{n} w_j^t}$ $i=1\cdots n$, $\sum_{i=1}^{n} p_i^t = \frac{\sum_{i=1}^{n} w_i^t}{\sum_{j=1}^{n} w_j^t} = 1$, $\varepsilon_t(\widehat{h_t}) = 1-\varepsilon_t(h_t)$

$\beta_t(\widehat{h_t}) = \frac{1}{2}\log\frac{1-\varepsilon_t(\widehat{h_t})}{\varepsilon_t(\widehat{h_t})} = \frac{1}{2}\log\frac{\varepsilon_t(h_t)}{1-\varepsilon_t(h_t)}$ $w_i^{t+1}(\widehat{h_t}) = w_i^t e^{-y_i\beta_t(\widehat{h_t})\cdot\widehat{h_t}(x_i)}$ $w_i^{t+1}(h_t) = w_i^t e^{-y_i\beta_t(h_t)\cdot h_t(x_i)}$

$$\frac{w_i^{t+1}(\widehat{h_t})}{w_i^{t+1}(h_t)} = \frac{w_i^t e^{-y_i\beta_t(\widehat{h_t})\cdot\widehat{h_t}(x_i)}}{w_i^t e^{-y_i\beta_t(h_t)\cdot h_t(x_i)}} = \frac{e^{\frac{1}{2}\log\frac{\varepsilon_t(\widehat{h_t})}{1-\varepsilon_t(\widehat{h_t})}\cdot\widehat{h_t}(x_i)}}{e^{\frac{1}{2}\log(\frac{1-\varepsilon_t}{\varepsilon_t})\cdot h_t(x_i)}} = \frac{\left(\frac{\varepsilon_t(h_t)}{1-\varepsilon_t(h_t)}\right)^{\frac{1}{2}\widehat{h_t}(x_i)}}{\left(\frac{1-\varepsilon_t}{\varepsilon_t(h_t)}\right)^{\frac{1}{2}h_t(x_i)}} = \left(\frac{\varepsilon_t(h_t)}{1-\varepsilon_t(h_t)}\right)^{\frac{1}{2}\widehat{h_t}(x_i)+\frac{1}{2}h_t(x_i)}$$

Since $\widehat{h_t}(x_i) = -h_t(x_i)$, $\frac{1}{2}\widehat{h_t}(x_i) + \frac{1}{2}h_t(x_i) = 0$. $\frac{w_i^{t+1}(\widehat{h_t})}{w_i^{t+1}(h_t)} = 1$. $w_i^{t+1}(\widehat{h_t}) = w_i^{t+1}(h_t)$

We get the same update for w in 4)

(d) The minimizer $\beta' = \operatorname{argmin}_{\beta\in\mathbb{R}} \hat{E}_t\cdot e^{-y\beta h_t(x)}$, $(x,y)\sim p^t$

$$= \operatorname{argmin}_{\beta\in\mathbb{R}} P_t[y\neq h_t(x)|(x,y)]\cdot e^{\beta} + P_t[y=h_t(x)|(x,y)]\cdot e^{-\beta}, \text{ let } a = P_t[y=h_t(x)|(x,y)]$$

$$= \operatorname{argmin}_{\beta\in\mathbb{R}} (1-a)\cdot e^{\beta} + a\cdot e^{-\beta}.$$

$f(\beta) = (1-a)e^{\beta} + a\cdot e^{-\beta} \Rightarrow f'(\beta) = (1-a)e^{\beta} - a\cdot e^{-\beta} = 0$ $\beta' = \frac{1}{2}\log\frac{a}{1-a} = \frac{1}{2}\log\frac{P_t[y=h_t(x)|(x,y)]}{P_t[y\neq h_t(x)|(x,y)]}$

Considering the distribution: $P_t[y\neq h_t(x)|(x,y)] = \sum_{h_t(x_i)\neq y_i} p_i^t = \sum_{i=1}^{n} p_i^t[h_t(x_i)\neq y_i] = \varepsilon_t$, so the minimizer is:

$\beta' = \frac{1}{2}\frac{1-\varepsilon_t}{\varepsilon_t}$ as stated in (3)

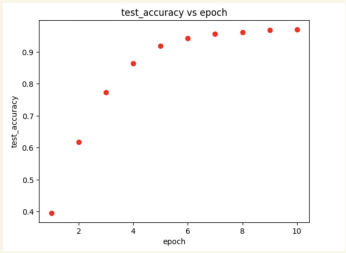
(e) First: $p_i^{t+1} = \frac{\exp(-y_i H_{t+1}(x_i))}{\sum_{i=1}^{n}\exp(-y_i H_{t+1}(x_i))}$ $\varepsilon_{t+1}(h_t) = \frac{\sum_{h_t(x_i)\neq y_i}\exp(-y_i H_{t+1}(x_i))\cdot\exp(\beta_t)}{\sum_{h_t(x_i)\neq y_i}\exp(-y_i H_{t+1}(x_i))\cdot\exp(\beta_t) + \sum_{h_t(x_i)=y_i}\exp(-y_i H_{t+1}(x_i))\cdot\exp(-\beta_t)}$, let $B = \sum_{j=1}^{n}\exp(-y_i H_{t+1}(x_i))$

$\sum_{h_t(x_i)\neq y_i}\exp(-y_i H_{t+1}(x_i)) = B\cdot\varepsilon_t$, so $\varepsilon_{t+1}(h_t) = \frac{B\cdot\varepsilon_t\cdot\exp(\beta_t)}{B\cdot\varepsilon_t\exp(\beta_t) + (B - B\varepsilon_t)\cdot\exp(-\beta_t)} = \frac{\varepsilon_t\cdot\frac{1-\varepsilon_t}{\varepsilon_t}}{\varepsilon_t\cdot\frac{1-\varepsilon_t}{\varepsilon_t} + 1-\varepsilon_t} = \frac{1-\varepsilon_t}{2-2\varepsilon_t} = \frac{1}{2}$

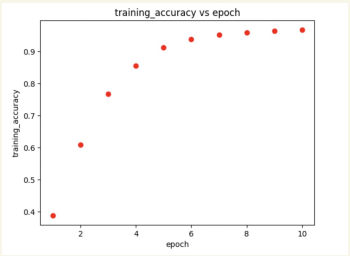
$1-B\cdot\varepsilon_t$

question 2:

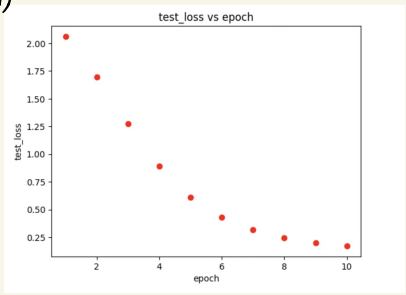
(b) (i)



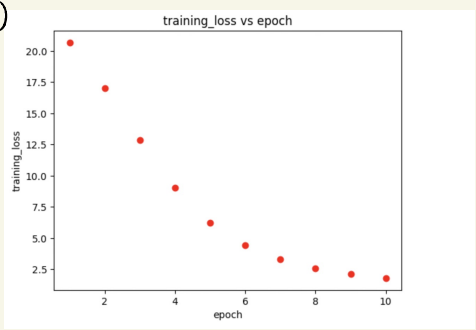
(ii)



(iii)



(iv)



(c) i) I tried the two flip suggested in the function. Here are the results:

Horizontally flip test data.

```
            labels = labels.to
            output = model(ima
            loss = criterion(o
            total_test_loss +=
            _, predicted_label
            total_test_correct
        print(total_test_cor
        print(total_test_los
```

```
5375
1.5957660675048828
```

5375 → Correct prediction out of 10000

1.5957660675048828 → Loss

Vertically flip the test data.

```
            total_test_loss += loss.it
            _, predicted_labels = torc
            total_test_correct += (pre
        print(total_test_correct)
        print(total_test_loss)
```

```
3775
2.5160036087036133
```

3775 → Correct prediction (out of 10000)

2.5160036087036133 → Loss

As we can see, Augmented data has a low Accuracy Rate in our training model

ii) Gaussian Noise:

```
            total_test_correct += (predic
        print(total_test_correct)
        print(total_test_loss)
```

```
9556
0.22907468676567078
```

As we could see, the noise slightly impact our accuracy, but over 95% of calculations are still correct.

d). This is the number of correct prediction with augmented data in Training and without

With Augmented Data.

```
28734
14.555351978167892
```

Accuracy Rate: 95.78%

Without Augmented data.

```
with torch.no_grad():
    total_test_loss = 0
    total_test_correct =
    for i, (images, label
      images = images.to
      labels = labels.to
      output = model(ima
      loss = criterion(ou
      total_test_loss +=
      _, predicted_labels
      total_test_correct
    print(total_test_cor
    print(total_test_loss
```

```
18394
72.10601663589478
```

Accuracy rate: 61.31%