$\boxed{1}$

① $y_n = 1, \hat{y}_n = 0.9$, two decisions

$$L = -[y_n \ln(\hat{y}_n) + (1-y_n)\ln(1-\hat{y}_n)] = -[\ln(0.9)]$$

② $y_n = 2$, one hot $= [0,0,1]$, $\hat{y} = [0.01, 0.09, 0.9]^T$

$$L = -\sum_K y_K \ln(\hat{y}_K) = -\ln(0.9)$$

③ $f(x) = -\ln\left(\frac{1}{1+e^{-x}}\right) = \ln(1+e^{-x})$

$f'(x) = \frac{1}{1+e^{-x}}$

$f''(x) = \frac{e^x}{(1+e^x)^2} \geq 0$ so its convex

④ loss is a convex compisition of logs/sums of linear fencs, these operctions preserve convexity.

⑤ $L = -[y\ln(\hat{y}) + (1-y)\ln(1-\hat{y})]$

$\hat{y} = \frac{1}{1+e^{-z}}$

$$\frac{\delta L}{\delta \hat{y}} = \frac{-y}{\hat{y}} + \frac{1-y}{1-\hat{y}}$$

$$\frac{d\hat{y}}{dz} = \hat{y}(1-\hat{y})$$

$$\frac{\delta L}{\delta z} = \hat{y} - y$$

⑥

$\hat{y} = \text{Softmax } z$

$L = -\sum_i y_i \ln(\hat{y}_i)$

$$\frac{\delta \hat{y}_i}{\delta z_j} = \hat{y}_i(\delta_{ij} - \hat{y}_j)$$

$$\frac{\delta L}{\delta z_j} = \hat{y}_j - y_j$$

2

$$MSE = \frac{1}{2N}\sum_{n=1}^{N}[(y_1 - \hat{y}_1)^2 + (y_2 - \hat{y}_2)^2]$$

$$MAE = \frac{1}{2N}\sum_{n=1}^{N}(|y_1 - \hat{y}_1| + |y_2 - \hat{y}_2|)$$

$$MAPE = \frac{1}{2N}\sum_{n=1}^{N}\frac{|y_1 - \hat{y}_1|}{|y_1|} + \frac{|y_2 - \hat{y}_2|}{|y_2|}$$

3

1. 100, 2. 3, 3. = 1.761 4. 2, 5. 2

4 Bagging will not work if the base learners highly correlated data
Random Forest, because they are trained on bootstrap samples,
reducing correlation

5 Bagging, reduces variance as we are taking averages, boosting
reduces bias by minimizing error from previous rounds, may overfit noise.

6 Stacking of same degree doesnt have benefit, as its space is very
correlated
Stacking diffrent types/structures may help capture other patterns.

7 1-I
2-2

8 1. Parameters done at beginning, such as depth of tree, configurations.
2. Validation set helps choose hyperparameters, preventing overfitting
3. Test set is not seen until we evaluate our model.

9 1. Maximizing the margin makes the decision boundary less sensitive
to noise
2. Nonlinear SVM maximises margin in the dimensions it creates in the
kernel
3. The kernel function computes the inner products in an above
dimension.

$\boxed{10}$ 1. Standard can make most frequent seem very accurate
Weighted accuracy is more fair gives importance to all.

2. Synthetic data generation

$\boxed{11}$ Yes, when our targets are unevenly distributed, we can try and transform target

$\boxed{12}$ $H(p) = -\sum_{K=1}^{K} pk \log pk$

1. $0 \leq pk \leq 1$ & $\ln(pk) \geq 0$
$Hp \geq 0$
2 $\sum pk = 1$

$L = -\sum pk \ln(pk) + \lambda(\sum pk - 1)$

$\frac{\delta L}{\delta pk} = -\ln(pk) - 1 + \lambda = 0$

$pk = e^{\lambda - 1}$

$\sum pk = 1 \rightarrow \frac{1}{K}$, entropy occurs at uniform distribution