1) [6 points] Explain what is the bias-variance trade-off? Describe few techniques to reduce bias and variance respectively.

Solution:

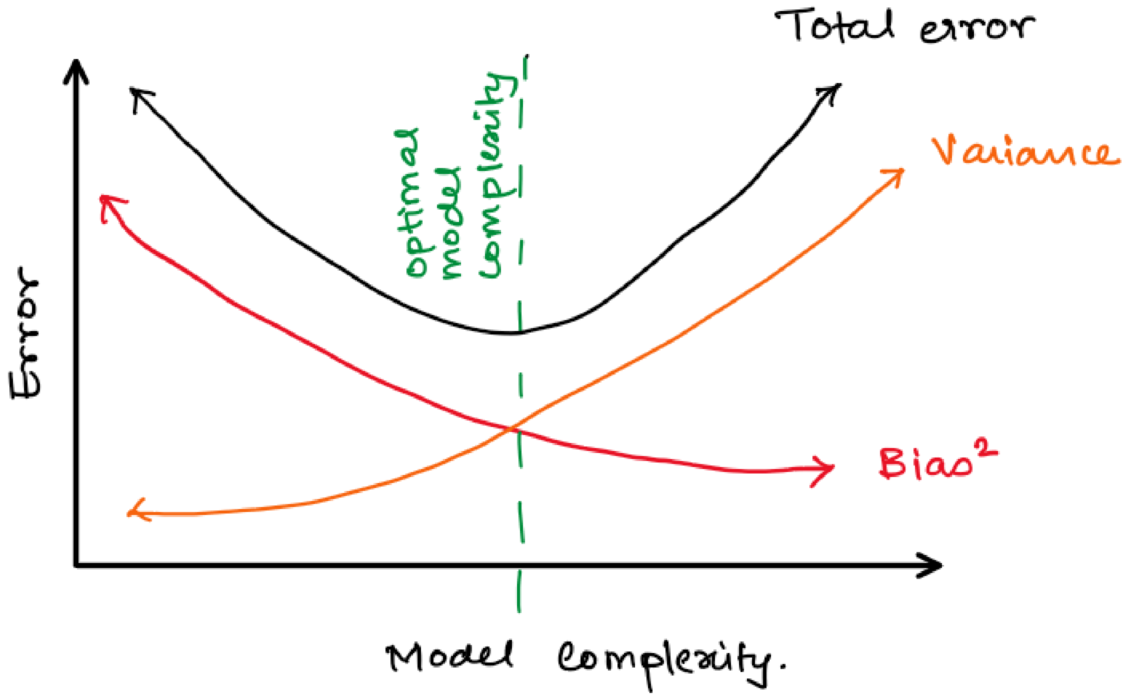
Suppose we have a model. Its Jtrain =1% and Jtest=11%. This is the case of high variance. It is because the model fits the training data very well but does poorly on test. This shows that model overfits the training data.

Now suppose we have a model. Its Jtrain =15% and Jtest=16%. This is the case of high bias because model fails to fit the training data and cause underfitting. (Model fail to capture the underlying pattern)

Now assume our model has Jtrain=15% or Jtrain=30%. This is the case of high bias and high variance because the model fit both the training and test data very poorly.

Once again consider if the model has Jtrain=0.5% and Jtest=1%. This is a low bias and low variance case because model fits approximately to train and test both.

High bias causes underfitting and high variance causes overfitting. If we draw out graphically, the above bias and variance problem can be summarized as below:



This show, if the model is simple (degree of polynomial is small) then it has high bias and low variance. If the model is complex (degree of polynomial is high) It has high variance and low bias. Therefore, as

Machine Learning Engineer we need to find a balance/tradeoff complexity between high bias (underfitting) and high variance (overfitting) i.e: a tradeoff between model being too complex and too

simple. This bias and variance tradeoff is done keeping in view that the test error remain minimized.

Techniques to reduce variance:
1.Multiple algorithm for e.g. using multiple decision tree (random forest). Variance of a single tree is
high but averaging many trees reduced variance without increasing bias.
2.Regularization: By regularization we limit the value of weights. As a result, overfitting is reduced.
3. Increasing Training Data: By adding more samples to the dataset we can prevent the high variance
which gives rise to overfitting. If you have 3 degrees of polynomial, you can perfectly match (fit) around 3 samples. But we can't perfectly match a 10000 samples.

2) [4 points] What is k-fold cross-validation? Why do we need it?

Solution:
It is a resampling method used to evaluate machine learning model on a limited data. K fold is a technique in which data is divided into K-segment and using K-1 for training and 1_{st} segment for testing/Cross validation and then repeating the process using another segment for testing and the remaining K-1 segment for training, and keep repeating the process k times. There are 2 main uses of K fold cross validation.
1. When we have small sample of data, training data is limited. So how can we test our model? We don’t want to waste data for testing/cross validation if it already small for training. So, to utilize all the data for training and testing we use K-fold cross validation.
2. To best select hyper parameter alpha. For e.g. you can apply alpha$_1$ for k-1 segment for training and 1 for testing/cross validation. Then apply alpha$_2$ for k-1 for training and 1 for testing and keep repeating k times with alpha$_K$. At the end take average of all hyperparameter and select that as hyperparameter and once the hyper parameter is selected used all the data to train the model.
3. To have best model: Lets suppose you use 1_{st} segment for testing and it its biased, then the result is not reliable but if you can apply k fold cross validation then you can aggregate the result and get more reliable result and have best model.

Question 2: [6 points] Assume the following confusion matrix of a classifier. Please compute its
1) precision,
2) recall, and
3) F$_1$-score.

Predicted results

		Class 1	Class 2
Actual values	Class 1	50	30
	Class 2	40	60

Classified

		Positive	Negative
Actual	Positive	TP 50	FN 30
	Negative	FP 40	TN 60

precision $p = \frac{TP}{TP+FP}$ f1-score $= \frac{2pr}{p+r}$

Recall $r = \frac{TP}{TP+FN}$

Solving:

$$p = \frac{50}{90} = 0.556$$

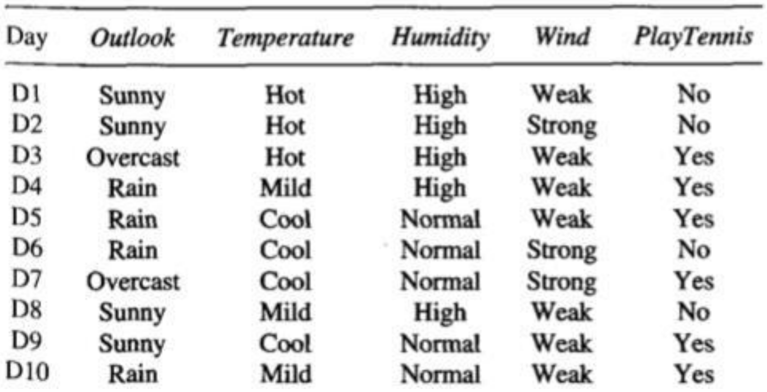
$$r = \frac{50}{80} = 0.625$$

$$f_1\text{-score} = \frac{2}{(9/5 + 8/5)} = 0.588$$

$$\boxed{f_1\text{-sore} = 0.588}$$

Question 3:

[10 points] Build a decision tree using the following training instances (using information gain approach):



Day	*Outlook*	*Temperature*	*Humidity*	*Wind*	*PlayTennis*
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes

Step ①:

$P = 6 \; ; N = 4$

Entropy of whole Dataset E(s)

$$= -\frac{6}{10} \log_2\left(\frac{6}{10}\right) - \frac{4}{10} \log_2\left(\frac{4}{10}\right)$$

$$\boxed{E(S) = 0.971}$$

step ②: Information gain for each attribute

① outlook:

Attributes	P	n	entropy
sunny	1	3	0.811
overcast	2	0	0
rain	3	1	0.811

$$I(\text{outlook}) = \frac{4}{10} \times 0.811 + \frac{4}{10} \times 0.811 + 0$$

$$= 0.6488$$

$$\text{Gain}(\text{outlook}) = E(S) - I(\text{outlook}) = 0.3221$$

② Temperature :

Attribute	p	n	entropy
Hot	1	2	0.918
mild	2	1	0.918
cool	3	1	0.811

$$I(\text{temperature}) = \frac{3}{10} \times 0.918 + \frac{3}{10} \times 0.918 + \frac{4}{10} \times 0.811$$

$$= 0.8752$$

$$\text{Gain (Temperature)} = E(s) - I(\text{temperature}) = 0.0958$$

③ Humidity :

Attributes	p	n	entropy
High	2	3	0.971
Normal	4	1	0.722

$$I(\text{Humidity}) = 0.5 \times 971 + 0.5 \times 0.722$$

$$\text{Gain(Humidity)} = 0.124$$

④ Wind :

Attribute	p	n	entropy
Weak	5	2	0.863
strong	1	2	0.918

$$I(\text{Wind}) = 0.7 \times 0.863 + 0.3 \times 0.918$$

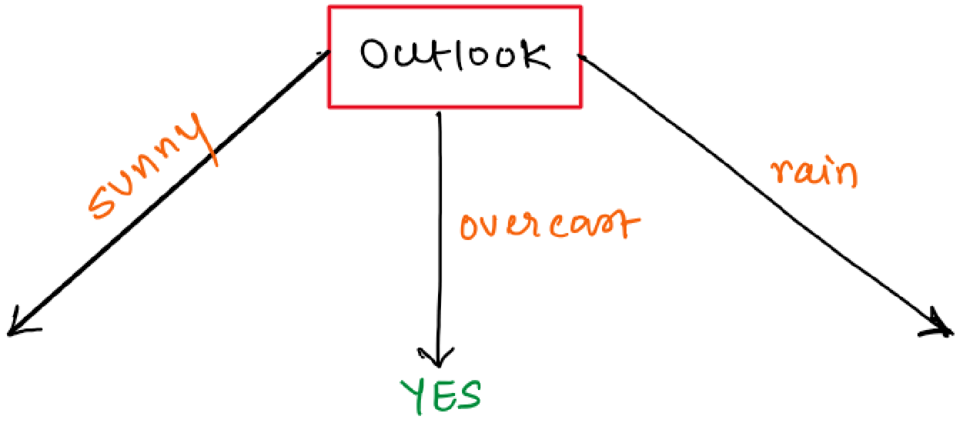
$$= 0.8795$$

$$\text{Gain (Wind)} = 0.0915$$

Comparing gain of all attributes,

attribute outlook has larger value of information gain compare to other attributes. Therefore, I choose outlook as root node.

Decision tree till now:



utlook Sunny :

step① : $P = 1$; $N = 3$

Entropy ($\text{outlook}_{\text{sunny}}$) = 0.811

step② :

① Temperature:

attribute	P	N	E
Hot	0	2	0
cool	1	0	0
mild	0	1	0

I(temp) = 0

Gain (temp) = 0.811

② Humidity :

A	P	N	E
High	0	3	0
Normal	1	0	0

I(Humidity) = 0

Gain(Humidity) = 0.811

③ Wind

A	P	N	E
Weak	1	2	0.918
strong	0	1	0

I(wind) = 0. 6885

Gain(wind) = 0. 122

Outlook (Rain) :

Step ① :

$P = 3 : N = 1$

$E = 0.811$

Step ② :

① Temp

A	P	n	E
mild	2	0	0
cool	1	1	

Gain < 0.811

② Humidity

A	P	N	E
High	1	0	0
normal	2	1	

Gain < 0.811

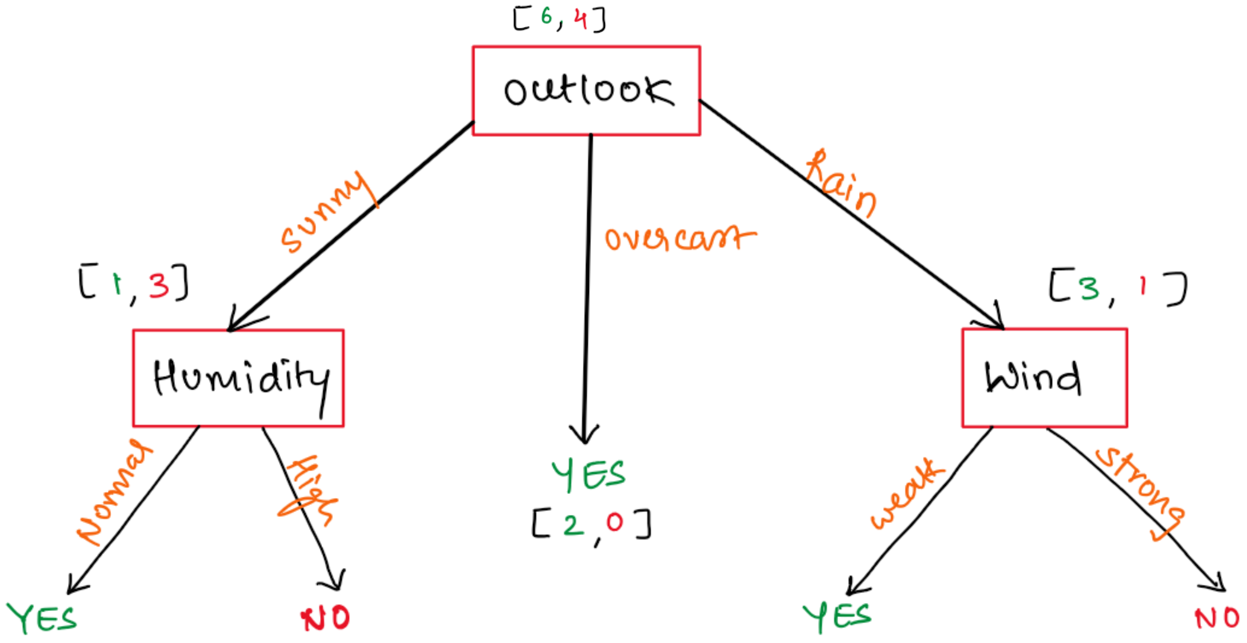
③ Wind

A	P	N	E
weak	3	0	0
strong	0	1	0

$I(\text{wind}) = 0$

$\therefore$ Gain $= 0.811$

final Decision Tree :



2) [4 points] Decide the p-value (i.e., p_{chance}) of the root node using Chi-square test.
[Hint: Please refer to page 41 – 45 of Lecture 5 slides for Chi-square test. After you have obtained the critical value *CV* (or *Q* as we used in lecture slides), using the following online tool to obtain the p-value, i.e., *P(X2 > CV)* (you need to enter degree of freedom and the critical value *CV*): https://stattrek.com/online-calculator/chi-square.aspx]

* P-value of root node (Outlook):

$$\hat{p}_i = \frac{p}{p+n} |s_i| \qquad Q = \sum_i \frac{(p_i - \hat{p}_i)^2}{\hat{p}_i} + \frac{(n_i - \hat{n}_i)^2}{\hat{n}_i}$$

$$\hat{n}_i = \frac{n}{p+n} |s_i|$$

$$\hat{p}_1 = \frac{6}{10} \times 4 = 2.4 \qquad \hat{p}_2 = \frac{6}{10} \times 2 = 1.2 \qquad \hat{p}_3 = \frac{6}{10} \times 4 = 2.4$$

$$\hat{n}_1 = \frac{4}{10} \times 4 = 1.6 \qquad \hat{n}_2 = \frac{4}{10} \times 2 = 0.8 \qquad \hat{n}_3 = \frac{4}{10} \times 4 = 1.6$$

$$Q = \frac{(1-2.4)^2}{2.4} + \frac{(3-1.6)^2}{1.6} + \frac{(2-1.2)^2}{1.2} + \frac{(0-0.8)^2}{0.8} + \frac{(3-2.4)^2}{2.4} + \frac{(1-1.6)^2}{1.6}$$

$$= 3.75$$

degree of freedom = 3-1 = 2

$$\boxed{\therefore \text{ p-value } = 0.15}$$

Question 4. [10 points] In ensemble learning, there are several popular fusion methods for Class Label type classifiers, e.g., majority vote, weighted majority vote, and naïve Bayes methods. Assuming we have 3 classifiers, and their predicted results are given in the table 1. The confusion matrix of each classifier is given in table 2. Please give the final decision using **Naïve Bayes** as the fusion method:

Table 1 Predicted results of each classifier

Sample x	Result
Classifier 1	Class 1
Classifier 2	Class 1
Classifier 3	Class 2

Table 2 Confusion matrix of each classifier

i) Classifier 1

	Class1	Class2
Class1	40	10
Class2	30	20

ii) Classifier 2

	Class1	Class2
Class1	20	30
Class2	20	30

iii) Classifier 3

	Class1	Class2
Class1	50	0
Class2	40	10

$P(\omega_1 | d_{1,1}(x)=1) = 40/70$ $\quad$ $P(\omega_2 | d_{1,1}(x)=1) = 30/70$

$P(\omega_1 | d_{2,1}(x)=1) = 20/40$ $\quad$ $P(\omega_2 | d_{2,1}(x)=1) = 20/40$

$P(\omega_1 | d_{3,2}(x)=1) = 0/70$ $\quad$ $P(\omega_2 | d_{3,2}(x)=1) = 10/1$

$\underline{\text{class 1}} = \frac{4}{7} \times \frac{1}{2} \times 0 = 0$

$\underline{\text{class 2}}: \underline{\underline{0.214}}$

predicted results will (shall) be class 2.