For Questions 1- 4, please submit a word file or a PDF file;
For Question 5 (programming question), please submit an .ipynb file.

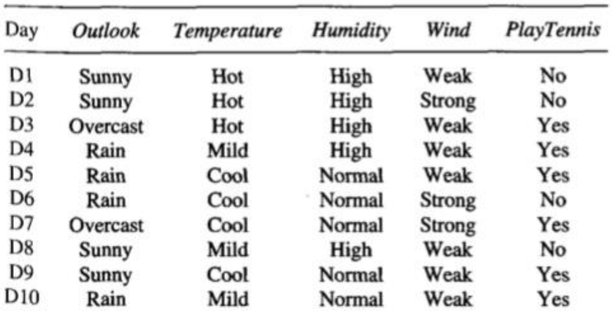
Question 1: [4 points] Explain what is the bias-variance trade-off? Describe few techniques to reduce bias and variance respectively.

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

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

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

Question 4. [10 points] The naïve Bayes method is an ensemble method as we learned in Module 5. 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 the Naïve Bayes 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

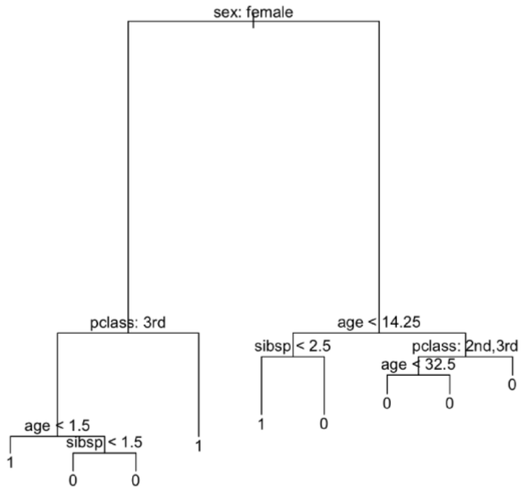
	Class1	Class2
Class1	50	0
Class2	40	10

Question 5: Programming (40 points):

Use **decision tree** and **random forest** to train the titanic.csv dataset included in the assignment.

Step 1: Read in Titanic.csv and observe a few samples, some features are categorical, and others are numerical. If some features are missing, fill them in using the average of the same feature of other samples. Take a random 80% samples for training and the rest 20% for test.

Step 2: Fit a decision tree model using independent variables 'pclass + sex + age + sibsp' and dependent variable 'survived'. Plot the full tree. Make sure 'survived' is a qualitative variable taking 1 (yes) or 0 (no) in your code. You may see a tree similar to this one (the actual structure and size of your tree can be different):



Step 3: Use the *GridSearchCV()* function to find the best parameter max_leaf_nodes to prune the tree. Plot the pruned tree which shall be smaller than the tree you obtained in Step 2.

Step 4: For the pruned tree, report its accuracy on the test set for the following:

percent survivors correctly predicted (on test set)
percent fatalities correctly predicted (on test set)

Step 5: Use the *RandomForestClassifier()* function to train a random forest using the value of max_leaf_nodes you found in Step 3. You can set n_estimators as 50. Report the accuracy of random forest on the test set for the following:

percent survivors correctly predicted (on test set)
percent fatalities correctly predicted (on test set)

Check whether there is improvement as compared to a single tree obtained in Step 4.