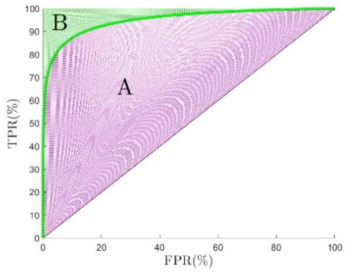
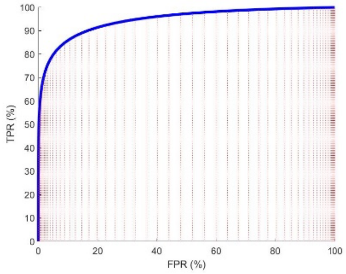
EE2211 Tutorial 10

Question 1:
We have two classifiers showing the same accuracy with the same cross-validation. The more complex model (such as a 9th-order polynomial model) is preferred over the simpler one (such as a 2nd-order polynomial model).

- a) True
- [✓ b) False]

Question 2: According to the plots below, the Gini Coefficient is equal to Two times the Area Under the ROC minus One.





[✓ a) True]
b) False

Question 3:
Suppose the binary classification problem, which you are dealing with, has highly imbalanced classes. The majority class has 99 hundred samples and the minority class has 1 hundred samples. Which of the following metric(s) would you choose for assessing the classification performance? (Select all relevant metric(s) to get full credit)

- a) Classification Accuracy
- [✓ b) Cost sensitive accuracy]
- [✓ c) Precision and recall]
- d) None of these

Question 4:
Given below is a scenario for Training error rate Tr, and Validation error rate Va for a machine learning algorithm. You want to choose a hyperparameter (P) based on Tr and Va.

P	Tr	Va
10	0.10	0.25
9	0.30	0.35
8	0.22	0.15
7	0.15	0.25
6	0.18	0.15

Which value of P will you choose based on the above table?

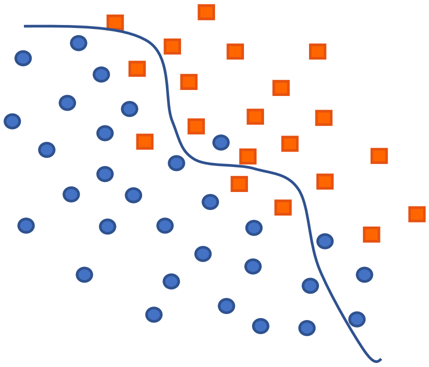
a) 10
b) 9
c) 8
d) 7
e) 6

(Binary and Multicategory Confusion Matrices)

Question 5:

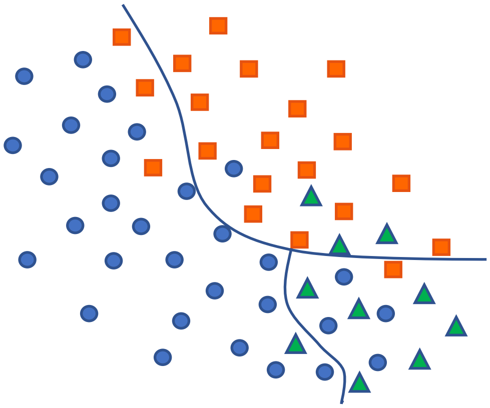
Tabulate the confusion matrices for the following classification problems.

(a) Binary problem (the class-1 and class-2 data points are respectively indicated by squares and circles)



16/20	4/30
4/20	26/30

(b) Three-category problem (the class-1, class-2 and class-3 data points are respectively indicated by squares, circles and triangles).



(5-fold Cross-validation)

Question 6:

Get the data set "from sklearn.datasets import load_iris". Perform a 5-fold Cross-validation to observe the best polynomial order (among orders 1 to 10 and without regularization) for validation prediction. Note that, you will have to partition the whole dataset for training/validation/test parts, where the size of validation set is the same as that of test. Provide a plot of the average 5-fold training and validation error rates over the polynomial orders. The randomly partitioned data sets of the 5-fold shall be maintained for reuse in evaluation of future algorithms.

(Plot the ROC and Compute the AUC for Binary Classification)

Question 7:

Download the `spambase` data set from the UCI Machine Learning repository https://archive.ics.uci.edu/ml/machine-learning-databases/spambase/ and use the following function to pack the data:

```
def load_data(Train=False):
    import csv
    data = []
    ## Read the training data
    f = open('spambase.data')
    reader = csv.reader(f)
    next(reader, None)
    for row in reader:
        data.append(row)
    f.close()
    ## x[:-1]: omit the last element of each x row
    X = np.array([x[:-1] for x in data]).astype(np.float)
    ## x[-1]: the first element from the right instead of from the left
    y = np.array([x[-1] for x in data]).astype(np.float)
    del data # free up the memory
    if Train:
        # returns X_train, X_test, y_train, y_test
        return train_test_split(X, y, test_size=0.2, random_state=8)
    else:
        return X, y
```

Randomly split the dataset into two parts, 80% for training and 20% for testing. Note that, there is no need to run cross validation here. Compute the test Classification Error Rate and the AUC based on the optimal linear regression model without regularization. In other words, use the training set to train a linear model and use the test set to check the classification performance in terms of Classification Error Rate, ROC and AUC.

Hint: to plot the ROC curve, the population of output predictions (say, values stored in vector `y_predict`) needs to be separated according to the two known output classes (0 or 1 values in the target vector `y_test`). Let `y_predict_for_PosSamples` (when `y_test==1`) and `y_predict_for_NegSamples` (when `y_test==0`) denote these two populations of prediction. Then compute the TPR (=1-FNR) and the FPR at various threshold/operating points in order to plot the ROC curve. To obtain the highest possible resolution for the ROC plot, you can set the decision threshold according to each element of the lower population of the two predicted classes of data (`y_predict_for_PosSamples` , `y_predict_for_NegSamples`).