Question 2

Basic Model

1. The data is divided into training set (80%) and test set (20%) using sci-kit learn train_test_split with random seed=98

2. [CM4] The model has been trained with sci-kit learn KNeighborsClassifier's default parameters using training set and model is tested on testing set.

Iris Dataset:

knn with Default parameters

```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score

knn = KNeighborsClassifier()
knn.fit(X_train, y_train)
y_pred = knn.predict(X_test)
a_scores=(accuracy_score(y_test, y_pred)*100)
print("Accuracy for default parameters is ",a_scores,"%")
```

```
Accuracy for default parameters is  95.23809523809523 %
```

Heart Disease Dataset:

knn on default parameters

```
knn = KNeighborsClassifier()
knn.fit(X_train_entire, y_train_entire)
y_pred = knn.predict(X_test)
accu_scores=(accuracy_score(y_test, y_pred)*100)
print("Accuracy for default parameters is ",accu_scores,"%")
```

```
Accuracy for default parameters is  86.04651162790698 %
```

For Default Parameters (K=5), the accuracies observed are as below:

Dataset	Accuracy
Iris Dataset	95.23%
Heart Disease Dataset	86.05%

3. Tuning the classifier to find the best k:

i)For Iris – 5-fold cross validation is used on the training set to train the classifier

For k in the range {1, 5, 10, 15, 20, 25, 30, 35}, sci-kit learn cross validation is utilized here.

a) The model is fit using the training set
b) The average accuracies across all folds is stored

5 fold CrossValidation for parameter tuning

```
from sklearn.model_selection import cross_val_score
k_list=[1,5,10,15,20,25,30,35]

#to store accuracy scores
accu_scores=[]

#to store variance scores
var_scores=[]


#perform 5-fold cross validation
for k in k_list:
    knn=KNeighborsClassifier(n_neighbors=k)
    scores=cross_val_score(knn,X_train,y_train,cv=5,scoring= 'accuracy')
    #taking mean/variance of accuracies

    accu_scores.append(scores.mean()*100)
    var_scores.append(scores.std())
```

```
print(accu_scores)
```

```
[91.61764705882354, 92.79411764705883, 90.44117647058823, 90.44117647058823, 84.55882352941175, 84.55882352941175, 84.632352941
17646, 85.80882352941175]
```

```
print(var_scores)
```

```
[0.030706783849736914, 0.04565933734194125, 0.06063390625908325, 0.060633906259083256, 0.027116307227332034, 0.0591900346985238
6, 0.07893374647246851, 0.06974062177281988]
```

The average accuracy and variance for K values are as below:

K	Average Accuracy across all folds	Variance
1	91.62	0.0307
5	92.79	0.0457
10	90.44	0.0606
15	90.44	0.0606
20	84.55	0.0271
25	84.55	0.0591
30	84.63	0.0789
35	85.80	0.0697

ii)For Heart Disease – Instead of 5-fold cross validation, the training set is further split into 90% train set and 10% validation set (used to make sure our model does not overfit and to tune the parameters)

Splitting the training dataset further into training(90%) and validation set(10%)

```
X_train, X_val, y_train, y_val = train_test_split(X_train_entire, y_train_entire,
                                                  test_size=0.1,
                                                  random_state=98)
print("Size of Training set-",X_train.shape)
print("Size of Validation set -",X_val.shape)
```

```
Size of Training set- (152, 22)
Size of Validation set - (17, 22)
```

The categorical features of the dataset are one-hot encoded so that order/ranking is not introduced while performing KNN.

For k in the range {1, 5, 10, 15, 20, 25, 30, 35},

a) The KNN classifier is fit on the new training set
b) Accuracy scores are produced on the **validation set**

Performing knn over a range of k on validation set

```
k=[1,5,10,15,20,25,30,35]

#to store accuracy values
a_scores=[]

for i in k:
    knn = KNeighborsClassifier(n_neighbors=i)
    knn.fit(X_train, y_train)
    y_pred_val = knn.predict(X_val)
    a_scores.append(accuracy_score(y_val, y_pred_val)*100)

a_scores
```

```
[70.58823529411765,
 76.47058823529412,
 76.47058823529412,
 70.58823529411765,
 64.70588235294117,
 58.82352941176471,
 58.82352941176471,
 58.82352941176471]
```

K	Validation Accuracy
1	70.59 %
5	76.47 %
10	76.47 %
15	70.58 %
20	64.70%
25	58.82%
30	58.82%
35	58.82%