Assignment - 1
I.Krutika
18MCMT20
MTech CS

Neural Networks Assignment 3

Introduction:

Experiment with Support Vector Machine (SVM) on real world problems. This assignment is mainly done using Scikit-learn python library

The datasets used are:

- Arcene Cancer Data set (https://archive.ics.uci.edu/ml/datasets/Arcene)
  - Number of features : 10,000
  - Number of Training Data : 100
  - Number of Testing Data(Validation Set is used) : 100
  - K-Digit PCA : 10 and 100
- Telco Customer Churn Data set (https://www.kaggle.com/blastchar/telco-customer-churn)
  - Number of features : 45
  - Number of Training Data : 7043
  - Number of Testing Data(20% of training set is used) : 1408
  - K-Digit PCA : 10 and 20

Imports Used:

Below are the imports used throughout the Assignment. Warnings are also ignored

In [1]:

```
import numpy as np
from sklearn import svm,metrics
from sklearn.preprocessing import StandardScaler
from sklearn import preprocessing
from sklearn.decomposition import PCA
from sklearn.model_selection import KFold,GridSearchCV
from sklearn.metrics import accuracy_score
import pandas as pd
import warnings
warnings.filterwarnings('ignore')
```

Arcene Cancer SVM

Extracting Data from files

Data Set	File Name
Training Data	arcene_train.data
Training Lables	arcene_train.labels
Validation Data	arcene_valid.data
Validation Lables	arcene_valid.labels

In [2]:

```
numberOfFeaures=10000
trainData=validData=trainLables=validLables=np.array([])
fileNames=['arcene_train.data','arcene_valid.data']
for file in fileNames:
    with open(file) as myFile:
        data=myFile.read().split(' ')
        while '\n' in data: data.remove('\n')
        data=np.array(data).reshape(int(len(data)/numberOfFeaures),numberOfFeaures)
        if file=='arcene_train.data':
            trainData=data
        if file=='arcene_valid.data':
            validData=data
fileNames=['arcene_train.labels','arcene_valid.labels']
for file in fileNames:
     with open(file) as myFile:
        data=myFile.read().split('\n')
        while '' in data: data.remove('')
        if file=='arcene_train.labels':
              trainLables=np.array(data)
        if file=='arcene_valid.labels':
              validLables=np.array(data)
print("Train Data Shape : ",trainData.shape)
print("Train Lables Shape : ",trainLables.shape)
print("Valid Lables Shape : ",validLables.shape)
print("Valid Data Shape : ",validData.shape)
```

```
Train Data Shape :  (100, 10000)
Train Lables Shape :  (100,)
Valid Lables Shape :  (100,)
Valid Data Shape :  (100, 10000)
```

K-Digit PCA

Below code block(method) will do the following

- Standardizing data before applying PCA
- Normalizing the data before applying K-Digit PCA(k1,k2 are passed to the function)
- Converting 10000 features to 10 features

In [3]:

```
def getPca10And100(train,test,k1,k2):
    train = preprocessing.MinMaxScaler().fit_transform(train)
    test = preprocessing.MinMaxScaler().fit_transform(test)
    scaler = StandardScaler()
    train = scaler.fit_transform(train)
    test = scaler.transform(test)
    train = preprocessing.MinMaxScaler().fit_transform(train)
    test = preprocessing.MinMaxScaler().fit_transform(test)
    pca10 = PCA(n_components=k1)
    pca10.fit(train)
    pcaTrain10 = pca10.transform(train)
    pcaTest10 = pca10.transform(test)

    pca100 = PCA(n_components=k2)
    pca100.fit(train)
    pcaTrain100 = pca100.transform(train)
    pcaTest100 = pca100.transform(test)
    return [pcaTrain10,pcaTrain100],[pcaTest10,pcaTest100]
```

Below are the methods that are used in common for both the data sets.

getAccuracy: Returns the accuracy by counting the number of correct outputs

getClassifiers: Returns two SVM classifiers.

1) Linear Kernear
2) RBF Kernel

generalizationError: Used to report Generalization Error inequality based on Mean Error of errors at each fold and Mean of number of Support vectors at each fold.

$$E⟦Out Sample Error⟧ <=\frac{E⟦Number of SVs⟧}{N-1}$$

measuresOfConMatrix: Gives us various metrics of Confusion Matrix such as, Precision, Recall/Sensitivity, Specificity and Accuracy.

performCrossValidation: Performs cross validation and returns Mean Error Matrix, Mean SUpport Vectors Matrix, index of PCA Data Sets and index of Classifiers at which minimum cross validation error has occured.

In [4]:

```
def getAccuracy(predict,actual):
    correct=0
    for i in range(actual.shape[0]):
        if predict[i]==actual[i]:
            correct+=1
    return correct/actual.shape[0]


def getClassifiers():
    return [svm.SVC(gamma='auto',kernel='linear'),svm.SVC(gamma='auto',kernel='rbf')]

def generalizationError(classifiersSize,dataSetsSize,meanError,meanSupportVectors,numberOfInputs):
    print("E[Out_Sample Error]\t<=\tE[[Number of SVs]]/(N-1)")
    print("------------------------------------------------------------")
    for i in range(classifiersSize):
        for j in range(dataSetsSize):
            print("|\t%0.2f\t" % meanError[i][j],"\t<=\t",(meanSupportVectors[i][j]/(numberOfInputs-1)),"\t|
")
            print("------------------------------------------------------------")
class measuresOfConMatrix:
    def __init__(self,cm):
        self.cm=cm
        self.TP=self.getTP()
        self.TN=self.getTN()
        self.FP=self.getFP()
        self.FN=self.getFN()
        self.tot=self.TP+self.TN+self.FP+self.FN
    def errorRate(self):
        return np.around(((self.FP+self.FN)/self.tot),decimals=3)

    def accuracy(self):
        return np.around((self.TP+self.TN)/self.tot,decimals=3)

    def precision(self):
        return np.around((self.TP/(self.TP+self.FP)),decimals=3)
    def recall(self):
        return np.around((self.TP/(self.TP+self.FN)),decimals=3)
    def specificity(self):
        return np.around((self.TN/(self.FP+self.TN)),decimals=3)

    def getTP(self):
        return np.diag(self.cm)
    def getFP(self):
        FP = []
        for i in range(2):
            FP.append(sum(self.cm[:,i]) - self.cm[i,i])
            return np.array(FP)
    def getFN(self):
        FN = []
        for i in range(2):
            FN.append(sum(self.cm[i,:]) - self.cm[i,i])
        return np.array(FN)
    def getTN(self):
        TN = []
        for i in range(2):
```

```
        for i in range(2):
            temp = np.delete(self.cm, i, 0)   # delete ith row
            temp = np.delete(temp, i, 1)  # delete ith column
            TN.append(sum(sum(temp)))
        return np.array(TN)

def performCrossValidation(lables,dataSets):
    numberOfFolds=5
    errorTrack=[]
    minError=99999
    minClassifierIndex=minPcaDataIndex=-1
    classifierIndex=pcaIndex=0
    meanErrorOutputSamples=np.zeros(len(classifiers)*len(pcaDataSets)).reshape(len(classifiers),len(pcaDataS
ets))
    meanOfSupportVectors=np.zeros(len(classifiers)*len(pcaDataSets)).reshape(len(classifiers),len(pcaDataSet
s))
    for classifier in classifiers:
        pcaIndex=0
        for pcaData in dataSets:
            kfold = KFold(numberOfFolds,False, 1)
            error=[]
            supportVectors=[]
            for train, test in kfold.split(pcaData):
                trainingSet=pcaData[train]
                testingSet=pcaData[test]
                lablesTrain=lables[train]
                lablesTest=lables[test]
                classifier.fit(trainingSet, lablesTrain)
                predicted=classifier.predict(testingSet)
                accuracy=getAccuracy(predicted,lablesTest)
                err=1-accuracy
                error.append(err)
                supportVectors.append(np.sum(classifier.n_support_))
                cm=metrics.confusion_matrix(lablesTest, predicted)
                measuresCm=measuresOfConMatrix(cm)
                acc=measuresCm.accuracy()
                spe=measuresCm.specificity()
                print("Kernel : ",classifier.kernel,", K-Digit PCA :",pcaData.shape[1])
                print("Support Vectors : ",classifier.n_support_," : Total number of support vectors : ",np.
sum(classifier.n_support_))
                print("Confusion matrix:\n%s" % cm)
                print("Classification report for classifier %s:\n%s\n"
                      % (classifier, metrics.classification_report(lablesTest, predicted)))
                print("\t\tAccuracy\tSpecificity")
                print("\t-1\t","%.2f\t"%acc[0],"\t%.2f"%spe[0])
                print("\t 1\t","%.2f\t"%acc[1],"\t%.2f"%spe[1])
                print("++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++")
            meanError=np.mean(np.array(error))
            meanSupportVectos=np.mean(np.array(supportVectors))
            meanErrorOutputSamples[classifierIndex][pcaIndex]=meanError
            meanOfSupportVectors[classifierIndex][pcaIndex]=meanSupportVectos
            print("Mean Cross Validation Error : %0.2f" % meanError)
            errorTrack.append(meanError)
            if minError > meanError:
                minError=meanError
                minClassifierIndex=classifierIndex
                minPcaDataIndex=pcaIndex
            pcaIndex+=1
            print("------------------------------------------------------------------------------------")
            print("------------------------------------------------------------------------------------")
            print("------------------------------------------------------------------------------------")
            print("------------------------------------------------------------------------------------")
            print("------------------------------------------------------------------------------------")
        classifierIndex+=1
    return minClassifierIndex,minPcaDataIndex,meanErrorOutputSamples,meanOfSupportVectors
```

Grid Search Model Selection

This process is done in order to select the appropriate C and Gamma values so that the number of support vectors reduces and GridSearchCV gives the best combination of kernel,gamma and C values

In [5]:

```
pcaDataSets,pcaValidationSets=getPca10And100(trainData,validData,10,100)
#Create a dictionary of possible parameters
params_grid = {'C': [0.001,0.005, 0.01,0.05, 0.1,0.5, 1, 5,10,50, 100,1000],
          'gamma': [0.0001, 0.001, 0.01, 0.1],
          'kernel':['linear','rbf'] }

#Create the GridSearchCV object
clf = GridSearchCV(svm.SVC(class_weight='balanced'), params_grid,cv=5)

for i in range(len(pcaDataSets)):
    #Fit the data with the best possible parameters
    clf.fit(pcaDataSets[i], trainLables)
    #Print the best estimator with it's parameters
    print(clf.best_score_)
    print(clf.best_estimator_)
    bestClassifier=clf.best_estimator_
    bestPcaData=pcaDataSets[i]
    pcaValidationSet=pcaValidationSets[i]
    bestClassifier.fit(bestPcaData, trainLables)
    predicted=bestClassifier.predict(bestPcaData)
    trainAccuracy=getAccuracy(predicted,trainLables)
    predicted=bestClassifier.predict(pcaValidationSet)
    testAccuracy=getAccuracy(predicted,validLables)
    cancerCm=metrics.confusion_matrix(validLables, predicted)
    measuresCancer=measuresOfConMatrix(cancerCm)
    cancerAcc=measuresCancer.accuracy()
    cancerSpe=measuresCancer.specificity()
    marginCount=0
    nonMarginCount=0
    for dualCoeff in (bestClassifier.dual_coef_[0]):
        if(abs(dualCoeff)<bestClassifier.C):
            marginCount+=1
        elif(abs(dualCoeff)==bestClassifier.C):
            nonMarginCount+=1

    print("Kernel : ",bestClassifier.kernel,", K-Digit PCA :",bestPcaData.shape[1])
    print("Support Vectors : ",bestClassifier.n_support_," : Total number of support vectors : ",np.sum(best
Classifier.n_support_))
    print("No. of Margin SVs : ",marginCount,", No. of Non Margin SVs : ",nonMarginCount)
    print("Train Data set Accuracy : %0.2f" % (trainAccuracy*100),"%")
    print("Validation Data set Accuracy : %0.2f" % (testAccuracy*100),"%")
    print("Confusion matrix:\n%s" % cancerCm)
    print("Classification report for classifier %s:\n%s\n"
          % (bestClassifier, metrics.classification_report(validLables, predicted)))
    print("\t\tAccuracy\tSpecificity")
    print("\t-1\t","%.2f\t"%cancerAcc[0],"\t%.2f"%cancerSpe[0])
    print("\t 1\t","%.2f\t"%cancerAcc[1],"\t%.2f"%cancerSpe[1])
    print("++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++")
```

```
0.83
SVC(C=100, cache_size=200, class_weight='balanced', coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma=0.001, kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False)
Kernel :  rbf , K-Digit PCA : 10
Support Vectors :  [27 26]  : Total number of support vectors :  53
No. of Margin SVs :  43 , No. of Non Margin SVs :  0
Train Data set Accuracy : 89.00 %
Validation Data set Accuracy : 79.00 %
Confusion matrix:
[[43 13]
 [ 8 36]]
Classification report for classifier SVC(C=100, cache_size=200, class_weight='balanced', coef0=
0.0,
  decision_function_shape='ovr', degree=3, gamma=0.001, kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.84      0.77      0.80        56
          1       0.73      0.82      0.77        44

avg / total       0.80      0.79      0.79       100


                Accuracy        Specificity
        -1       0.79           0.82
         1       0.83           0.84
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
0.89
SVC(C=0.05, cache_size=200, class_weight='balanced', coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma=0.0001, kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False)
Kernel :  linear , K-Digit PCA : 100
Support Vectors :  [49 44]  : Total number of support vectors :  93
No. of Margin SVs :  93 , No. of Non Margin SVs :  0
Train Data set Accuracy : 100.00 %
Validation Data set Accuracy : 86.00 %
Confusion matrix:
[[48  8]
 [ 6 38]]
Classification report for classifier SVC(C=0.05, cache_size=200, class_weight='balanced', coef0
=0.0,
  decision_function_shape='ovr', degree=3, gamma=0.0001, kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.89      0.86      0.87        56
          1       0.83      0.86      0.84        44

avg / total       0.86      0.86      0.86       100


                Accuracy        Specificity
        -1       0.86           0.86
         1       0.88           0.89
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

Cross Validation

Below code block uses ***getClassifiers, getPca10And100, performCrossValidation*** and prints the details for each of the classifier and pcaDataSet combination for each fold.

a) Precision, Recall/Sensitivity, Specificity and Accuracy.

b) Number of Support Vectors

c) Confusion Matrix

d) SVM Used

e) At the end of 5 folds it displays Mean Cross Validation Error

In [6]:

```
classifiers=getClassifiers()
pcaDataSets,pcaValidationSets=getPca10And100(trainData,validData,10,100)
minClassifierIndex,minPcaDataIndex,meanErrorCancer,meanSupportVectorsCancer=performCrossValidation(trainLabl
es,pcaDataSets)
```

```
Kernel :  linear , K-Digit PCA : 10
Support Vectors :  [26 28]  : Total number of support vectors :  54
Confusion matrix:
[[10  3]
 [ 2  5]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.83      0.77      0.80        13
          1       0.62      0.71      0.67         7

avg / total       0.76      0.75      0.75        20


                Accuracy        Specificity
        -1       0.75           0.71
         1       0.79           0.83
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  linear , K-Digit PCA : 10
Support Vectors :  [25 26]  : Total number of support vectors :  51
Confusion matrix:
[[6 6]
 [2 6]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.75      0.50      0.60        12
          1       0.50      0.75      0.60         8

avg / total       0.65      0.60      0.60        20


                Accuracy        Specificity
        -1       0.60           0.75
         1       0.75           0.75
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  linear , K-Digit PCA : 10
Support Vectors :  [24 24]  : Total number of support vectors :  48
Confusion matrix:
[[6 5]
 [3 6]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.67      0.55      0.60        11
          1       0.55      0.67      0.60         9

avg / total       0.61      0.60      0.60        20


                Accuracy        Specificity
        -1       0.60           0.67
         1       0.67           0.67
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  linear , K-Digit PCA : 10
Support Vectors :  [26 24]  : Total number of support vectors :  50
Confusion matrix:
[[7 4]
 [3 6]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support
```

```
          -1       0.70      0.64      0.67        11
           1       0.60      0.67      0.63         9

avg / total       0.65      0.65      0.65        20


                Accuracy        Specificity
        -1       0.65           0.67
         1       0.68           0.70
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  linear , K-Digit PCA : 10
Support Vectors :  [28 26]  : Total number of support vectors :  54
Confusion matrix:
[[6 3]
 [3 8]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

          -1       0.67      0.67      0.67         9
           1       0.73      0.73      0.73        11

avg / total       0.70      0.70      0.70        20


                Accuracy        Specificity
        -1       0.70           0.73
         1       0.70           0.67
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Mean Cross Validation Error : 0.34
------------------------------------------------------------------------------
------------------------------------------------------------------------------
------------------------------------------------------------------------------
------------------------------------------------------------------------------
------------------------------------------------------------------------------
Kernel :  linear , K-Digit PCA : 100
Support Vectors :  [40 36]  : Total number of support vectors :  76
Confusion matrix:
[[12  1]
 [ 0  7]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

          -1       1.00      0.92      0.96        13
           1       0.88      1.00      0.93         7

avg / total       0.96      0.95      0.95        20


                Accuracy        Specificity
        -1       0.95           1.00
         1       1.00           1.00
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  linear , K-Digit PCA : 100
Support Vectors :  [38 36]  : Total number of support vectors :  74
Confusion matrix:
[[10  2]
 [ 1  7]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

          -1       0.91      0.83      0.87        12
           1       0.78      0.88      0.82         8

avg / total       0.86      0.85      0.85        20


                Accuracy        Specificity
        -1       0.85           0.88
         1       0.90           0.91
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

```
Kernel :  linear , K-Digit PCA : 100
Support Vectors :  [39 35]  : Total number of support vectors :  74
Confusion matrix:
[[9 2]
 [4 5]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.69      0.82      0.75        11
          1       0.71      0.56      0.63         9

avg / total       0.70      0.70      0.69        20


                Accuracy        Specificity
        -1       0.70           0.56
         1       0.64           0.69
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  linear , K-Digit PCA : 100
Support Vectors :  [40 35]  : Total number of support vectors :  75
Confusion matrix:
[[10  1]
 [ 1  8]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.91      0.91      0.91        11
          1       0.89      0.89      0.89         9

avg / total       0.90      0.90      0.90        20


                Accuracy        Specificity
        -1       0.90           0.89
         1       0.90           0.91
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  linear , K-Digit PCA : 100
Support Vectors :  [41 33]  : Total number of support vectors :  74
Confusion matrix:
[[9 0]
 [2 9]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.82      1.00      0.90         9
          1       1.00      0.82      0.90        11

avg / total       0.92      0.90      0.90        20


                Accuracy        Specificity
        -1       0.90           0.82
         1       0.82           0.82
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Mean Cross Validation Error : 0.14
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
Kernel :  rbf , K-Digit PCA : 10
Support Vectors :  [43 36]  : Total number of support vectors :  79
Confusion matrix:
[[10  3]
 [ 2  5]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support
```

```
         -1       0.83      0.77      0.80        13
          1       0.62      0.71      0.67         7

avg / total       0.76      0.75      0.75        20


                Accuracy        Specificity
        -1       0.75           0.71
         1       0.79           0.83
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  rbf , K-Digit PCA : 10
Support Vectors :  [44 35]  : Total number of support vectors :  79
Confusion matrix:
[[10  2]
 [ 4  4]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.71      0.83      0.77        12
          1       0.67      0.50      0.57         8

avg / total       0.70      0.70      0.69        20


                Accuracy        Specificity
        -1       0.70           0.50
         1       0.64           0.71
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  rbf , K-Digit PCA : 10
Support Vectors :  [45 34]  : Total number of support vectors :  79
Confusion matrix:
[[10  1]
 [ 6  3]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.62      0.91      0.74        11
          1       0.75      0.33      0.46         9

avg / total       0.68      0.65      0.62        20


                Accuracy        Specificity
        -1       0.65           0.33
         1       0.52           0.62
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  rbf , K-Digit PCA : 10
Support Vectors :  [45 34]  : Total number of support vectors :  79
Confusion matrix:
[[9 2]
 [5 4]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.64      0.82      0.72        11
          1       0.67      0.44      0.53         9

avg / total       0.65      0.65      0.64        20


                Accuracy        Specificity
        -1       0.65           0.44
         1       0.56           0.64
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  rbf , K-Digit PCA : 10
Support Vectors :  [46 31]  : Total number of support vectors :  77
Confusion matrix:
[[9 0]
 [7 4]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
```

```
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.56      1.00      0.72         9
          1       1.00      0.36      0.53        11

avg / total       0.80      0.65      0.62        20


                Accuracy        Specificity
        -1       0.65           0.36
         1       0.48           0.56
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Mean Cross Validation Error : 0.32
----------------------------------------------------------------------------
----------------------------------------------------------------------------
----------------------------------------------------------------------------
----------------------------------------------------------------------------
----------------------------------------------------------------------------
Kernel :  rbf , K-Digit PCA : 100
Support Vectors :  [43 37]  : Total number of support vectors :  80
Confusion matrix:
[[12  1]
 [ 2  5]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.86      0.92      0.89        13
          1       0.83      0.71      0.77         7

avg / total       0.85      0.85      0.85        20


                Accuracy        Specificity
        -1       0.85           0.71
         1       0.81           0.86
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  rbf , K-Digit PCA : 100
Support Vectors :  [44 36]  : Total number of support vectors :  80
Confusion matrix:
[[11  1]
 [ 3  5]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.79      0.92      0.85        12
          1       0.83      0.62      0.71         8

avg / total       0.80      0.80      0.79        20


                Accuracy        Specificity
        -1       0.80           0.62
         1       0.73           0.79
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  rbf , K-Digit PCA : 100
Support Vectors :  [45 35]  : Total number of support vectors :  80
Confusion matrix:
[[11  0]
 [ 8  1]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.58      1.00      0.73        11
          1       1.00      0.11      0.20         9

avg / total       0.77      0.60      0.49        20


                Accuracy        Specificity
```

```
        -1       0.60           0.11
         1       0.43           0.58
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  rbf , K-Digit PCA : 100
Support Vectors :  [45 35]  : Total number of support vectors :  80
Confusion matrix:
[[10  1]
 [ 7  2]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.59      0.91      0.71        11
          1       0.67      0.22      0.33         9

avg / total       0.62      0.60      0.54        20


                Accuracy        Specificity
        -1       0.60           0.22
         1       0.46           0.59
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  rbf , K-Digit PCA : 100
Support Vectors :  [47 33]  : Total number of support vectors :  80
Confusion matrix:
[[ 9  0]
 [11  0]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.45      1.00      0.62         9
          1       0.00      0.00      0.00        11

avg / total       0.20      0.45      0.28        20


                Accuracy        Specificity
        -1       0.45           0.00
         1       0.29           0.45
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Mean Cross Validation Error : 0.34
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
```

Training and Testing with best SVM model after Cross Validation

Now we choose the combination of SVM classifier with PCA dataset that has given us the minimum cross validation error in order to test the validation error.

It is observed that SVM Classifier with Linear Kernel and 100 digit PCA has given the least **cross validation error of 14%**

Below is the code that chooses the best and performs training on the entire Arcene dataset and then Tests the Validation Set.

Accuracy on Test Set : 86%

In [7]:

```
bestClassifier=classifiers[minClassifierIndex]
bestPcaData=pcaDataSets[minPcaDataIndex]
pcaValidationSet=pcaValidationSets[minPcaDataIndex]

bestClassifier.fit(bestPcaData, trainLables)
predicted=bestClassifier.predict(bestPcaData)
trainAccuracy=getAccuracy(predicted,trainLables)
predicted=bestClassifier.predict(pcaValidationSet)
testAccuracy=getAccuracy(predicted,validLables)
cancerCm=metrics.confusion_matrix(validLables, predicted)
measuresCancer=measuresOfConMatrix(cancerCm)
cancerAcc=measuresCancer.accuracy()
cancerSpe=measuresCancer.specificity()
print(bestClassifier.dual_coef_[0].shape)
marginCount=0
nonMarginCount=0
for dualCoeff in (bestClassifier.dual_coef_[0]):
    if(abs(dualCoeff)<bestClassifier.C):
        marginCount+=1
    elif(abs(dualCoeff)==bestClassifier.C):
        nonMarginCount+=1

print("Kernel : ",bestClassifier.kernel,", K-Digit PCA :",bestPcaData.shape[1])
print("Support Vectors : ",bestClassifier.n_support_," : Total number of support vectors : ",np.sum(bestClas
sifier.n_support_))
print("No. of Margin SVs : ",marginCount,", No. of Non Margin SVs : ",nonMarginCount)
print("Train Data set Accuracy : %0.2f" % (trainAccuracy*100),"%")
print("Validation Data set Accuracy : %0.2f" % (testAccuracy*100),"%")
print("Confusion matrix:\n%s" % cancerCm)
print("Classification report for classifier %s:\n%s\n"
      % (bestClassifier, metrics.classification_report(validLables, predicted)))
print("\t\tAccuracy\tSpecificity")
print("\t-1\t","%.2f\t"%cancerAcc[0],"\t%.2f"%cancerSpe[0])
print("\t 1\t","%.2f\t"%cancerAcc[1],"\t%.2f"%cancerSpe[1])
print("++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++")
```

```
(93,)
Kernel :  linear , K-Digit PCA : 100
Support Vectors :  [49 44]  : Total number of support vectors :  93
No. of Margin SVs :  93 , No. of Non Margin SVs :  0
Train Data set Accuracy : 100.00 %
Validation Data set Accuracy : 86.00 %
Confusion matrix:
[[48  8]
 [ 6 38]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.89      0.86      0.87        56
          1       0.83      0.86      0.84        44

avg / total       0.86      0.86      0.86       100


                Accuracy        Specificity
        -1       0.86           0.86
         1       0.88           0.89
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

Generalization Error for Arcene Dataset

In the below code block it is observed that the all the SVM classifiers follow satisfy the inequality for Arcene Dataset $$E⟦Out Sample Error⟧ <=\frac{E⟦Number of SVs⟧}{N-1}$$

In [8]:

```
generalizationError(len(classifiers),len(pcaDataSets),meanErrorCancer,meanSupportVectorsCancer,pcaDataSets[0
].shape[0])
```

```
E[Out_Sample Error]     <=      E[[Number of SVs]]/(N-1)
-----------------------------------------------------------
|       0.34            <=       0.5191919191919192     |
-----------------------------------------------------------
|       0.14            <=       0.7535353535353535     |
-----------------------------------------------------------
|       0.32            <=       0.7939393939393938     |
-----------------------------------------------------------
|       0.34            <=       0.8080808080808081     |
-----------------------------------------------------------
```

Telco Customer Churn SVM

Reading Data from CSV

Here the data is present in csv format. It is read using Pandas Library.

The dataset is divided into 80% Traing data and 20% Testing Data

In [9]:

```
data = pd.read_csv("WA_Fn-UseC_-Telco-Customer-Churn.csv")
df = pd.DataFrame(data)
df=pd.get_dummies(df,columns=['gender','Partner','Dependents','PhoneService','MultipleLines','InternetServic
e','OnlineSecurity','OnlineBackup','DeviceProtection','TechSupport','StreamingTV','StreamingMovies','Contrac
t','PaperlessBilling','PaymentMethod'])
df['TotalCharges'] = pd.to_numeric(df['TotalCharges'], errors='coerce')
df.dropna(inplace=True)
df.replace(('Yes', 'No'), (1, -1), inplace=True)
df1 = df.drop(['Churn'], axis=1)
churnTrain=df1.iloc[:5634,1:]
churnTest=df1.iloc[5634:,1:]
churnTrainLables=df['Churn'][:5634]
churnTestLables=df['Churn'][5634:]
```

Grid Search Model Selection

This process is done in order to select the appropriate C and Gamma values so that the number of support vectors reduces and GridSearchCV gives the best combination of kernel,gamma and C values

In [10]:

```
pcaDataChurn,pcaValidationChurn=getPca10And100(churnTrain,churnTest,10,18)
#Create a dictionary of possible parameters
params_grid = {'C': [0.001, 0.01, 0.1,1,10,100],
          'gamma': [0.001, 0.01, 0.1],
          'kernel':['linear','rbf'] }

#Create the GridSearchCV object
clf = GridSearchCV(svm.SVC(class_weight='balanced'), params_grid)

for i in range(len(pcaDataChurn)):
    #Fit the data with the best possible parameters
    clf.fit(pcaDataChurn[i], churnTrainLables)
    #Print the best estimator with it's parameters
    print(clf.best_score_)
    print(clf.best_estimator_)
    bestClassifier=clf.best_estimator_
    bestPcaData=pcaDataChurn[i]
    pcaValidationSet=pcaValidationChurn[i]
    bestClassifier.fit(bestPcaData, churnTrainLables)
    predicted=bestClassifier.predict(bestPcaData)
    trainAccuracy=getAccuracy(predicted,churnTrainLables.values)
    predicted=bestClassifier.predict(pcaValidationSet)
    testAccuracy=getAccuracy(predicted,churnTestLables.values)
    cancerCm=metrics.confusion_matrix(churnTestLables, predicted)
    measuresCancer=measuresOfConMatrix(cancerCm)
    cancerAcc=measuresCancer.accuracy()
    cancerSpe=measuresCancer.specificity()
    marginCount=0
    nonMarginCount=0
    for dualCoeff in (bestClassifier.dual_coef_[0]):
        if(abs(dualCoeff)<bestClassifier.C):
            marginCount+=1
        elif(abs(dualCoeff)==bestClassifier.C):
            nonMarginCount+=1
    print("Kernel : ",bestClassifier.kernel,", K-Digit PCA :",bestPcaData.shape[1])
    print("Support Vectors : ",bestClassifier.n_support_," : Total number of support vectors : ",np.sum(best
Classifier.n_support_))
    print("No. of Margin SVs : ",marginCount,", No. of Non Margin SVs : ",nonMarginCount)
    print("Train Data set Accuracy : %0.2f" % (trainAccuracy*100),"%")
    print("Validation Data set Accuracy : %0.2f" % (testAccuracy*100),"%")
    print("Confusion matrix:\n%s" % cancerCm)
    print("Classification report for classifier %s:\n%s\n"
          % (bestClassifier, metrics.classification_report(churnTestLables, predicted)))
    print("\t\tAccuracy\tSpecificity")
    print("\t-1\t","%.2f\t"%cancerAcc[0],"\t%.2f"%cancerSpe[0])
    print("\t 1\t","%.2f\t"%cancerAcc[1],"\t%.2f"%cancerSpe[1])
    print("++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++")
```

```
0.7328718494852681
SVC(C=10, cache_size=200, class_weight='balanced', coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma=0.1, kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False)
Kernel :  rbf , K-Digit PCA : 10
Support Vectors :  [2205  835]  : Total number of support vectors :  3040
No. of Margin SVs :  2284 , No. of Non Margin SVs :  0
Train Data set Accuracy : 75.51 %
Validation Data set Accuracy : 71.67 %
Confusion matrix:
[[717 304]
 [ 92 285]]
Classification report for classifier SVC(C=10, cache_size=200, class_weight='balanced', coef0=0
.0,
  decision_function_shape='ovr', degree=3, gamma=0.1, kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.89      0.70      0.78      1021
          1       0.48      0.76      0.59       377

avg / total       0.78      0.72      0.73      1398


                Accuracy        Specificity
        -1       0.72           0.76
         1       0.84           0.89
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
0.7337593184238551
SVC(C=1, cache_size=200, class_weight='balanced', coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma=0.1, kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False)
Kernel :  rbf , K-Digit PCA : 18
Support Vectors :  [2221  856]  : Total number of support vectors :  3077
No. of Margin SVs :  2319 , No. of Non Margin SVs :  0
Train Data set Accuracy : 75.42 %
Validation Data set Accuracy : 73.53 %
Confusion matrix:
[[742 279]
 [ 91 286]]
Classification report for classifier SVC(C=1, cache_size=200, class_weight='balanced', coef0=0.
0,
  decision_function_shape='ovr', degree=3, gamma=0.1, kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.89      0.73      0.80      1021
          1       0.51      0.76      0.61       377

avg / total       0.79      0.74      0.75      1398


                Accuracy        Specificity
        -1       0.73           0.76
         1       0.85           0.89
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

Cross Validation

Performs Cross Validation and Prints the same details as done in earlier data set

In [11]:

```
classifiers=getClassifiers()
pcaDataChurn,pcaValidationChurn=getPca10And100(churnTrain,churnTest,10,18)
minClassifierChurn,minPcaDataChurn,meanErrorChurn,meanSupportVectorsChurn=performCrossValidation(churnTrainL
ables.values,pcaDataChurn)
```

```
Kernel :  linear , K-Digit PCA : 10
Support Vectors :  [1105 1103]  : Total number of support vectors :  2208
Confusion matrix:
[[740  92]
 [152 143]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
```

```
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.83      0.89      0.86       832
          1       0.61      0.48      0.54       295

avg / total       0.77      0.78      0.78      1127


                Accuracy        Specificity
        -1       0.78           0.48
         1       0.74           0.83
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  linear , K-Digit PCA : 10
Support Vectors :  [1118 1114]  : Total number of support vectors :  2232
Confusion matrix:
[[744  87]
 [128 168]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.85      0.90      0.87       831
          1       0.66      0.57      0.61       296

avg / total       0.80      0.81      0.80      1127


                Accuracy        Specificity
        -1       0.81           0.57
         1       0.78           0.85
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  linear , K-Digit PCA : 10
Support Vectors :  [1117 1111]  : Total number of support vectors :  2228
Confusion matrix:
[[742  86]
 [145 154]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.84      0.90      0.87       828
          1       0.64      0.52      0.57       299

avg / total       0.78      0.80      0.79      1127


                Accuracy        Specificity
        -1       0.80           0.52
         1       0.76           0.84
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  linear , K-Digit PCA : 10
Support Vectors :  [1083 1076]  : Total number of support vectors :  2159
Confusion matrix:
[[714 120]
 [134 159]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.84      0.86      0.85       834
          1       0.57      0.54      0.56       293

avg / total       0.77      0.77      0.77      1127


                Accuracy        Specificity
        -1       0.78           0.54
         1       0.77           0.84
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  linear , K-Digit PCA : 10
Support Vectors :  [1092 1088]  : Total number of support vectors :  2180
```

```
Confusion matrix:
[[733  84]
 [155 154]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.83      0.90      0.86       817
          1       0.65      0.50      0.56       309

avg / total       0.78      0.79      0.78      1126


                Accuracy        Specificity
        -1       0.79           0.50
         1       0.74           0.82
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Mean Cross Validation Error : 0.21
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
Kernel :  linear , K-Digit PCA : 18
Support Vectors :  [1092 1086]  : Total number of support vectors :  2178
Confusion matrix:
[[727 105]
 [138 157]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.84      0.87      0.86       832
          1       0.60      0.53      0.56       295

avg / total       0.78      0.78      0.78      1127


                Accuracy        Specificity
        -1       0.78           0.53
         1       0.76           0.84
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  linear , K-Digit PCA : 18
Support Vectors :  [1101 1097]  : Total number of support vectors :  2198
Confusion matrix:
[[735  96]
 [132 164]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.85      0.88      0.87       831
          1       0.63      0.55      0.59       296

avg / total       0.79      0.80      0.79      1127


                Accuracy        Specificity
        -1       0.80           0.55
         1       0.77           0.85
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  linear , K-Digit PCA : 18
Support Vectors :  [1101 1093]  : Total number of support vectors :  2194
Confusion matrix:
[[729  99]
 [136 163]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.84      0.88      0.86       828
          1       0.62      0.55      0.58       299
```

```
avg / total       0.78      0.79      0.79      1127


                Accuracy        Specificity
        -1       0.79           0.55
         1       0.77           0.84
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  linear , K-Digit PCA : 18
Support Vectors :  [1064 1058]  : Total number of support vectors :  2122
Confusion matrix:
[[712 122]
 [133 160]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.84      0.85      0.85       834
          1       0.57      0.55      0.56       293

avg / total       0.77      0.77      0.77      1127


                Accuracy        Specificity
        -1       0.77           0.55
         1       0.77           0.84
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  linear , K-Digit PCA : 18
Support Vectors :  [1074 1069]  : Total number of support vectors :  2143
Confusion matrix:
[[716 101]
 [138 171]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.84      0.88      0.86       817
          1       0.63      0.55      0.59       309

avg / total       0.78      0.79      0.78      1126


                Accuracy        Specificity
        -1       0.79           0.55
         1       0.76           0.84
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Mean Cross Validation Error : 0.21
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
Kernel :  rbf , K-Digit PCA : 10
Support Vectors :  [1117 1034]  : Total number of support vectors :  2151
Confusion matrix:
[[762  70]
 [185 110]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.80      0.92      0.86       832
          1       0.61      0.37      0.46       295

avg / total       0.75      0.77      0.75      1127


                Accuracy        Specificity
        -1       0.77           0.37
         1       0.70           0.81
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  rbf , K-Digit PCA : 10
Support Vectors :  [1130 1051]  : Total number of support vectors :  2181
Confusion matrix:
```

```
[[760  71]
 [158 138]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.83      0.91      0.87       831
          1       0.66      0.47      0.55       296

avg / total       0.78      0.80      0.78      1127


                Accuracy        Specificity
        -1       0.80           0.47
         1       0.74           0.83
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  rbf , K-Digit PCA : 10
Support Vectors :  [1132 1047]  : Total number of support vectors :  2179
Confusion matrix:
[[767  61]
 [166 133]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.82      0.93      0.87       828
          1       0.69      0.44      0.54       299

avg / total       0.79      0.80      0.78      1127


                Accuracy        Specificity
        -1       0.80           0.45
         1       0.73           0.82
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  rbf , K-Digit PCA : 10
Support Vectors :  [1095 1024]  : Total number of support vectors :  2119
Confusion matrix:
[[752  82]
 [163 130]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.82      0.90      0.86       834
          1       0.61      0.44      0.51       293

avg / total       0.77      0.78      0.77      1127


                Accuracy        Specificity
        -1       0.78           0.44
         1       0.73           0.82
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  rbf , K-Digit PCA : 10
Support Vectors :  [1117 1031]  : Total number of support vectors :  2148
Confusion matrix:
[[758  59]
 [180 129]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.81      0.93      0.86       817
          1       0.69      0.42      0.52       309

avg / total       0.77      0.79      0.77      1126


                Accuracy        Specificity
        -1       0.79           0.42
         1       0.71           0.81
```

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Mean Cross Validation Error : 0.21
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
Kernel :  rbf , K-Digit PCA : 18
Support Vectors :  [1118 1041]  : Total number of support vectors :  2159
Confusion matrix:
[[749  83]
 [166 129]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.82      0.90      0.86       832
          1       0.61      0.44      0.51       295

avg / total       0.76      0.78      0.77      1127


                Accuracy        Specificity
        -1       0.78           0.44
         1       0.73           0.82
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  rbf , K-Digit PCA : 18
Support Vectors :  [1134 1053]  : Total number of support vectors :  2187
Confusion matrix:
[[757  74]
 [157 139]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.83      0.91      0.87       831
          1       0.65      0.47      0.55       296

avg / total       0.78      0.80      0.78      1127


                Accuracy        Specificity
        -1       0.80           0.47
         1       0.74           0.83
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  rbf , K-Digit PCA : 18
Support Vectors :  [1127 1051]  : Total number of support vectors :  2178
Confusion matrix:
[[749  79]
 [168 131]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.82      0.90      0.86       828
          1       0.62      0.44      0.51       299

avg / total       0.77      0.78      0.77      1127


                Accuracy        Specificity
        -1       0.78           0.44
         1       0.72           0.82
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  rbf , K-Digit PCA : 18
Support Vectors :  [1105 1031]  : Total number of support vectors :  2136
Confusion matrix:
[[742  92]
 [156 137]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support
```

```
          -1       0.83      0.89      0.86       834
           1       0.60      0.47      0.52       293

avg / total       0.77      0.78      0.77      1127


                 Accuracy        Specificity
         -1       0.78           0.47
          1       0.74           0.83
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Kernel :  rbf , K-Digit PCA : 18
Support Vectors :  [1104 1036]  : Total number of support vectors :  2140
Confusion matrix:
[[745  72]
 [172 137]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

          -1       0.81      0.91      0.86       817
           1       0.66      0.44      0.53       309

avg / total       0.77      0.78      0.77      1126


                 Accuracy        Specificity
         -1       0.78           0.44
          1       0.72           0.81
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Mean Cross Validation Error : 0.22
------------------------------------------------------------------------------
------------------------------------------------------------------------------
------------------------------------------------------------------------------
------------------------------------------------------------------------------
------------------------------------------------------------------------------
```

Training and Testing with best SVM model after Cross Validation

Minimum Cross Validation Error: **21%**

Best SVM Chosen : **Kernel - Linear, K-Digit PCA - 10**

Accuracy on Test Set : 78.40%

In [12]:

```
bestClassifierChurn=classifiers[minClassifierChurn]
bestPcaDataChurn=pcaDataChurn[minPcaDataChurn]
pcaValidationSetChurn=pcaValidationChurn[minPcaDataChurn]
bestClassifierChurn.fit(bestPcaDataChurn, churnTrainLables)
predicted=bestClassifierChurn.predict(bestPcaDataChurn)
trainAccuracy=getAccuracy(predicted,churnTrainLables.values)
predicted=bestClassifierChurn.predict(pcaValidationSetChurn)
testAccuracy=getAccuracy(predicted,churnTestLables.values)

churnCm=metrics.confusion_matrix(churnTestLables, predicted)
measuresChurn=measuresOfConMatrix(churnCm)
churnAcc=measuresChurn.accuracy()
churnSpe=measuresChurn.specificity()
print("Kernel : ",bestClassifierChurn.kernel,", K-Digit PCA :",bestPcaDataChurn.shape[1])
print("Support Vectors : ",bestClassifierChurn.n_support_," : Total number of support vectors : ",np.sum(bes
tClassifierChurn.n_support_))
print("No. of Margin SVs : ",marginCount,", No. of Non Margin SVs : ",nonMarginCount)
print("Train Data set Accuracy : %0.2f" % (trainAccuracy*100),"%")
print("Validation Data set Accuracy : %0.2f" % (testAccuracy*100),"%")
print("Confusion matrix:\n%s" % churnCm)

print("Classification report for classifier %s:\n%s\n"
      % (bestClassifierChurn, metrics.classification_report(churnTestLables, predicted)))
print("\t\tAccuracy\tSpecificity")
print("\t-1\t","%.2f\t"%churnAcc[0],"\t%.2f"%churnSpe[0])
print("\t 1\t","%.2f\t"%churnAcc[1],"\t%.2f"%churnSpe[1])

print("++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++")
```

```
Kernel :  linear , K-Digit PCA : 10
Support Vectors :  [1377 1374]  : Total number of support vectors :  2751
No. of Margin SVs :  2319 , No. of Non Margin SVs :  0
Train Data set Accuracy : 79.14 %
Validation Data set Accuracy : 78.33 %
Confusion matrix:
[[901 120]
 [183 194]]
Classification report for classifier SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False):
             precision    recall  f1-score   support

         -1       0.83      0.88      0.86      1021
          1       0.62      0.51      0.56       377

avg / total       0.77      0.78      0.78      1398


                Accuracy        Specificity
        -1       0.78           0.52
         1       0.75           0.83
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

Generalization Error for Arcene Dataset

All SVMs for Telco Customer Churn Dataset satisfy the below inequality $$E[Out Sample Error] <=\frac{E[Number of SVs]}{N-1}$$

In [13]:

```
generalizationError(len(classifiers),len(pcaDataChurn),meanErrorChurn,meanSupportVectorsChurn,pcaDataChurn[0
].shape[0])
```

```
E[Out_Sample Error]     <=      E[[Number of SVs]]/(N-1)
-----------------------------------------------------
|       0.21            <=       0.39080418959701757    |
-----------------------------------------------------
|       0.21            <=       0.38469731936800994    |
-----------------------------------------------------
|       0.21            <=       0.38267353097816437    |
-----------------------------------------------------
|       0.22            <=       0.3834546422865258     |
-----------------------------------------------------
```

Observations:

1) The generalization error inequality $E⟦Out Sample Error⟧ <=\frac{E⟦Number of SVs⟧}{N-1}$ is satisfied by both the datasets

2) If the number of support vectors is the upper bound of the inequality becomes less and thus the error of out samples also gets reduced.

3) Normalization just before PCA gives better accuracy

4) In K-Didigit PCA the best results were achieved at K=100 in Arcene Cancer Data set and K=18 in Telco Custoer Churn Dataset

5) In both the model selection methods GridSearch and K-Fold Cross validation, the later has given better results of accuracy than the former. Grid Search was used to find out the best combination of C and gamma that can give better accuracy.

6) Using K-Fold Cross Validation the accyracies are

- Arcene Data Set : 86% on test set and 100% on train set
- Telco Customer Data Set : 78.33% on test set and 79.14%