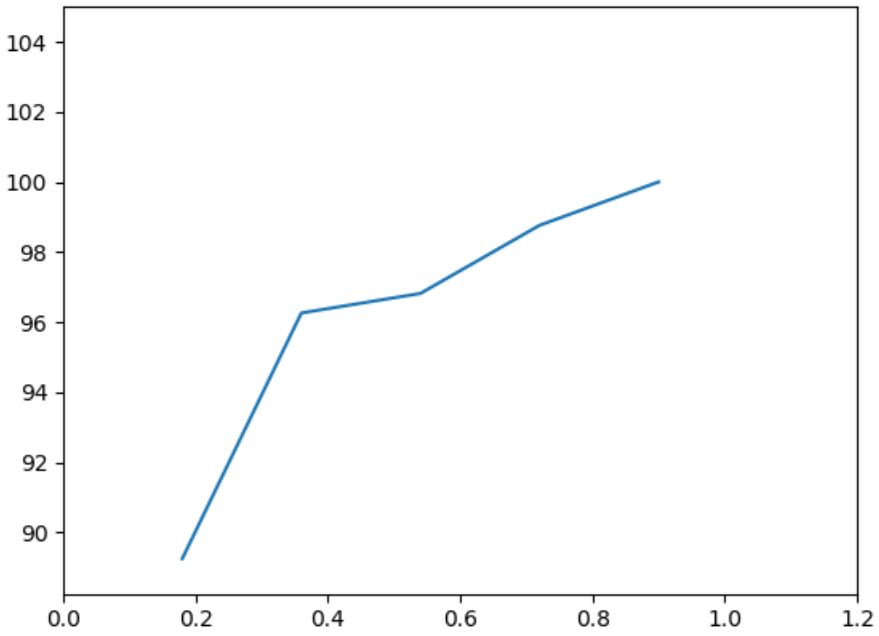
Question 1)

Knn is a simple classifier. It works on the basis of simply assigning ‘k’ nearest neighbors as members of the same class. To find which neighbor is nearest we usually use Euclidian distance.

Task I)

I selected the splits as [0.18, 0.36, 0.54, 0.72,0.90] (where 1=100% and 0=0%split)

Then I plotted the accuracy % vs split



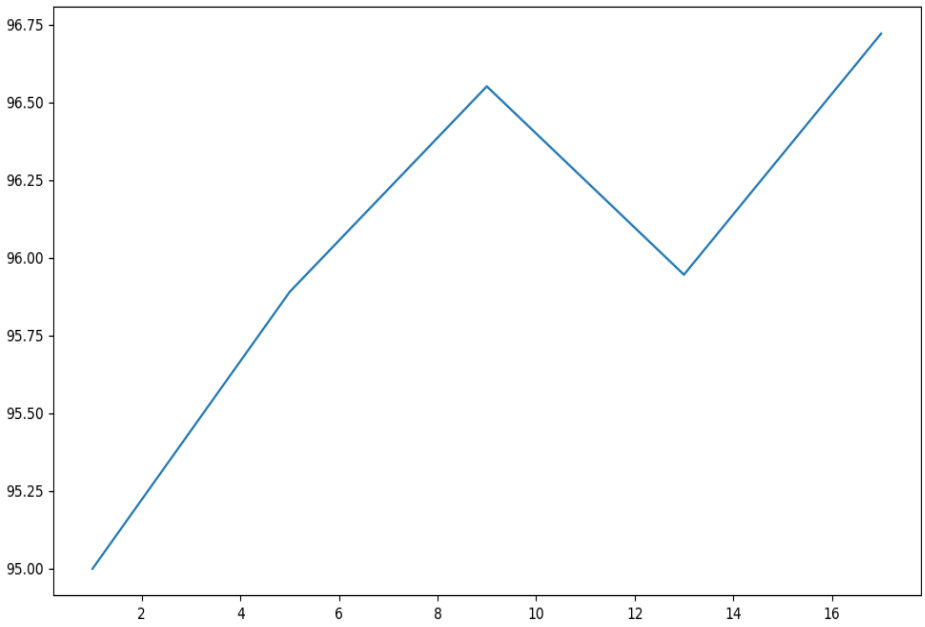
This shows that increasing split will increase the accuracy. We need to however be careful of overfitting the classifier with too many test sets.

Task II)

For this reason I have taken the split as 75%.

By plotting on the basis of k and split at 0.75 we get:

K’s selected-[1, 5, 9, 13,17]. Plotting accuracy vs K



Selecting the value of k greater than 17 gives best accuracy for the iris dataset. Overall running this test multiple times you can see that for greater k(around 20) we get the most accuracy.

Task III)
For cross validation I have used sklearn modules. For value of k as 19 and cross validation with 5 folds accuracy obtained for respective folds is
[array([0.96666667, 0.96666667, 0.93333333, 0.93333333, 1.])]
Mean: [0.96]

Code:

```
from sklearn.neighbors import KN
from sklearn.model_selection import cross_val_score
import numpy as np
import pandas as pd
df=pd.read_csv("iris.csv")
x=df.iloc[:,1:4]
y=df.iloc[:,4]
accuracy=[]
mean=[]
neighbours=[19]
for k in neighbours:
    knn=KN(n_neighbors=k)
    scores=cross_val_score(knn,x,y,cv=5,scoring='accuracy')
    accuracy.append(scores)
    mean.append(scores.mean())
print(accuracy)
```

Reference

Question 2

Naïve bayes is a classifier that uses naïve approach to classify into labels. It assumes that the features are not related to each other at all. It just checks the probability of a certain feature returning the respective label. It usually is one of the fast and accurate classifiers among the other options.

I tried many different iterations for devising gender features but without setting arbitrary class labels like extrovert or introvert, muscular or skinny, normal or rebel it is impossible to obtain more information other than unique characteristics of men and women. Even with arbitrary labels the answer is fuzzy and subjective. This makes the accuracy calculation tough. Even with subjective method however the gender classifier had more accuracy than any of my attempts. The dataset name itself has only names to analyze. The most accuracy I had was determining the gender of a name using simply the length of the name. Any names with less than 4 characters was said to be male by the classifier. This however wasn't accurate as compared to the last letter method.
Results of some tests
bob : male
billie : female
bill : female
jim : male
kate : female
kim : male
sarah : female
janet : female
tim : male
raj : male
dave:female
katrina:female
michael:female
simon:female
Arthur:female
accuracy: is less than 50% . This models accuracy depends on length of name.

The last letter example however has remarkable accuracy and it worked correctly for almost all names given to it, even the local names. I could find only one or two names like rafeh or aditya that gave incorrect classification. The model seems to have been built upon the last alphabet for women usually being a,y,e,i(vowels mostly) while men's names ending with consonants. Names that don't satisfy these conditions are labeled wrong.

However since the model classifies gender labels properly, you can use it to infer someone's gender with just their name. This can be useful (for companies that is!).Companies can devise strategies like showing products that are more preferred by women/men just by knowing your name. Names can easily be found out by social media like email id or Facebook .

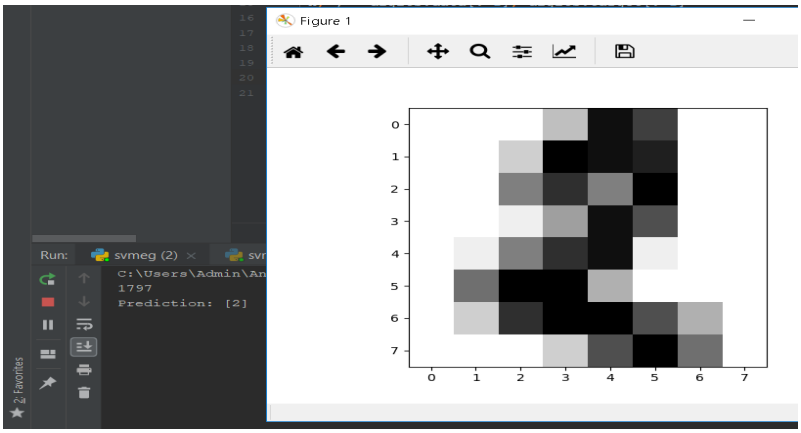
Question 3:

SVM:

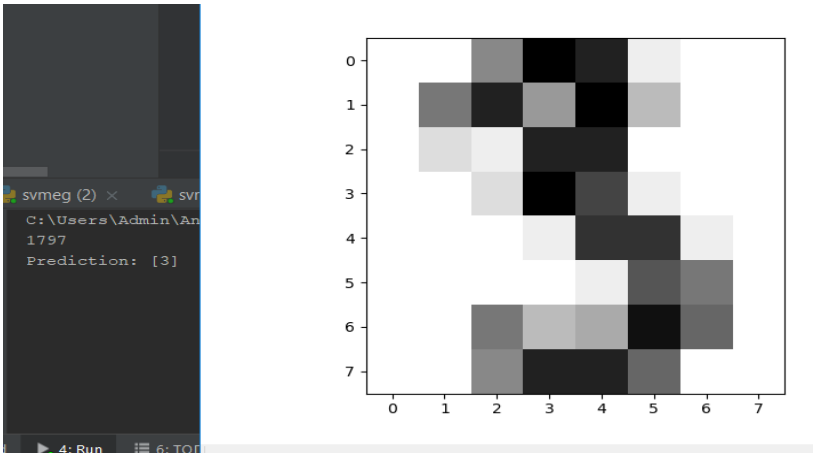
It constructs a hyperplane for classification of data into categories.

The objective of the hyperplane is to divide the data/dimensions such that the variance between the plane and datapoints is maximum. This results in best classification accuracy and also allows for further correct classification of future data.

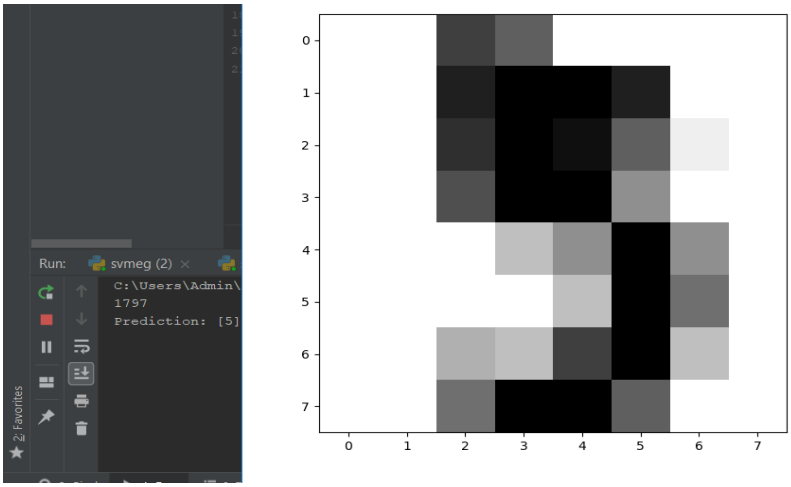
It has issues with high dimensional data because the data starts getting farther apart from each other and the variation of different planes also gets similar.



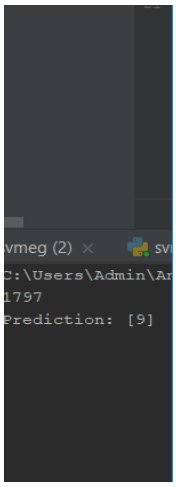
Some predictions by svm:
image:2 predicted:2

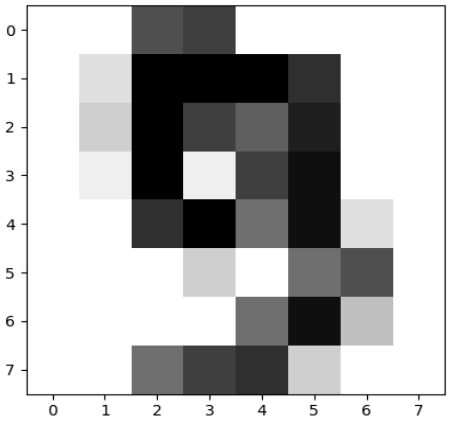


handwriting:3 prediction:3

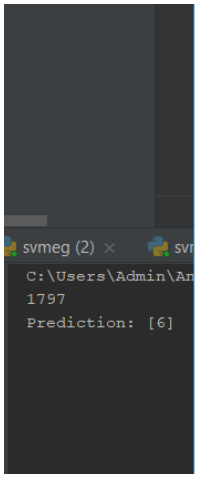


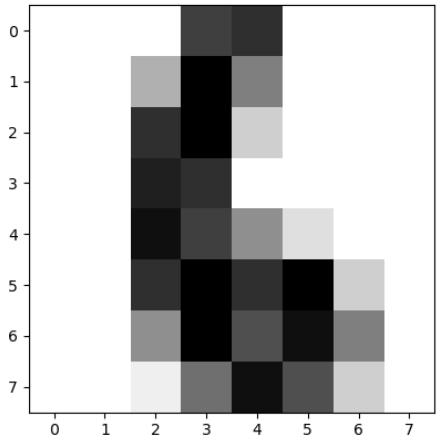
handwriting:5 prediction:5
(nb:looks close to 9)



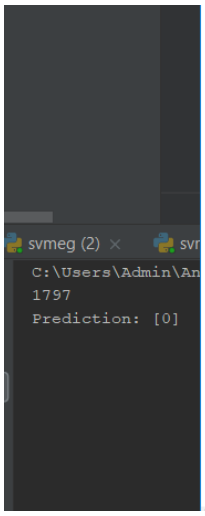


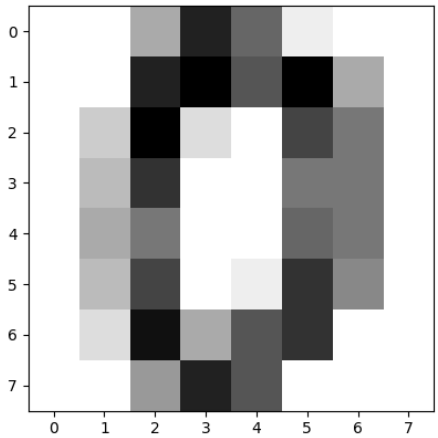
handwriting:9 prediction:9





handwriting:6 prediction:6





Handwriting:0 prediction:0

SVM produced accurate results for all tested digits.