INF552: Programming Assignment 6 [Support Vector Machines]

Priyambada Jain (priyambj@usc.edu)
Sai Sree Kamineni (skaminen@usc.edu)
Varad Kulkarni (vdkulkar@usc.edu)

Part 1 Implementation

Python Version: 2.7

Part(a)

Command to run – python SVM_lin.py
Input File – *linsep.txt*
Output - Weights = [7.2500563 -3.86188924]
Bias -0.13460531154
Equation of Curve => 7.25005629553*(x1)+ -3.86188924177*(x2) +-0.13460531154=0

Part(b)

Command to run – python SVM_nonlin.py
Input File – *nonlinsep.txt*
Output - Equation of Curve => 0.000138641398201(1+ 12.7478093059 *x1+0.199130319586 *x2)^3+0.00286066544765(1+ -10.2609690007 *x1+2.07391791405 *x2)^3+0.000207859219208(1+ 1.33933129592 *x1+-10.2909882202 *x2)^3+-0.00320750502579(1+ -9.4676088548 *x1+2.36139524894 *x2)^3+0.0=0

Note : All the input files should be placed in the same folder as the code.

Data Structures Used - Numpy array

Code Level Optimizations - Used cvxopt package for Quadratic Programming solver. Used Polynomial Kernel of Degree 3.

Challenges Faced – Identifying the best non-linear transformation for Kernel functions was an issue. Had to try different functions to identify better lagrangians.

Part 2: Software Familiarization

Python – 2.7
Package – sklearn

Part(a)

```
from sklearn import svm
import numpy as np
X = []
y = []
file =open('linsep.txt','r')
for line in file:
        X.append(map(float,line.split(",")[:2]))
        y.append(float(line.split(",")[2]))

X = np.array(X)
y = np.array(y)
model = svm.SVC(kernel='linear')
model.fit(X, y)
model.score(X, y)
```

Part(b)

```
from sklearn import svm
import numpy as np
X = []
y = []
file =open('nonlinsep.txt','r')
for line in file:
        X.append(map(float,line.split(",")[:2]))
        y.append(float(line.split(",")[2]))

X = np.array(X)
y = np.array(y)
model = svm.SVC(kernel='rbf')
model.fit(X, y)
model.score(X, y)
```

Explanation – SVC is the Support vector classification method. It takes various parameters like Kernel, degree, gamma etc.

Kernel is to specify the type of kernel function to be used. (linear, poly, rbf, sigmoid etc)

Degree is used only when kernel is “poly”. It specifies the degree of polynomial function to be used. Default is 3.

Gamma is used when kernel is Radial Basis Function. By default it is set to auto, but will take any value from user.

Fit method fits the train data.

Score method tests the data on the curve and gives the accuracy of separation.

Part 3: Applications

- **Face detection** – SVMs classify parts of the image as a face and non-face and create a square boundary around the face.
- **Text and hypertext categorization** – SVMs allow Text and hypertext categorization for both inductive and transductive models. They use training data to classify documents into different categories. It categorizes on the basis of the score generated and then compares with the threshold value.
- **Classification of images** – Use of SVMs provides better search accuracy for image classification. It provides better accuracy in comparison to the traditional query based searching techniques.
- **Bioinformatics** – It includes protein classification and cancer classification. We use SVM for identifying the classification of genes, patients on the basis of genes and other biological problems.
- **Protein fold and remote homology detection** – Apply SVM algorithms for protein remote homology detection.
- **Handwriting recognition** – We use SVMs to recognize hand written characters used widely.
- **Generalized predictive control(GPC)** – Use SVM based GPC to control chaotic dynamics with useful parameters.

Part 4: References

- https://data-flair.training/blogs/applications-of-svm/
- https://en.wikipedia.org/wiki/Support_vector_machine
- http://scikit-learn.org/stable/modules/svm.html
- https://docs.opencv.org/2.4/doc/tutorials/ml/introduction_to_svm/introduction_to_svm.html
- https://medium.com/.../chapter-2-svm-support-vector-machine-theory-f0812effc72
- http://www.statsoft.com/textbook/support-vector-machines