Amy Pitts
Final Exam
Instructor: Dr. Pablo Rivas

Data 440L: Machine Learning
Due: May 15, 2019

Exercise List

Questions 1,2

Exercise 1. ***Support Vector Machines:***

a) *Download the python program final.SVM.sinc.py which implements a 10-fold cross-validation approach to find the best set of hyper-parameters C, ϵ (epsilon), and γ (gamma), in an Support Vector Machine for Regression (SVR).*

b) *Download the python program finalGenData.py which generates data points of the "sinc" function contaminated with random noise.*

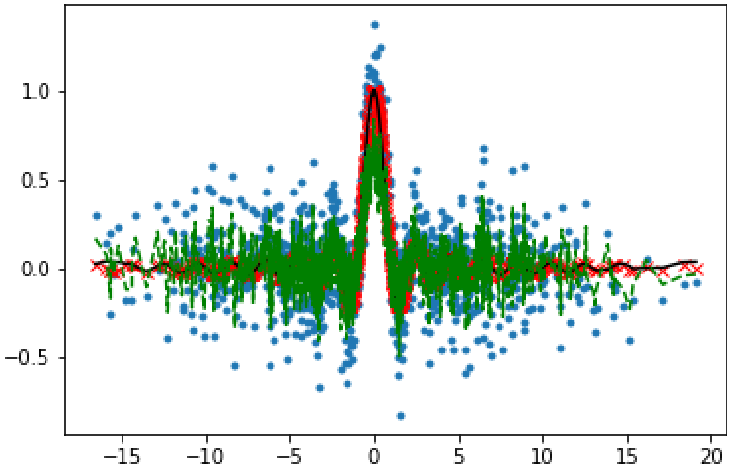
c) *Modify the program in 1.(a) to run for 1,000 samples, and then report the best set of hyperparameters found. Go here https://goo.gl/forms/X1maf8zSIjPhoyNA2 and report your results. You can do it as many times as you want, but at least one is required.*

d) ***Explain** your results. How do you interpret the hyper-parameters? Is there a great penalty? Is there room for errors without penalty? Is there a relationship between C and ϵ? Any other thoughts?*

e) ***(Extra credit +10)** Repeat 1.(c)-(d) but for 10,000 samples.*

Solution.

```
C 0.03125, epsilon 0.0, gamma 3.0517578125e-05. Testing set CV score: -0.385753
C 0.03125, epsilon 0.0, gamma 2.0. Testing set CV score: 0.163197
C 0.125, epsilon 0.0, gamma 2.0. Testing set CV score: 0.180956
C 0.5, epsilon 0.0, gamma 2.0. Testing set CV score: 0.242471
[LibSVM]Training set score: 0.999977
Testing set score: 0.999635
```



The best C is 0.5, the best epsilon is 0.0 with the best gamma being 2. The C is a relatively small value meaning there is not a high error term. The epsilon is very small, meaning that the range of values around y_i is not really a range but rather just the y_i values themselves. The gamma value is a whole number meaning the bell curve is spread out.

Exercise 2. ***More Support Vector Machines:*** *For this part you will use the digits dataset. For reading the dataset, you should use the following Python code in a file named finalGetDigits.py:*

```
import numpy as np
from numpy import genfromtxt

def getDataSet():
  # read digits data & split it into X and y for training and testing
  dataset = genfromtxt('features.csv', delimiter='␣')
  y = dataset[:, 0]
  X = dataset[:, 1:]

  dataset = genfromtxt('features-t.csv', delimiter='␣')
  y_te = dataset[:, 0]
  X_te = dataset[:, 1:]
  return X, y, X_te, y_te
```

Once saved do the following:

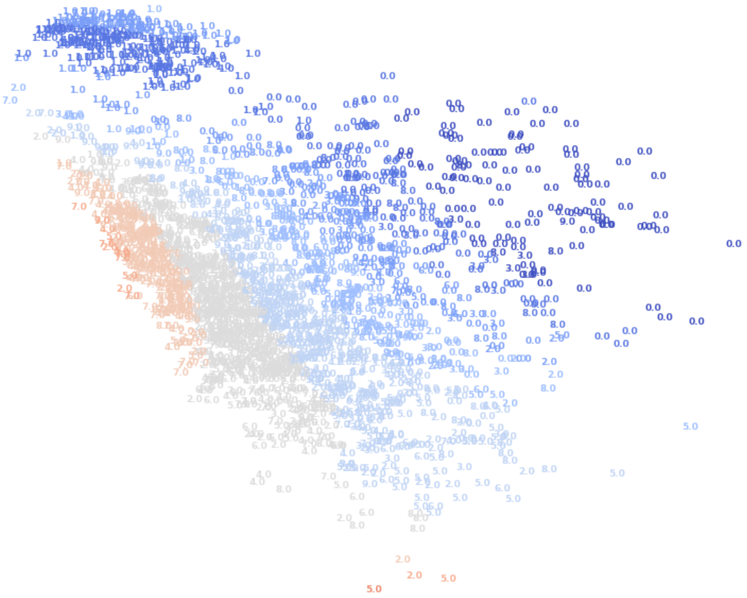
a) *Download the python program final.SVM.dig.py which implements a 10-fold cross-validation approach to find the best set of hyper-parameters C, ϵ (epsilon), and γ (gamma), in an Support Vector Machine for Regression (SVR) in the digits dataset.*

b) *Run the program in 2.(a) and modify it if you need to, to find and report the best set of hyperparameters and the final validation score. Save the plot and interpret it, give your comments about it.*

c) ***Explain*** *your results. How do you interpret the hyper-parameters? Is there a great penalty? Is there room for errors without penalty? Is there a relationship between C and ϵ? Any other thoughts?*

d) ***(Extra credit +30)*** *Prepare and share a single Colaboratory for both 1.(c)-(d) and 2.(b)-(c). You should use proper headings and comments and explanations. Pro Tip: you may have to combine all files into a single program or upload files on demand.*

Solution. **c)** BEST! C 4096.0, epsilon 1.4, gamma 0.0625. Testing set CV score: 0.337201



The C value is high which means that there is a high penalty on the error. The epsilon term describes how far away you can be from the y_i values. Since the y_i values represent the numbers that means that the epsilon range is $y_i + 1.4$ and $y_i - 1.4$ meaning the range could be + or - a number. The gamma of 0.0625 represent a small bell curve.

d) I did put all the work in a google-colab. I shared it with you. My user name is amypitts01.

Thank you!