TRINITY COLLEGE DUBLIN

School of Computer Science and Statistics

Week 2 Assignment CS7CS4/CSU44061 Machine Learning

Rules of the game:

- **Its ok to discuss with others, but we ask that you do not show any code you write to others. You must write answers in your own words and write code entirely yourself. All submissions will be checked for plagiarism.**
- **Reports must be typed (no handwritten answers please) and submitted as a pdf on Blackboard.**
- **When your answer includes a plot be sure to (i) label the axes, (ii) make sure all the text (including axes labels/ticks) is large enough to be clearly legible and (iii) explain in text what the plot shows.**
- **Include the cource of code written for the assignment as an appendix in your submitted pdf report and also include the code as a separate zip file (so we can easily execute it). Programs should be running code written in Python. Keep code brief and clean with meaningful variable names etc.**

Downloading Dataset

- Download the assignment dataset from `https://www.scss.tcd.ie/Doug.Leith/CSU44061/week2.php`. Important: You must fetch your own copy of the dataset, do not use the dataset downloaded by someone else.
- Please cut and paste the first line of the data file (which begins with a #) and include in your submission as it identifies your dataset.
- The data file consists of three columns of data (plus the first header line). The first two columns are input features and the third column is the ±1 valued target value.

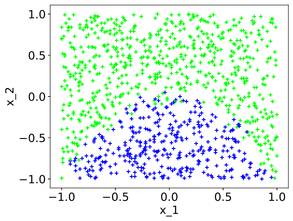
.

Assignment

Use sklearn for this assignment, there's no need to implement any of the models used here yourself. Include the code you write in an appendix to your submission. To read in the data file you can, for example, use pandas:

```
import numpy as np
import pandas as pd
df = pd.read_csv("week2.csv")
print(df.head())
X1=df.iloc[:,0]
X2=df.iloc[:,1]
X=np.column_stack((X1,X2))
y=df.iloc[:,2]
```

(a) (i) Visualise the data you downloaded by placing a marker on a 2D plot for each pair of feature values i.e. for each row in the data. On the plot the x-axis should be the value of the first feature, the y-axis the value of the second feature and the marker should be, for example, a + marker when the target value is +1 and a *o* when the target is −1. Your plot should look similar in style to this (with different data points of course!):



(ii) Use sklearn to train a logistic regression classifier on the data and report the parameter values of the trained model.

(iii) Now use the trained logistic regression classifier to predict the target values in the training data. Add these predictions to the 2D plot you generated in part (i), using a different colour so that the training data and the predictions can be distinguished. Show the decision boundary of the logistic regression classifier as a line on the plot (you'll need to use the parameter values of the trained model to figure out what line this should be - explain how you obtain it).

(iv) Briefly comment on how the predictions and the training data compare.

(b) Use sklearn to train linear SVM classifiers on your data (nb: be sure to use the LinearSVC function in sklearn, *not* the SVC function).

(i) Train linear SVM classifiers for a wide range of values of the penalty parameter C e.g. $C = 0.001$, $C = 1$, $C = 1000$. Report the parameter values of each trained model.

(ii) Use each of these trained classifiers to predict the target values in the training data. Plot these predictions and the actual target values from the data, together with the classifier decision boundary.

(iii) What is the impact on the model parameters of changing C, and why ? What is the impact on the SVM predictions?

(c) (i) Now create additional features by adding the square of each feature. Train a logistic regression classifier. Report the parameter values of each trained model.

(ii) Use the trained classifier to predict the target values in the training data. Plot these predictions and the actual target values from the data and compare.

(iii) For a bonus try to plot the classifier decision boundary (nb: it can be derived from the model parameters but its not a straight line anymore and you'll need to solve a quadratic equation). Don't worry if you can't do this though.