---------- MACHINE LEARNING HOMEWORK 2 ----------

PLEASE REFER TO THE IPYNB FILE for better comments
Comments are provided and explained with Markdowns

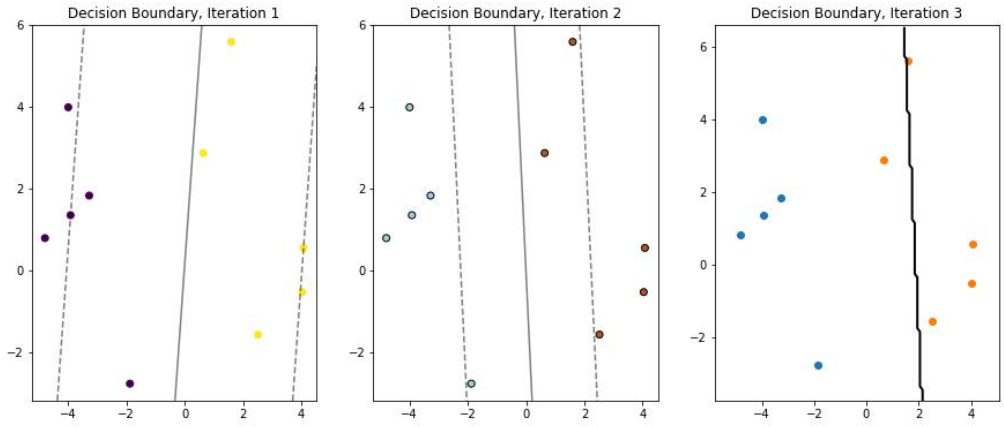
1 - Support Vector Machines

---- Part A ----

- Formulated the optimization function and the constraints for the corresponding linear maximum margin optimization problem without a regularization term.
- Formulated the Corresponding Lagrangian and the Lagrangian Dual.
- Find the details of this answer in ***ans_q3a.txt*** file.

---- Part B ----

- Performing Sequential Minimal Optimization (SMO) algorithm on the given dataset.
- In the plot, we can observe that the decision boundaries are converging with each iteration as the algorithm suggests.



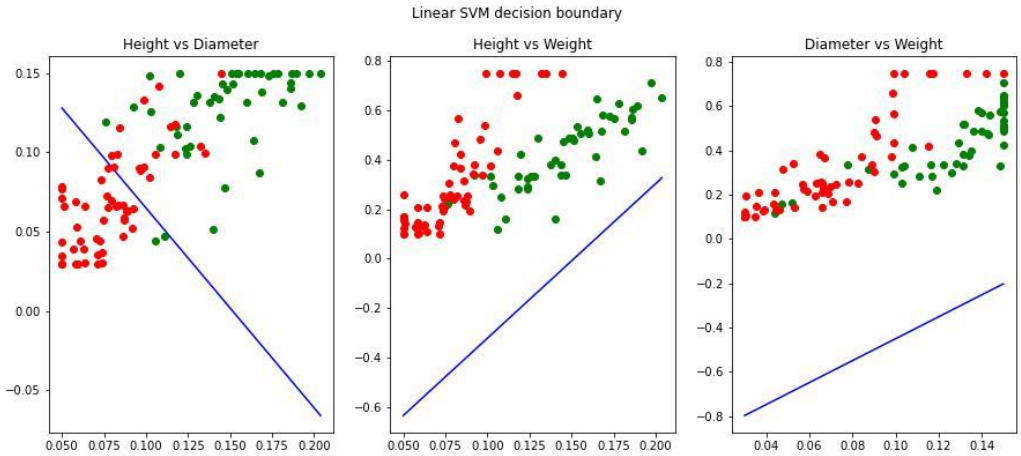
Reference - "Learning from Data" by Yaser S. Abu-Mostafa, Malik Magdon-Ismail, and Hsuan-Tien Lin.

2 - Support Vector Machines ONE-AGAINST-ALL

---- Part A ----

The steps for this involved -

- Loading the dataset and split it into train and test as required (first 6 rows of each material type for testing)
- Trained an SVM classifier using a linear kernel
- Tested the model on the test dataset and calculated the classification accuracy
- Plotted a graph for all combinations of features, showing the sum data distinction through 2D space as mentioned in the statement.
- The plot is included below.



OBSERVATIONS -

- While increasing the regularization weight value of non-zero C we see an increase in the model accuracy.
- I have experimented by varying the C value to 1, 5, 10, 50, and 100.
- After a little while, the accuracy stops increasing with the increasing value of C.
- In this case - the saturation point is at C = 10.
- For C = 50, 100; the model performance is not affected.
- The model appears to be not overfitting as the training and testing accuracies are consistent.

The best accuracy we're getting is -

For C = 50 -----

Training Accuracy: 97.06%

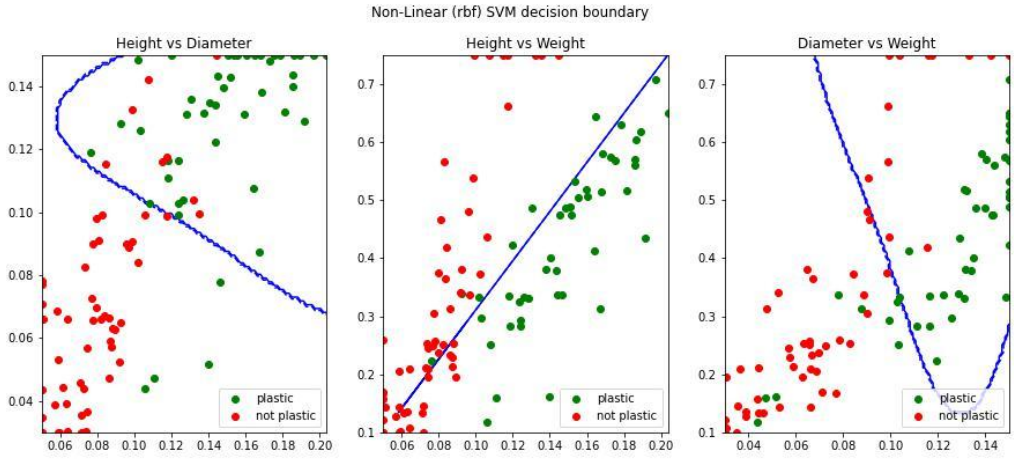
Testing Accuracy: 100.0%

---- Part B ----

The preprocessing steps are similar to part A. The only difference is while training the SVM classifier, we need to use a non-linear kernel. I am using the 'RBF' kernel.

OBSERVATIONS -

- As expected and observed earlier, while increasing the regularization weight value of C we see an increase in the model accuracy.
- I have experimented by varying the C value to 1, 5, 10, 20, 50, and 100.
- After a little while, the accuracy stops increasing with an increasing value of C.
- In this example - the saturation point is at C = 50.
- For C = 100; the model performance is not affected.
- The model does not appear to be overfitting as the training and testing accuracies are consistent.
- Plotted a graph for all combinations of features, showing the sum data distinction through 2D space as mentioned in the statement.



The best accuracy we're getting is -
For C = 50 -----
Training Accuracy: 96.08%
Testing Accuracy: 100.0%

3 - Decision Trees

---- Part A ----

- I used the ansq3a.txt file as the dataset for this. This dataset contains the first 2 rows of each material type given in the smaller dataset for homework 1.
- Using the three features - diameter, height, and weight, I am creating a decision tree for material type prediction.
- I am using the single-step lookahead and maximum information gain techniques as required in the question statement.

- For this, I am calculating the entropy first and calculating information gain for each feature to determine the maximum.
- After we get this, we can design a 2-level decision tree with thresholds for each feature to determine the material type.

---- Part B ----

- Build a decision tree learner for the classification problem.
- This decision tree depends on the max_depth parameter which is an arbitrary number to be provided externally.
- Maximum information gain condition is used to build the tree.

---- Part C ----

- Divided the dataset into training and testing sets and required
- Similar to question 2
- Running the decision tree learner to predict the material type label
- Iteratively calculating the training and testing accuracy by defining max_depth ranging from 1 to 9.
- Model performance can be measured by comparing accuracy for different max_depths

OBSERVATIONS -

- As expected, while increasing the max depth, the training, as well as testing accuracies, increase.
- After some time, at max depth level 6, the increase in training accuracy and the testing accuracy is relatively slower.
- We can also observe that for max depths 2,3,4,5 the testing accuracies are consistent.
- This can mean that the features at these depths are not learning meaningful information for testing data.
- Finally, the model seems to learn well at max_depth of 6 and 7 and does not appear to be overfitting as the training and testing accuracies are consistent with each other.