CS5691: Pattern Recognition and Machine Learning
Assignment #1

Topics: K-Nearest Neighbours, Naive Bayes, Regression **Deadline:** 28 Feb 2023, 11:55 PM

Teammate 1: (Yash Mathur) **Roll number:** CE19B106
I did this assignment completely on my own. All the codes are written in python. Libraries Used : numpy, pandas, matplotlib, math, random. Used `train_45`.csv and `test_45`.csv for all 3 datasets.

- Please refer to the **Additional Resources** tab on the Course webpage for basic programming instructions.
- This assignment has to be completed in teams of 2. Collaborations outside the team are strictly prohibited.
- Any kind of plagiarism will be dealt with severely. These include copying text or code from any online sources. These will lead to disciplinary actions according to institute guidelines. Acknowledge any and every resource used.
- Be precise with your explanations. Unnecessary verbosity will be penalized.
- Check the Moodle discussion forums regularly for updates regarding the assignment.
- You should submit a zip file titled **'rollnumber1_rollnumber2.zip'** on Moodle where rollnumber1 and rollnumber2 are your institute roll numbers. Your assignment will **NOT** be graded if it does not contain all of the following:
    1. Type your solutions in the provided LaTeX template file and title this file as **'Report.pdf'**. **State your respective contributions at the beginning of the report clearly.** Also, embed the result figures in your LaTeX solutions.
    2. Clearly name your source code for all the programs in **individual Google Colab files**. Please submit your code only as Google Colab file (.ipynb format). Also, embed the result figures in your Colab code files.
- We highly recommend using `Python 3.6+` and standard libraries like `NumPy, Matplotlib, Pandas, Seaborn`. Please use `Python 3.6+` as the only standard programming language to code your assignments. Please note: the TAs will only be able to assist you with doubts related to Python.
- You are expected to code all algorithms from scratch. **You cannot use standard inbuilt libraries for algorithms**. Using them will result in a straight zero on coding questions, `import` wisely!
- We have provided different training and testing sets for each team. f.e. train_1 and test_1 denotes training and testing set assigned to team id 1. Use sets assigned to your team only for all questions, reporting results using sets assigned to different team will result in straight zero marks.

- Any graph that you plot is unacceptable for grading unless it labels the x-axis and y-axis clearly.
- **Please start early and clear all doubts ASAP.**
- Please note that the TAs will **only** clarify doubts regarding problem statements. The TAs won't discuss any prospective solution or verify your solution or give hints.
- Please refer to the CS5691 PRML course handout for the late penalty instruction guidelines.
- Post your doubt only on Moodle so everyone is on the same page.

---

1. [**Regression**] You will implement linear regression as part of this question for the dataset1 provided here.

   Note that you can only regress over the points in the train dataset and you are not supposed to fit a curve on the test dataset. Whatever solution you get for the train data, you have to use that to make predictions on the test data and report results.

   (a) (2 marks) Use standard linear regression to get the best-fit curve. Split the data into train and validation sets and try to fit the model using a degree 1 polynomial then vary the degree term of the polynomial to arrive at an optimal solution.

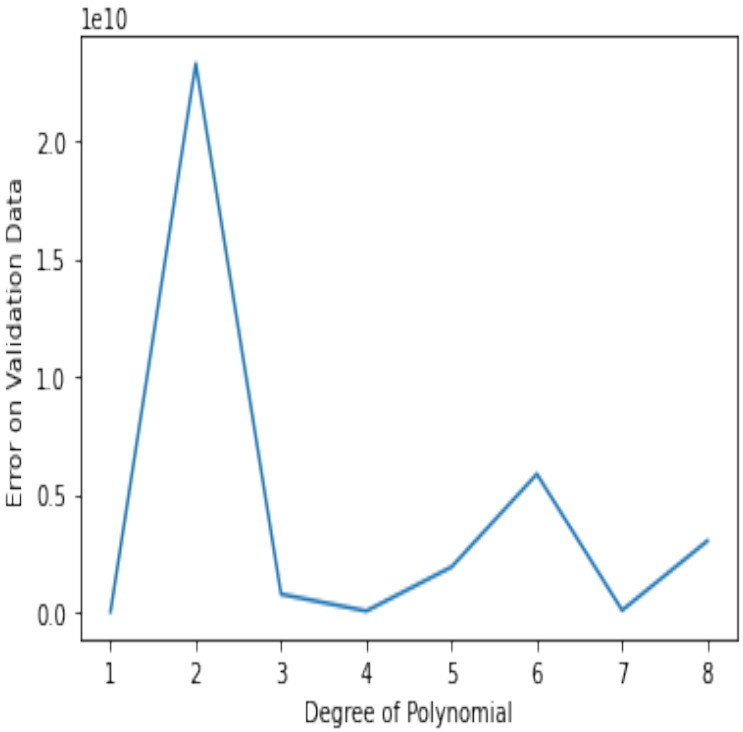
   For this, you are expected to report the following -

   - Plot different figures for train and validation data and for each figure plot curve of obtained function on data points for various degree term of the polynomial.(refer to fig. 1.4, Pattern Recognition and Machine Learning, by Christopher M. Bishop).
   - Plot the curve for Mean Square Error(MSE) Vs degree of the polynomial for train and validation data.(refer to fig. 1.5, Pattern Recognition and Machine Learning, by Christopher M. Bishop)
   - Report the error for the best model using Mean Square Error(MSE) for train and test data provided(Use closed-form solution).
   - Scatter plot of best model output vs expected output for both train and test data provided to you.
   - Report the observations from the obtained plots.

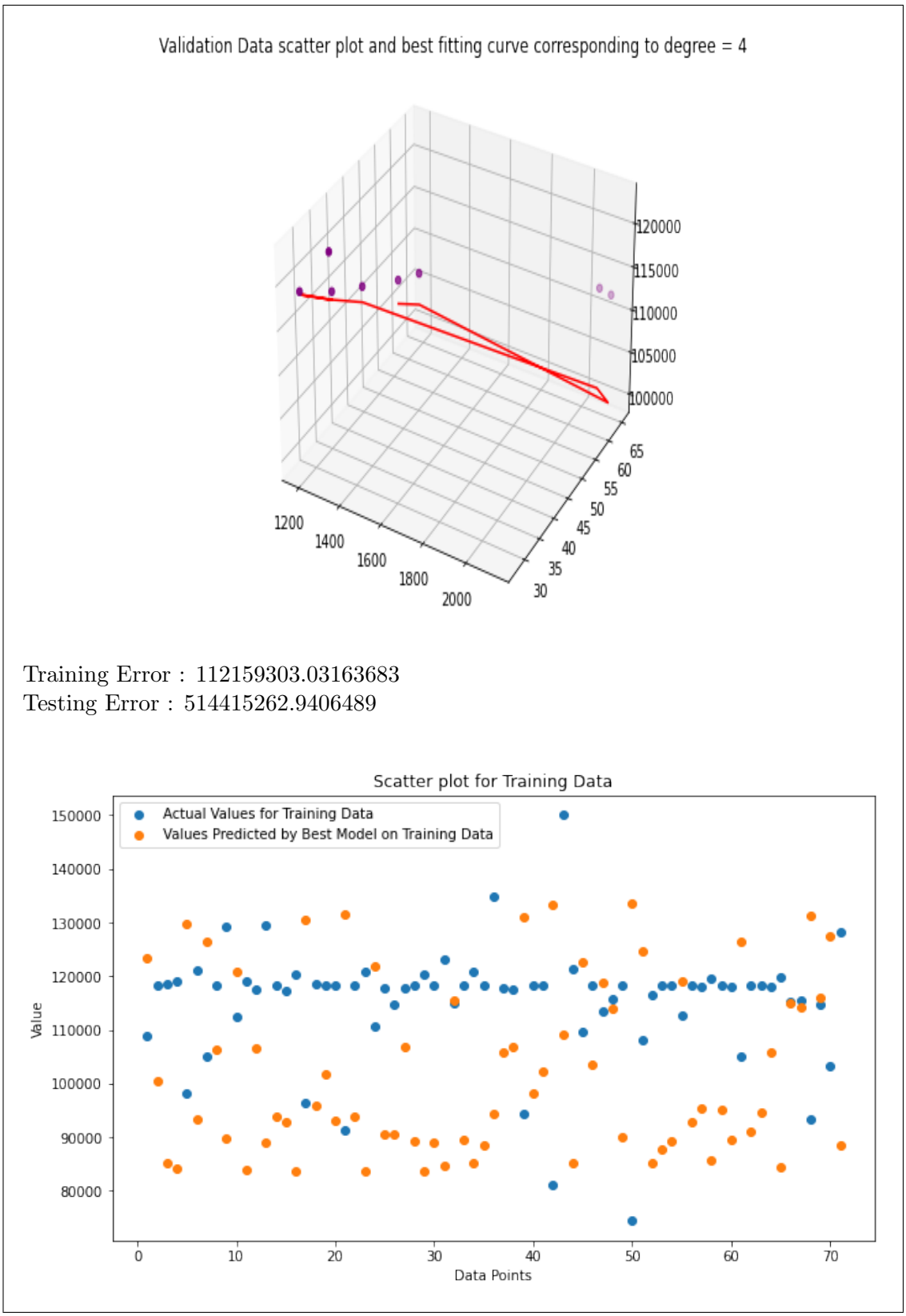
   Solution:
   Fitting Degree 1 Polynomial :

   Training Error : 16883013.909705855
   Testing Error : 1246983744.8886085

   Best Fitting Polynomial to Validation set : 4
   Graph for Error on Validation Data set vs Degree of Polynomial :



For degree 6 error suddenly increases due to overfitting.
Best Fitting Boundary for degree 4 polynomial on validation data :

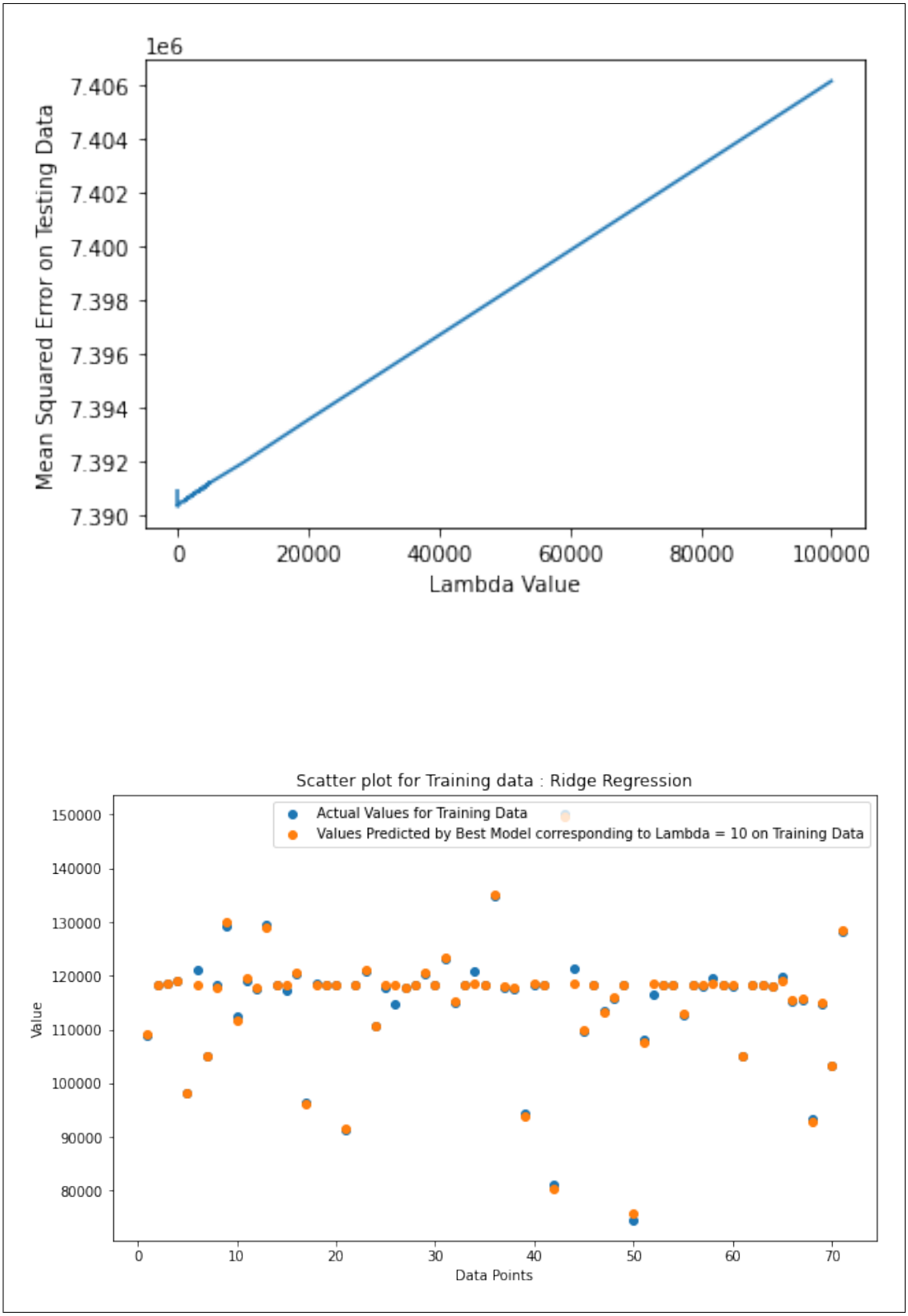


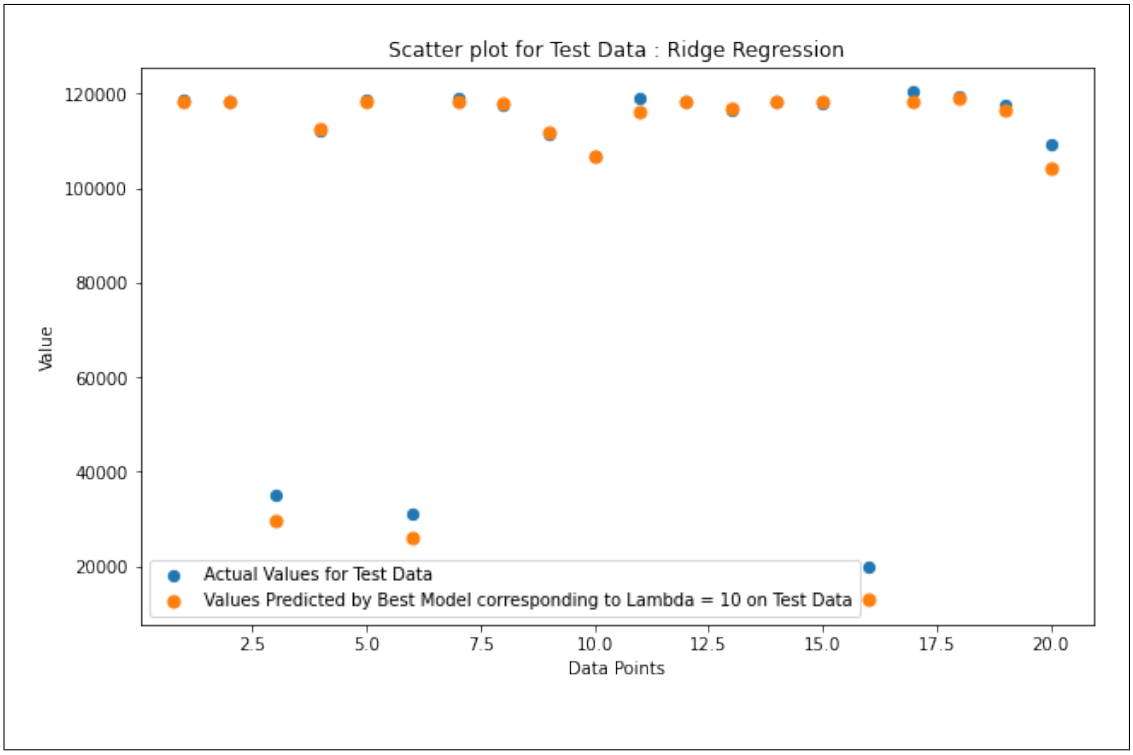
Training Error : 112159303.03163683
Testing Error : 514415262.9406489

(b) (3 marks) Split the data into train and validation sets and use ridge regression, then report for which value of lambda (λ) you obtain the best fit. For this, you are expected to report the following -

- Choose the degree from part (a), where the model overfits and try to control it using the regularization technique (Ridge regression).
- Use various choices of lambda(λ) and plot MSE test Vs lambda(λ).
- Report the error for the best model using Mean Square Error(MSE) for train and test data provided (Use closed-form solution).
- Scatter plot of best model output vs expected output for both train and test data provided to you.
- Report the observations from the obtained plots.

Solution:
Ridge Regression Best Lambda : 10
Training Error for lambda=10: 623435.9820944481
Testing Error for lambda=10 : 7390335.248822518
Extremely large lambda values causes increase in testing error since they lead to underfitting.


1e6
7.406
7.404
7.402
7.400
7.398
7.396
7.394
7.392
7.390
Mean Squared Error on Testing Data
0
20000
40000
60000
80000
100000
Lambda Value
Scatter plot for Training data : Ridge Regression
Actual Values for Training Data
Values Predicted by Best Model corresponding to Lambda = 10 on Training Data
150000
140000
130000
120000
110000
100000
90000
80000
Value
0
10
20
30
40
50
60
70
Data Points



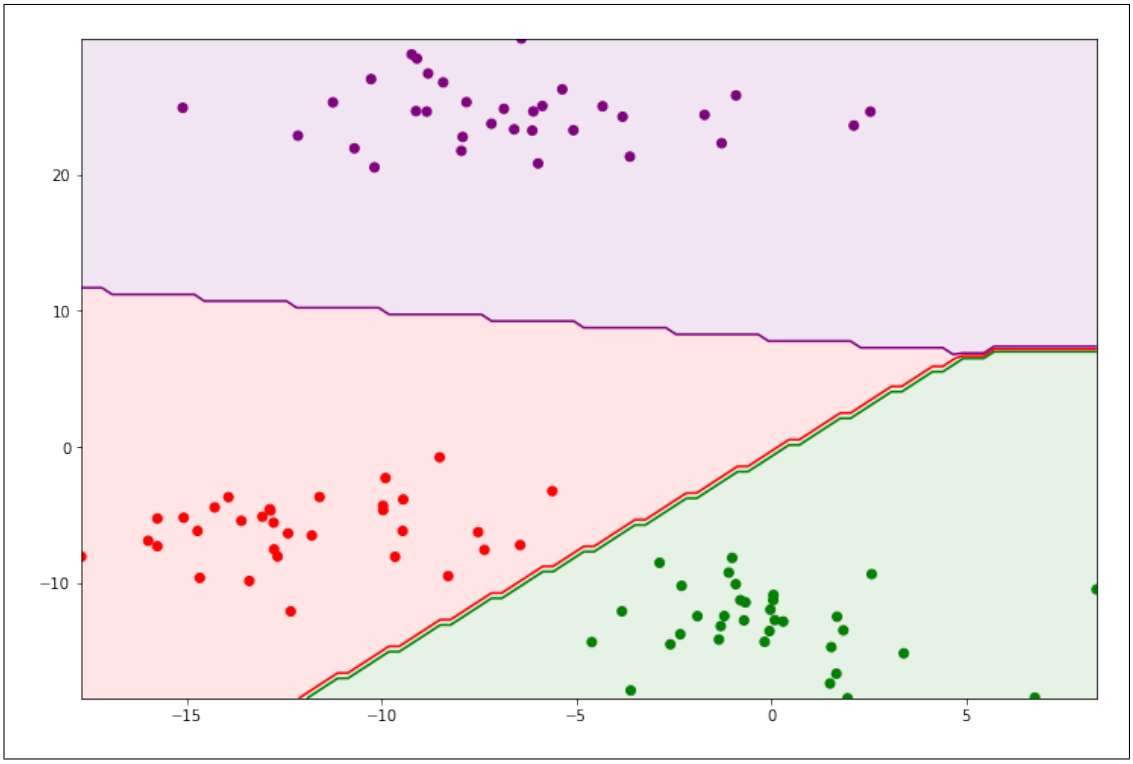
2. **[Naive Bayes Classifier]** In this Question, you are supposed to build Naive Bayes classifiers for the datasets assigned to your team. Train and test datasets for each team can be found here. For each sub-question below, the report should include the following:

   - Accuracy on both train and test data.
   - Plot of the test data along with your classification boundary.
   - confusion matrices on both train and test data.

   You can refer to sample plots here and can refer Section 2.6 of "Pattern classification" book by [Duda et al. 2001] for theory.

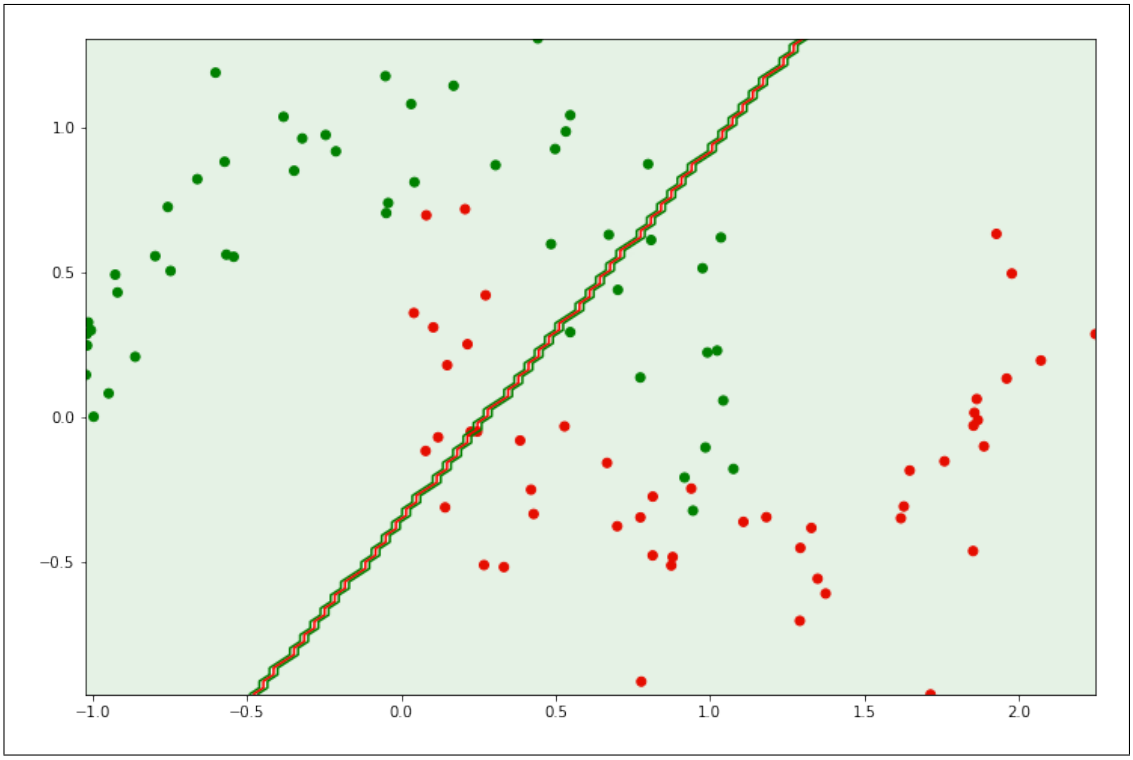
   (a) (1 mark) Implement Naive Bayes classifier with covariance = I on dataset2. where, I denotes the identity matrix.

   Solution:
   For Dataset 2 Training Accuracy = 100% and Test Accuracy = 100%.



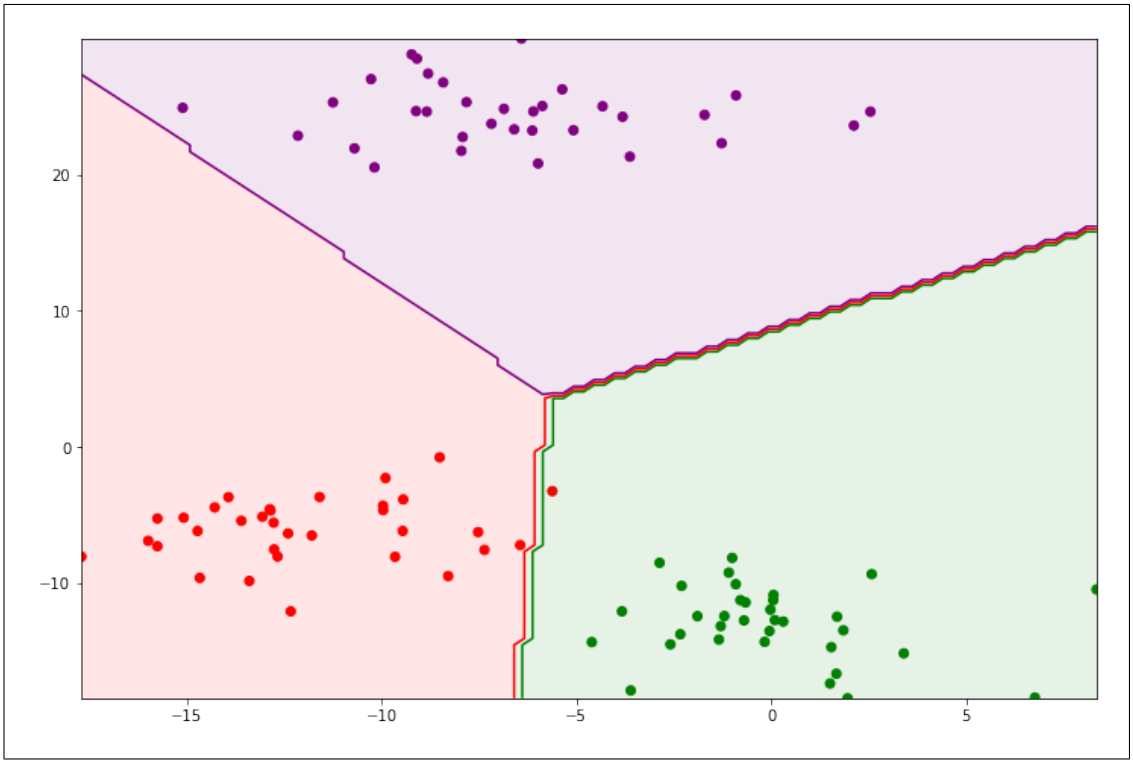
(b) (1 mark) Implement Naive Bayes classifier with covariance = I on dataset3. where, I denotes the identity matrix.

Solution:
For Dataset 3 Training Accuracy = 80% and Test Accuracy = 77%.



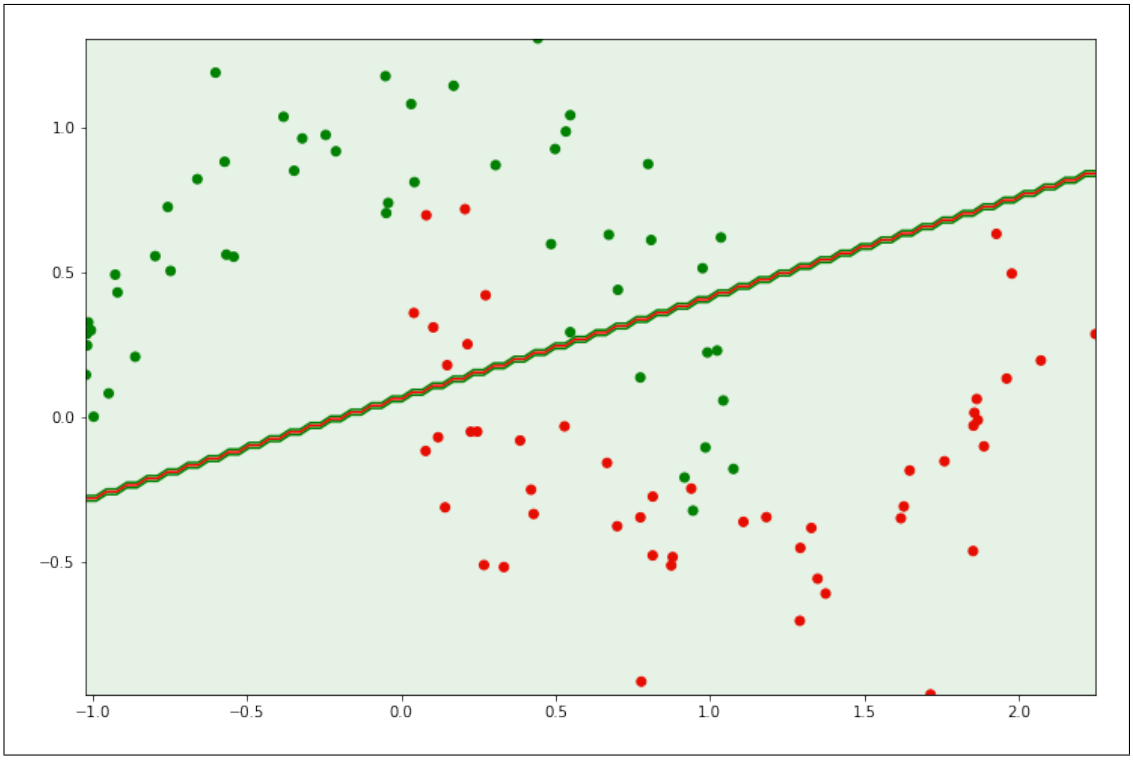
(c) (1 mark) Implement Naive Bayes classifier with covariance same for all classes on dataset2.

Solution:
For Dataset 2 Training Accuracy = 100% and Test Accuracy = 99%.



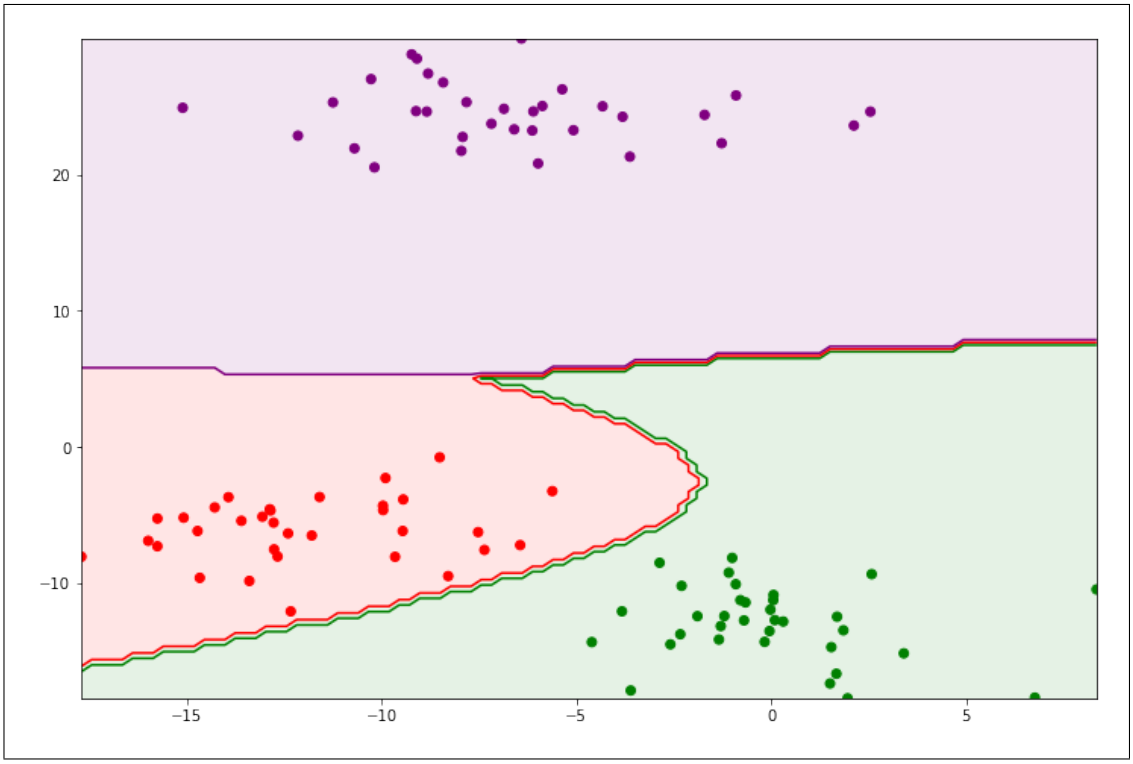
(d) (1 mark) Implement Naive Bayes classifier with covariance same for all classes on dataset3.

Solution:
For Dataset 3 Training Accuracy = 92% and Test Accuracy = 85%.



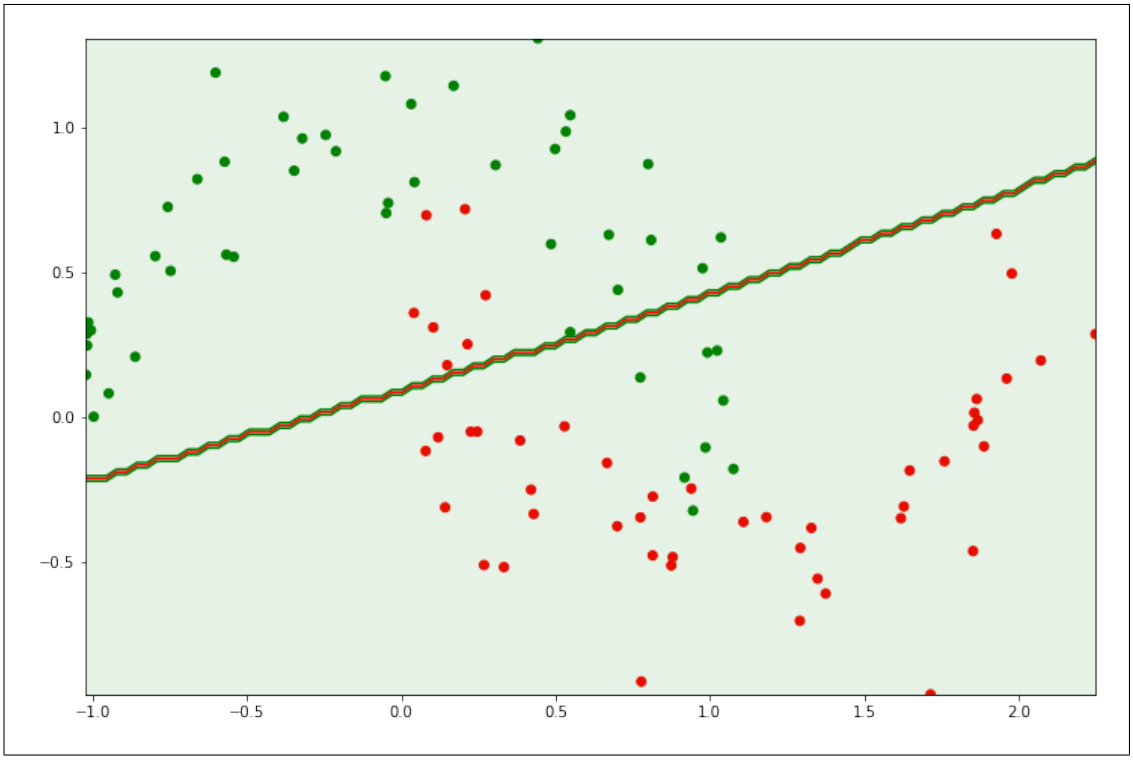
(e) (1 mark) Implement Naive Bayes classifier with covariance different for all classes on dataset2.

Solution:
For Dataset 2 Training Accuracy = 100% and Test Accuracy = 100%.



(f) (1 mark) Implement Naive Bayes classifier with covariance different for all classes on dataset3.

Solution:
For Dataset3 Training Accuracy = 84% and Test Accuracy = 85%.

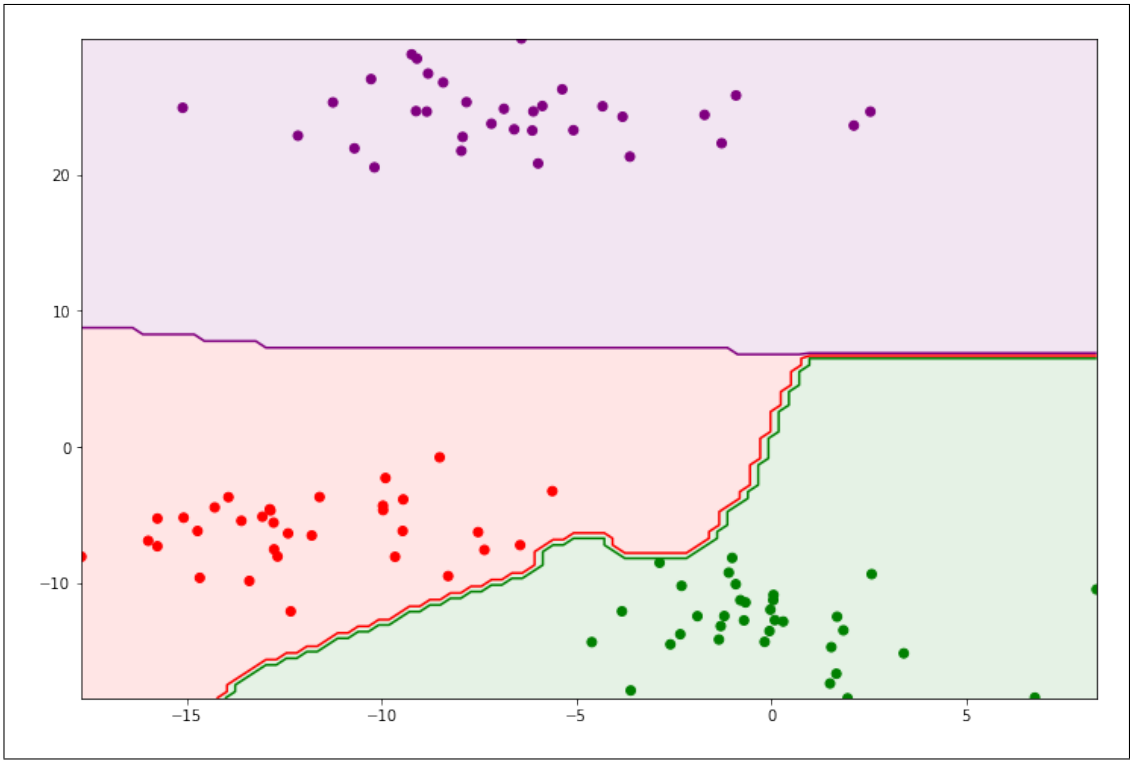


3. **[KNN Classifier]** In this Question, you are supposed to build the k-nearest neighbors classifiers on the datasets assigned to your team. Dataset for each team can be found here. For each sub-question below, the report should include the following:

   - Analysis of classifier with different values of k (number of neighbors). Split the data into train and validation sets and use validation set for finding optimal value of k.
   - Accuracy on both train and test data for the best model.
   - Plot of the test data along with your classification boundary for the best model.
   - confusion matrices on both train and test data for the best model.

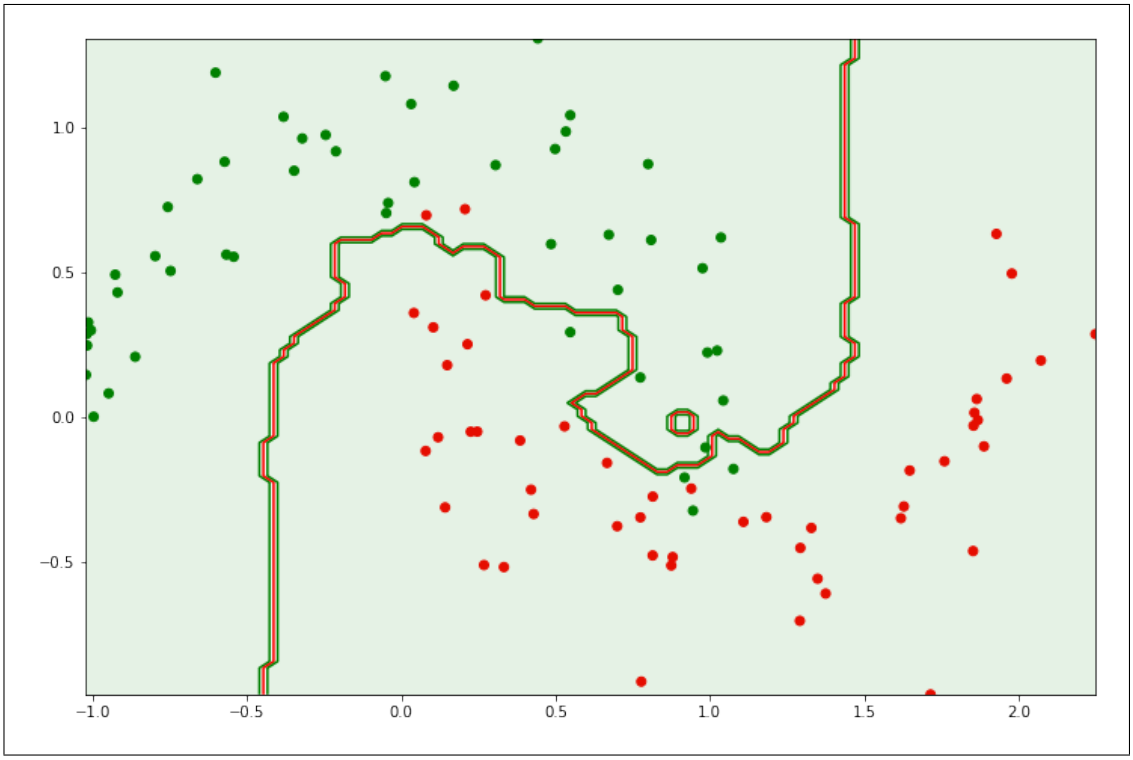
   (a) (2 marks) Implement k-nearest neighbors classifier on dataset2.

Solution:
For Dataset 2 I tried various values for k : [1,3,5,7,9,11,13,15,17]. On validation data set each one gave 100% accuracy. Hence I chose the smallest value of k ie 1.
On Testing with k value of 1 the accuracy came out to be 100%. The Classification boundary came out to similar to naive bayes decision boundary with covariance matrix as identity matrix.



(b) (2 marks) Implement k-nearest neighbors classifier on dataset3.

Solution:
For Dataset 3 I tried same k values as above. Got Accuracy of 96% for each of the k values. So I chose minimum k value ie 1.
On Test data accuracy came out to be 94%. Classification boundary came out to be way more complicated then naive bayes.


1.0
0.5
0.0
−0.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0