CS5691: Pattern Recognition and Machine Learning
Assignment #1

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

Teammate 1: (M.KARTHIK) **Roll number:** CS20B048
Teammate 2: (K.PAVAN NARSIMHA) **Roll number:** CS20B038

- 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.

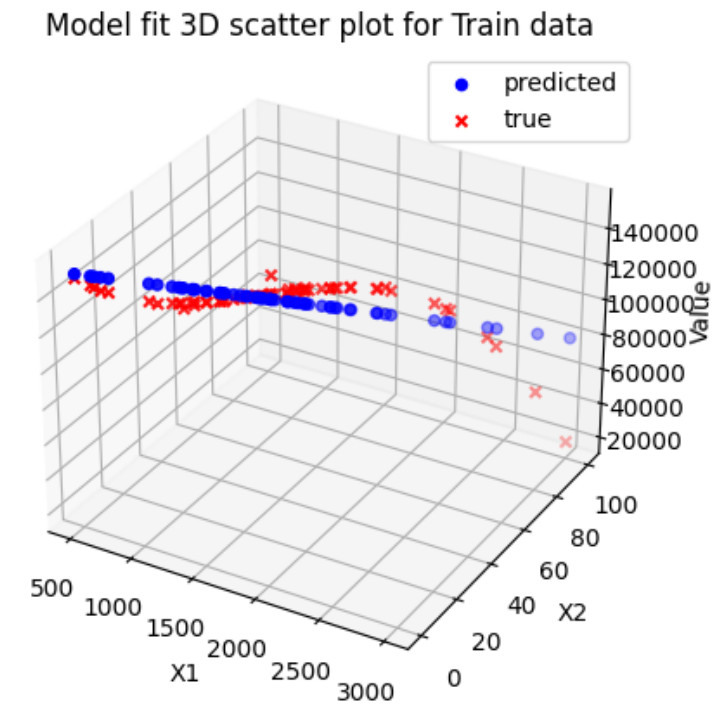
   For this, you are expected to report the following -

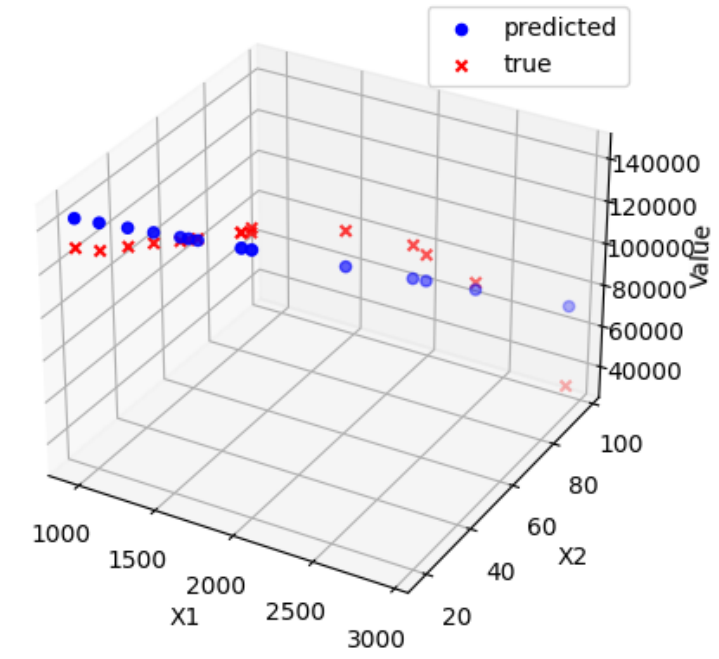
   - Plot different figures for train and validation data and for each figure plot curve of obtained function on data points for various degree term of the polynomial.(refer to fig. 1.4, Pattern Recognition and Machine Learning, by Christopher M. Bishop).
   - Plot the curve for Mean Square Error(MSE) Vs 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: Here i have splited the given data into train and validation data

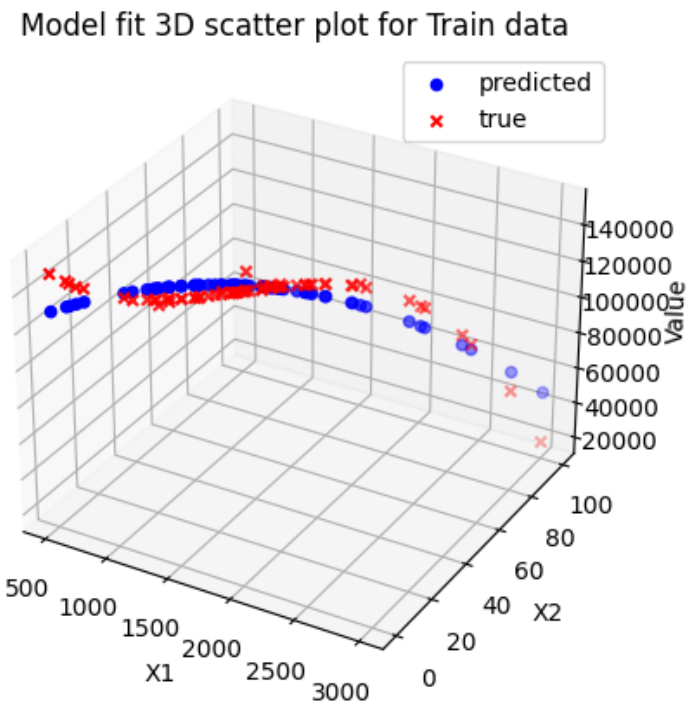
   - i have taken 80 percent of data into training and remaining into validation data

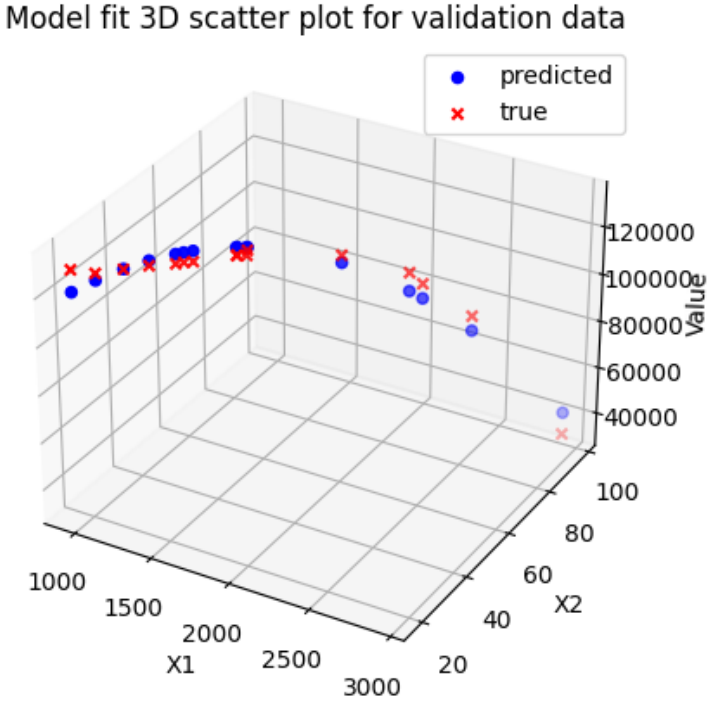
   1) plotting train and validation data for various degree scatter plot of Train and Validation data with degree 1



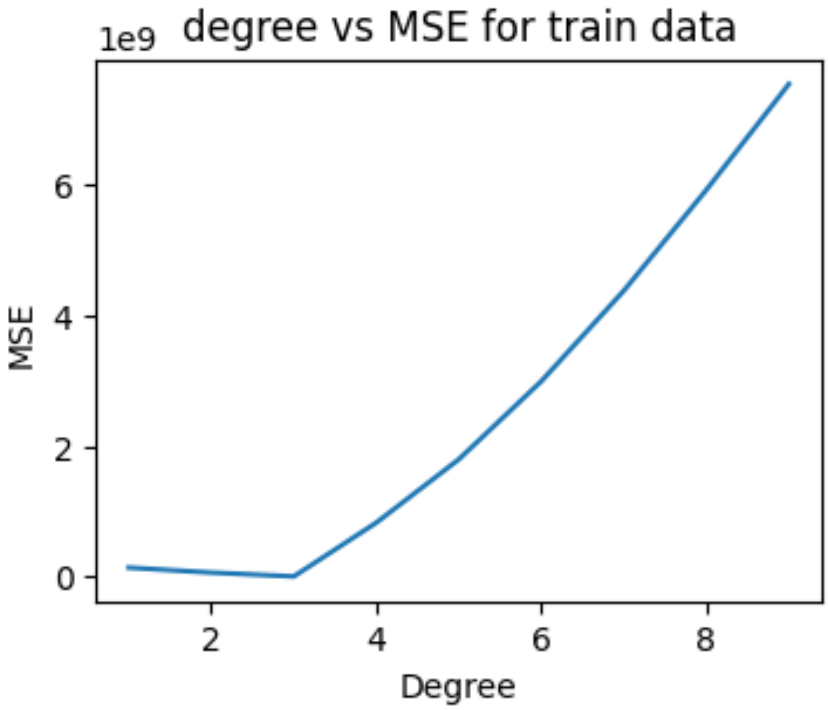


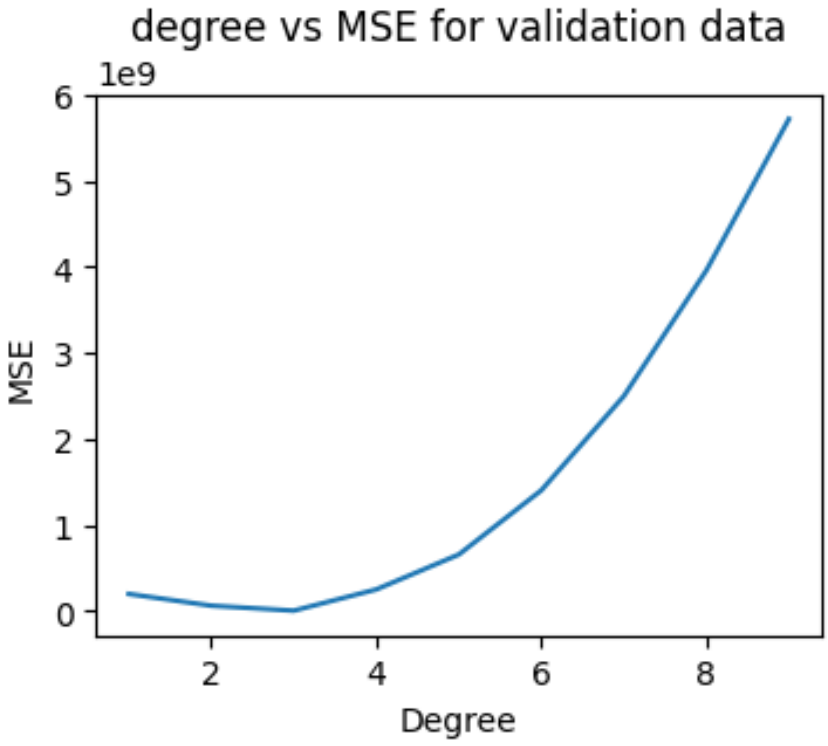
scatter plot of Train and Validation data with degree 2





2) MSE vs degree of ploynomial for train and validation data





from the above figures MSE is minimum at degree = 3 , so the best model is occuring

at degree = 3 for the given data

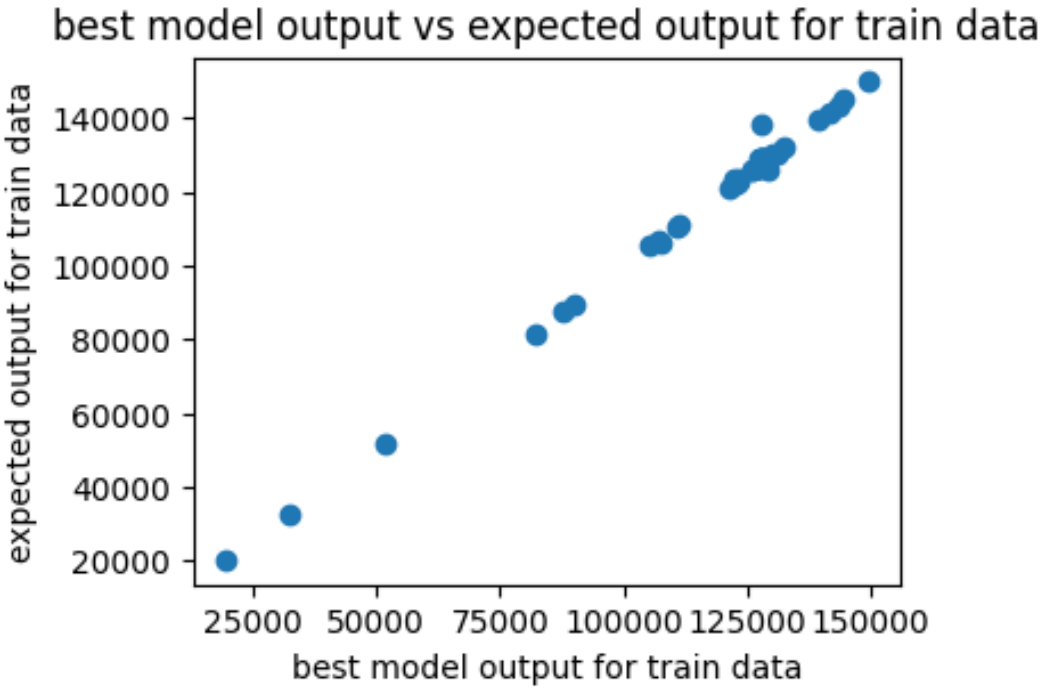
3)Error report

i.)Mean square error for train data = 1793055.2849815355

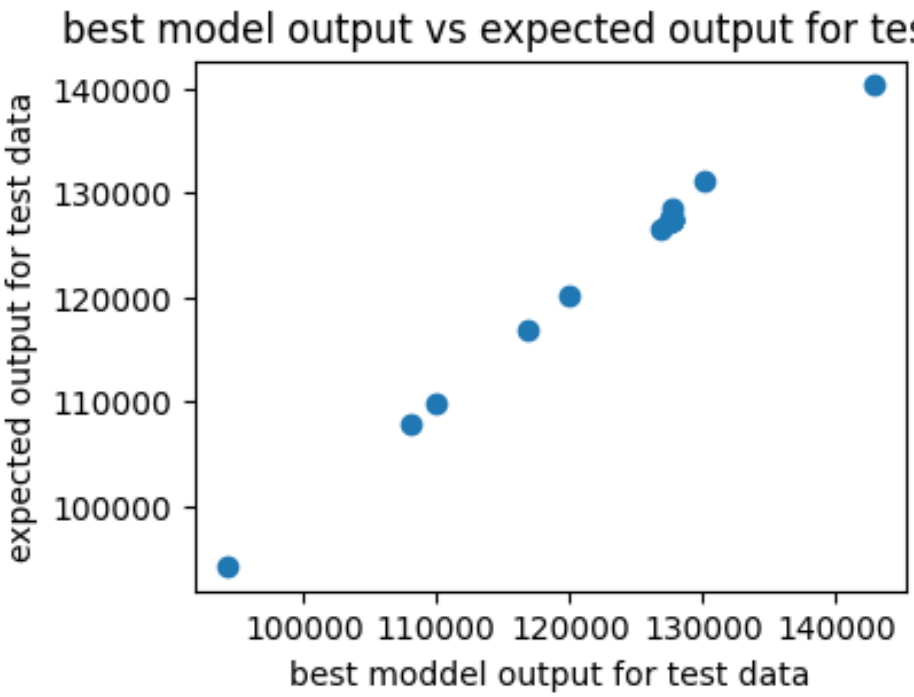
ii.)Mean square error for test data = 474281.7115487503

4)Scatter plot :

Scatter plot of Best Model Output vs Expected output for Train Data



Scatter plot of Best Model Output vs Expected output for Test Data



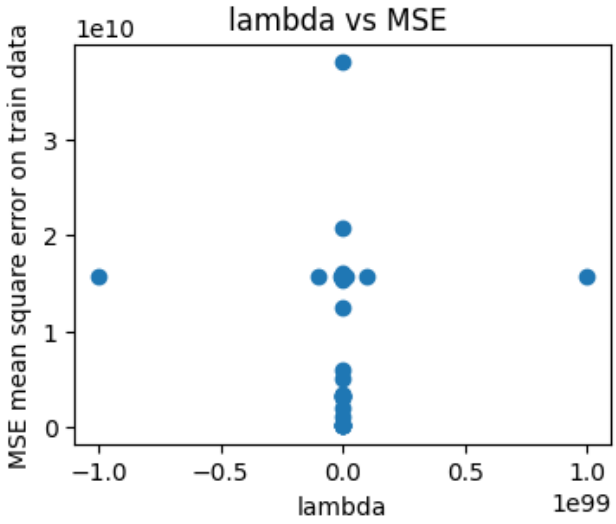
5)Observation:

- The best model occurs at degree = 3 for the given dataset
- Mean square error for train data is less than that of test data (since we trained on the test data)
- from the above two graphs(i.e for train and test data) we can observer that almost all the points are on the line y = x so our best model output and the expected output are almost equal which means that our prediction is correct

(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.

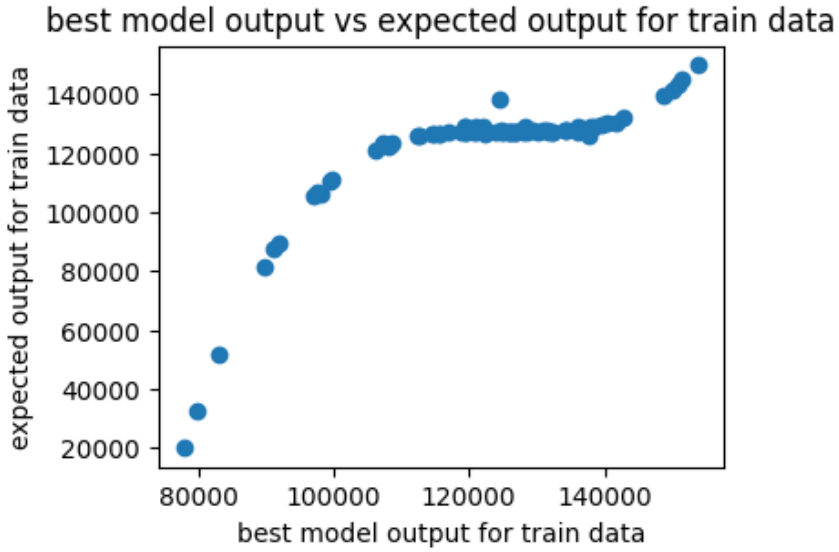
Solution: 1) controling using regularization technique
choose the degree from above such that mse should me minimum on train and maximum for validation data
2) Lambda vs MSE for test data



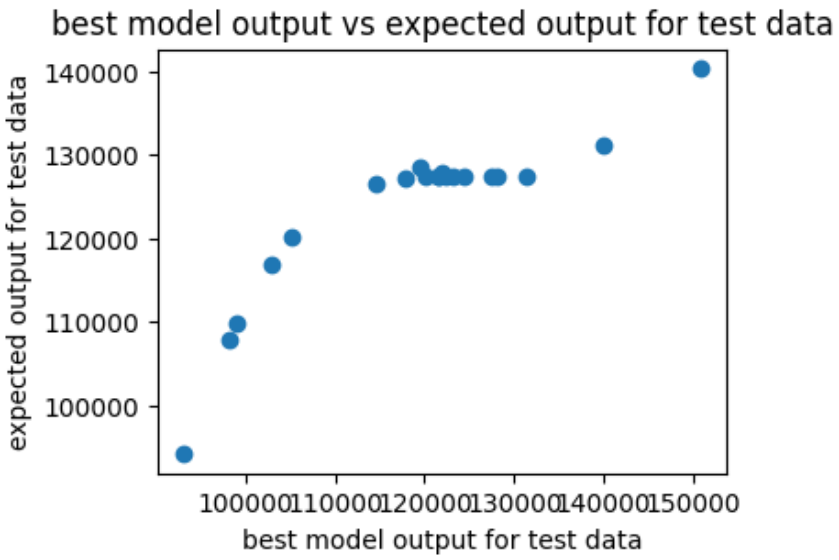
3)ERROR REPORT :

- MSE for train data = 149591348.23539045
- MSE for test data = 55806720.35376849

4) Scatter plot
best model output vs expected output for train data



best model output vs expected output for test data



5) Observation:

- yes , after adding lambda the mean square error is less than previous case(i.e without adding lambda) which means the overfit model gets controlled
- Before we got best model vs expected output as almost y =x where as here it deviated . but its not deviated as so much since we have controlled it by adding lambda
- So we can conclude that ridge regression can be controlled by adding lambda

Sir you can see the code **here**

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.

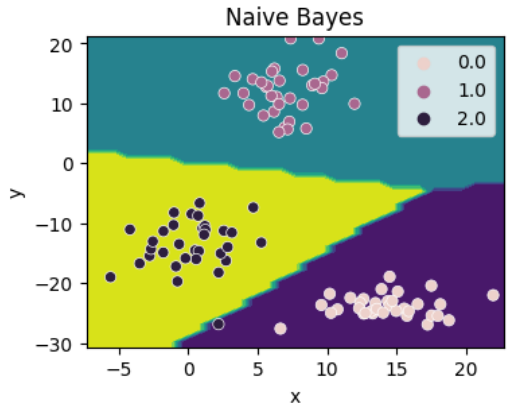
Solution: Sir you can see the code **here**
We know that the probability density in
Guassian distribution is as follows :

$$p(x|\mu,\Sigma) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right) \tag{1}$$

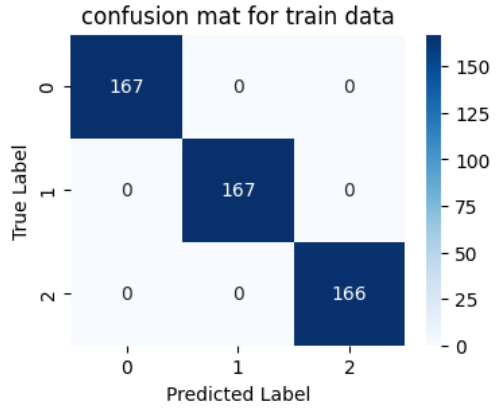
1) Accuracy on both train and test data

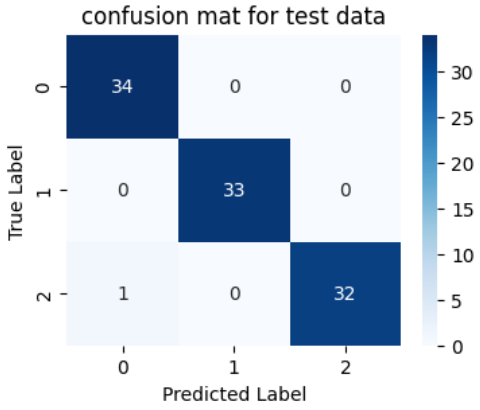
- Accuracy on train data: 100
- Accuracy on test data : 99

2) Plotting of test data along with classification boundary



3)confusion matrices on both train and test data



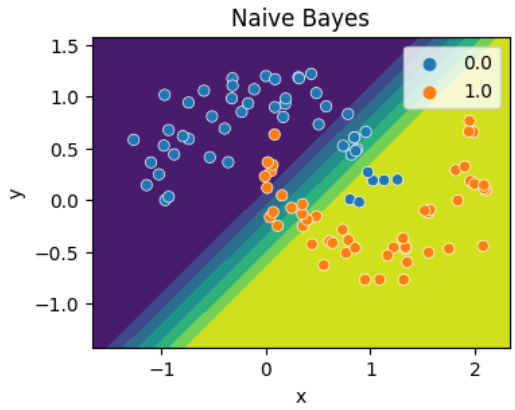


(b) (1 mark) Implement Naive Bayes classifier with covariance = I on dataset3. where, I denotes the identity matrix.

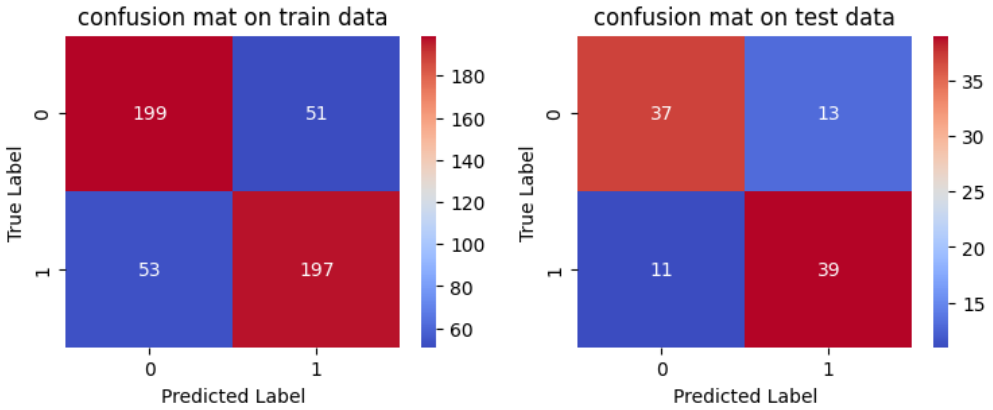
Solution: 1) Accuracy on both train and test data

- Accuracy on train data: 79.2
- Accuracy on test data : 76.3

2) Plotting of test data along with classification boundary



3)confusion matrices on both train and test data



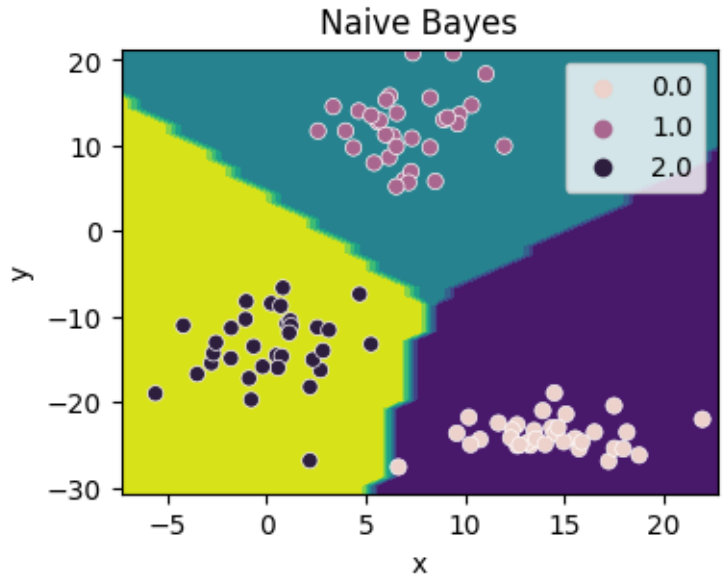
(c) (1 mark) Implement Naive Bayes classifier with covariance same for all classes on dataset2.

Solution:

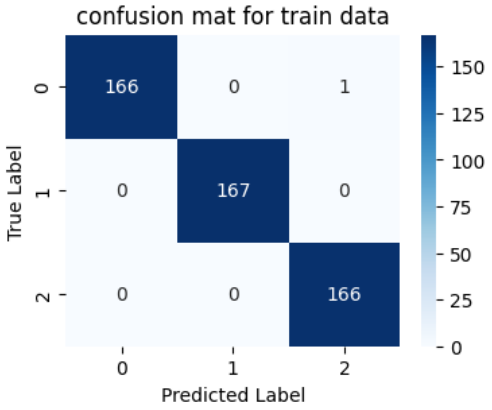
1) Accuracy on both train and test data

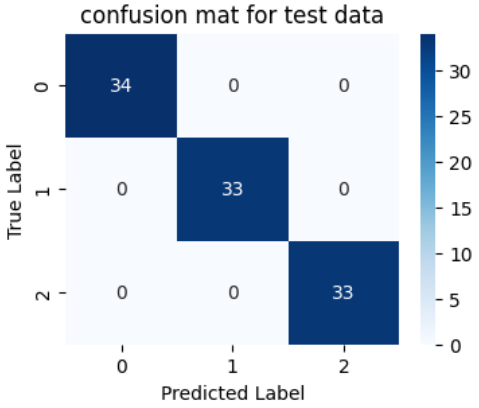
- Accuracy on train data: 99.8
- Accuracy on test data : 100

2) Plotting of test data along with classification boundary



3)confusion matrices on both train and test data





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

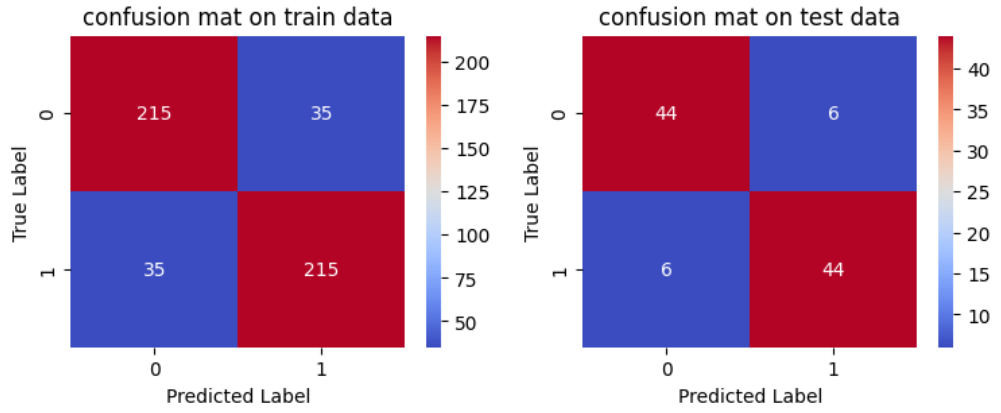
Solution:

1) Accuracy on both train and test data

- Accuracy on train data: 86.0
- Accuracy on test data : 88.0

2) Plotting of test data along with classification boundary

3)confusion matrices on both train and test data



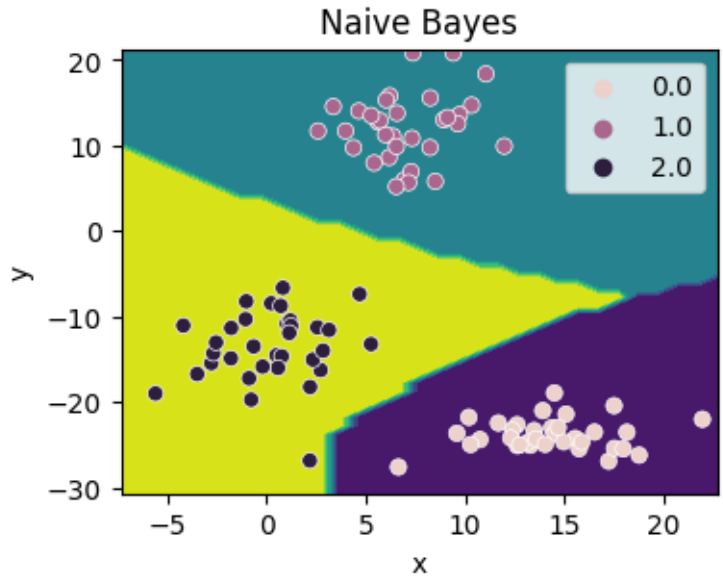
(e) (1 mark) Implement Naive Bayes classifier with covariance different for all classes on dataset2.

Solution:

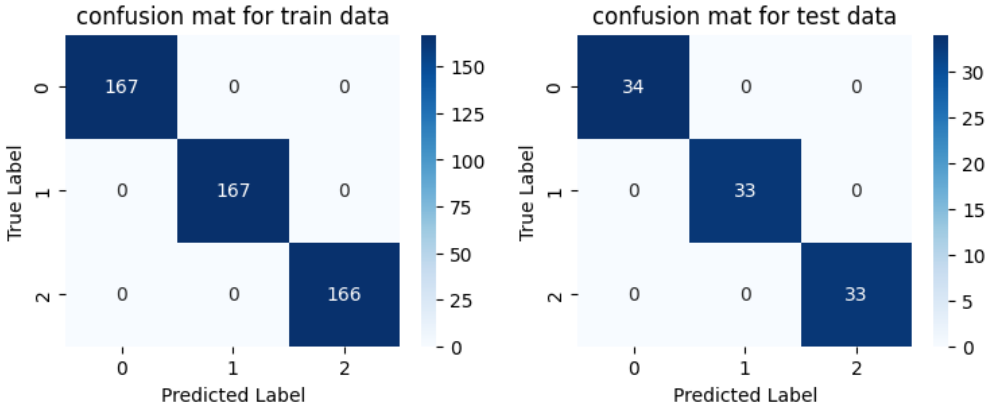
1) Accuracy on both train and test data

- Accuracy on train data: 100
- Accuracy on test data : 100

2) Plotting of test data along with classification boundary



3)confusion matrices on both train and test data



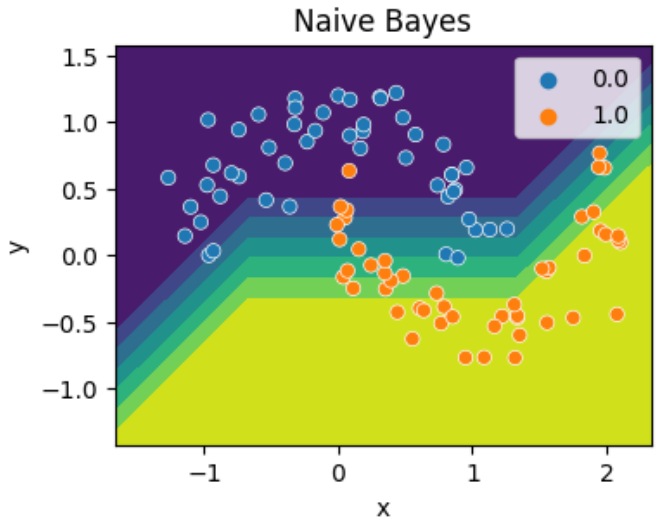
(f) (1 mark) Implement Naive Bayes classifier with covariance different for all classes on dataset3.

Solution:

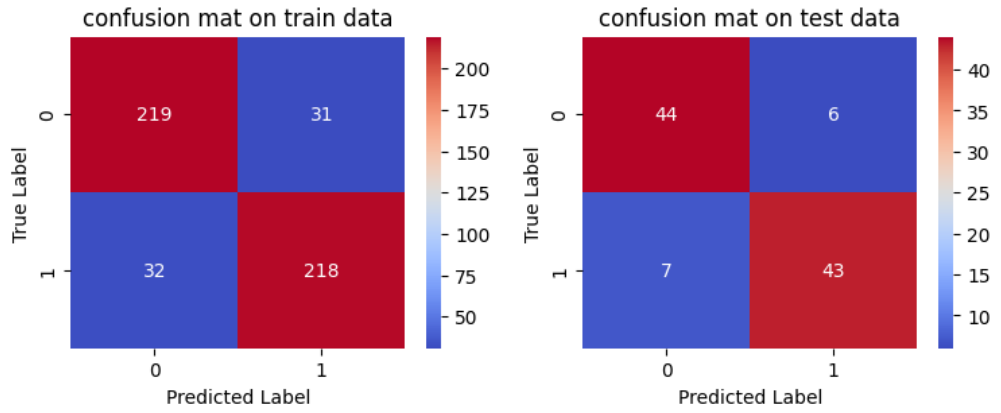
1) Accuracy on both train and test data

- Accuracy on train data: 87.4
- Accuracy on test data : 87.0

2) Plotting of test data along with classification boundary



3)confusion matrices on both train and test data



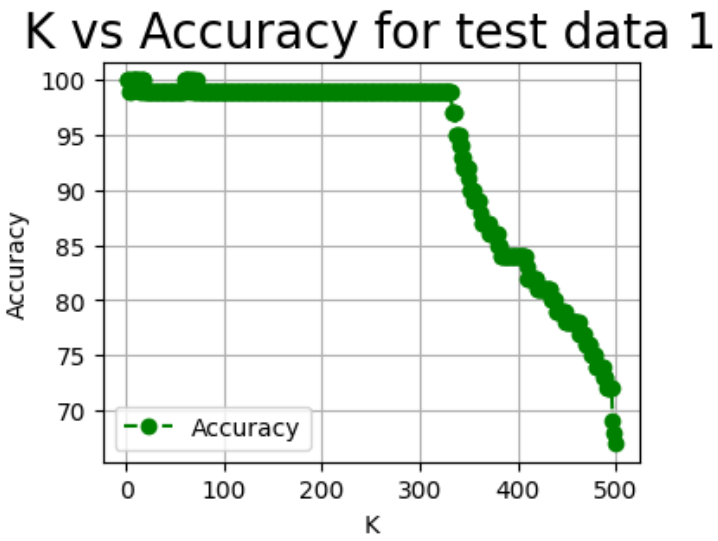
3. **[KNN Classifier]** In this Question, you are supposed to build the k-nearest neighbors classifiers on the datasets assigned to your team. Dataset for each team can be found here. For each sub-question below, the report should include the following:

   - Analysis of classifier with different values of k (number of neighbors).
   - 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:
   Sir you can see the code **here**

1) Analysis with different values of K on dataset-2
-¿ Below figure shows the graph K vs Accuracy

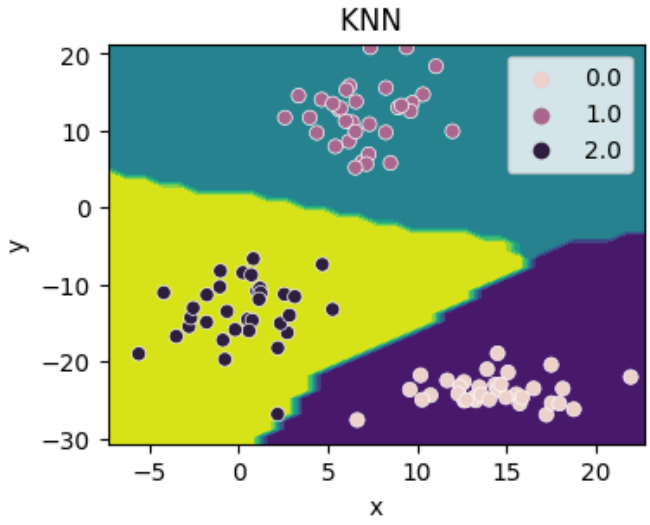


- so i have choosen k = 10 for the rest of the problem(BEST MODEL)

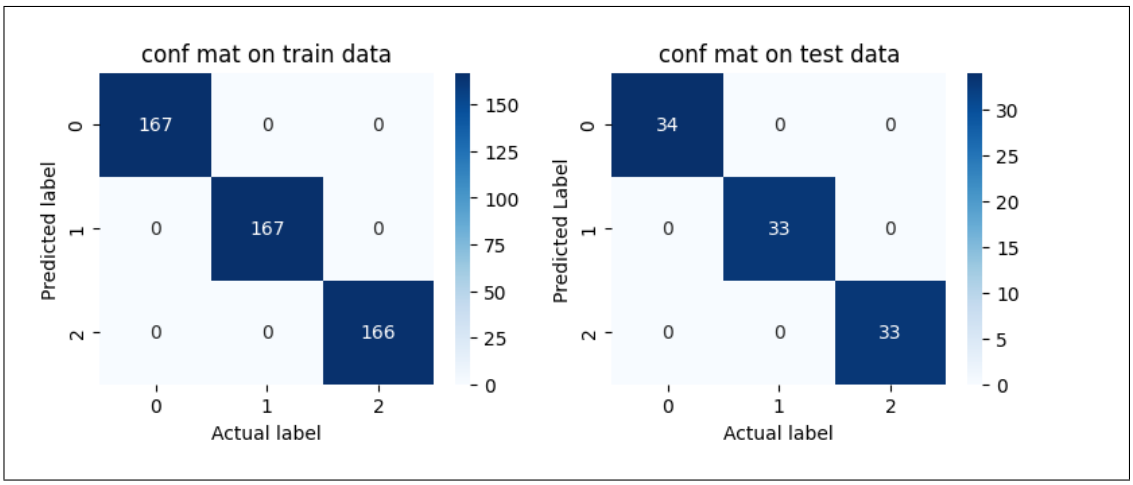
2) Accuracy on both train and test data for best model(K=10)

- Accuracy on train data = 100
- Accuracy on test data = 100

3) Plot of test data with my classification boundary for best model(k=10) on Dataset2



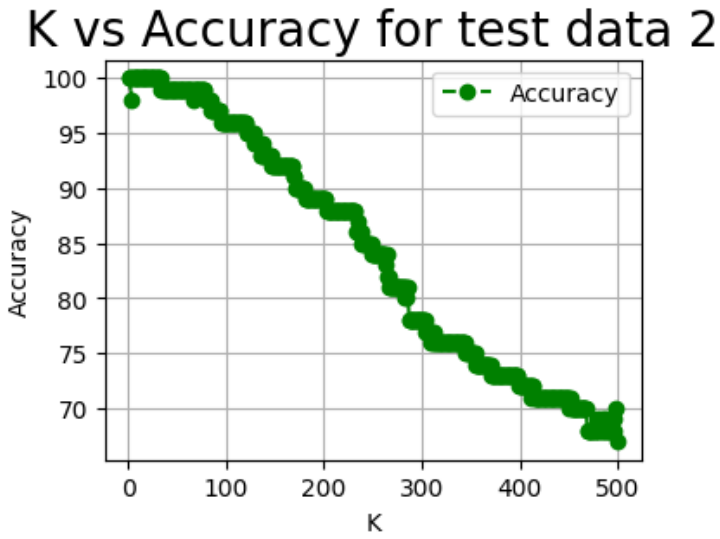
4) Confusion matrices on train and test data for best model(K=10) on Dataset-2



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

Solution:

1) Analysis with different values of K on dataset-3
-¿ Below figure shows the graph K vs Accuracy

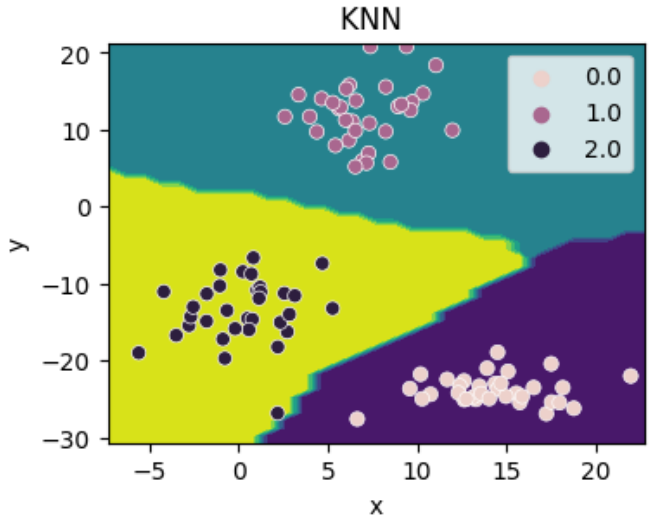


- so i have choosen k = 10 for the rest of the problem(BEST MODEL)

2) Accuracy on both train and test data for best model(K=10)

- Accuracy on train data = 100
- Accuracy on test data = 100

3) Plot of test data with my classification boundary for best model(k=10) on Dataset2



4) Confusion matrices on train and test data for best model(K=10) on Dataset-2