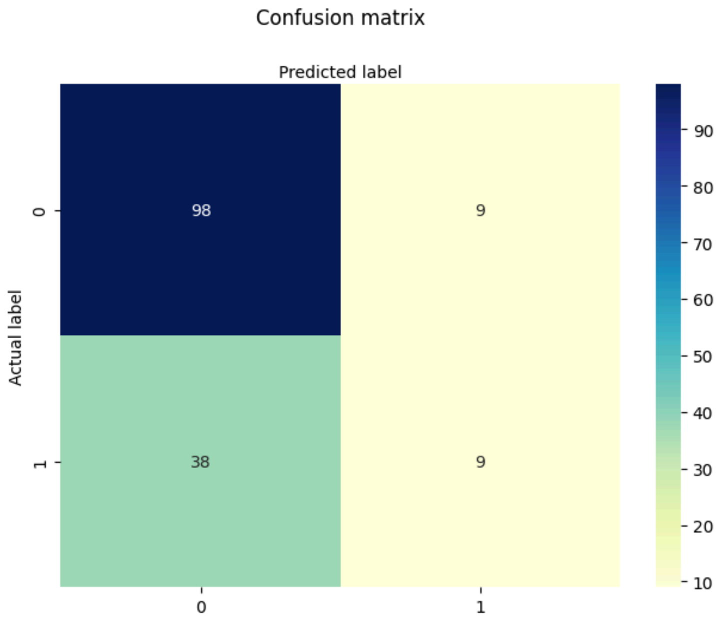
Nathon Tadeo
Student ID: 801265462
10/16/24
Homework 3
Github:https://github.com/nathon-tadeo/Intro-to-ML/blob/main/homework_3_intro_to_ml.ipynb

Problem 1 (20 points)

Using the diabetes dataset, build a logistic regression binary classifier for positive diabetes. Please use 80% and 20% split between training and evaluation (test). Make sure to perform proper scaling and standardization before your training. Draw your training results, including loss and classification accuracy over iterations. Also, report your results, including accuracy, precision, and recall, FI score. At the end, plot the confusion matrix representing your binary classifier.

Coding the logistic regression for the diabetes dataset was greatly facilitated because of the built-in function libraries for gradient descent, training, and validation. The data set went through the proper requirements of the problem from the training and test split, scaling, and standardization. The confusion matrix was plotted along with a classification report stating the accuracy, precision, recall, and F1 score. After all the preprocessing of the dataset, the accuracy equated to 0.6948, the precision was 0.5, the recall was 0.1915, and the F1 score equated to 0.2769. While the model's accuracy was quite reasonable(0.69) with the recall being 0.1915 it could be deemed that the model had trouble identifying true (+) diabetes cases. The low F1 score shows the imbalance between precision and recall. All plots are confusion matrices and graphs are contained in "Tadeo_Nathon 801265462 (2)Homework3".

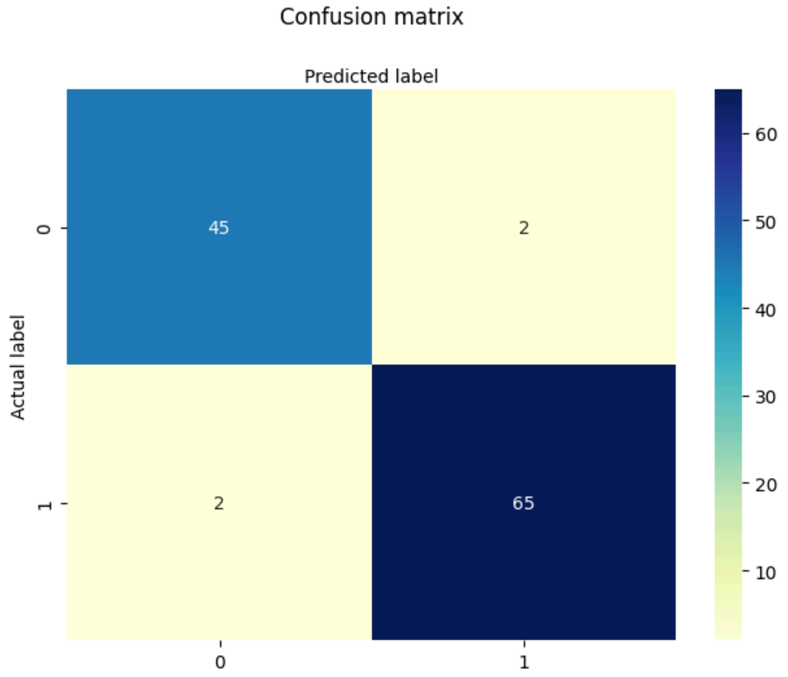


Problem 2 (20pts):

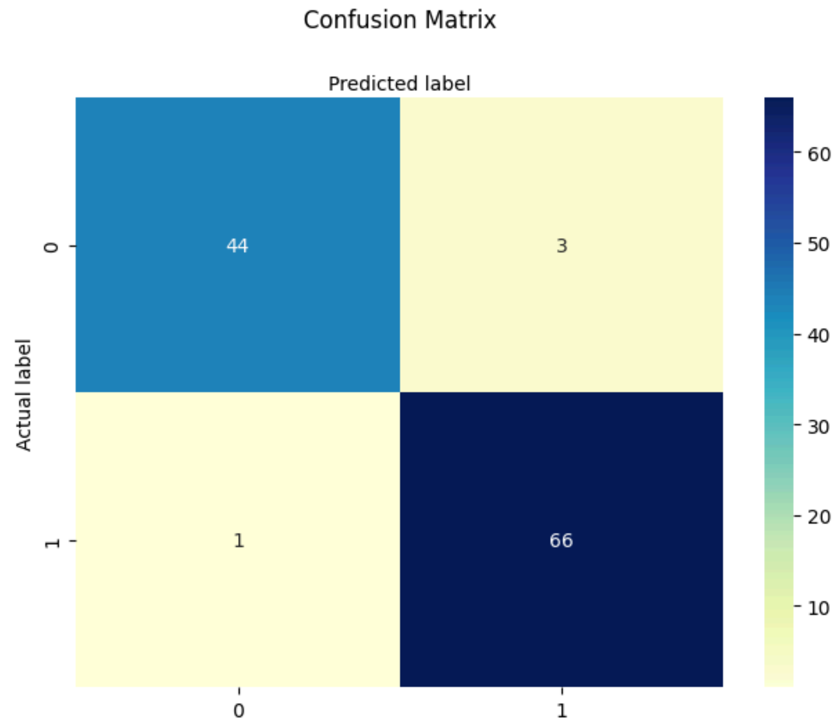
1. Use the cancer dataset to build a logistic regression model to classify the type of cancer (Malignant vs. benign). First, create a logistic regression that takes all 30 input features for classification. Please use 80% and 20% split between training and evaluation (test). Make sure to perform proper scaling and standardization before your training. Draw your training results, including loss and classification accuracy over iterations. Also, report your results, including accuracy, precision, recall and F1 score. At the end, plot the confusion matrix representing your binary classifier.
2. How about adding a weight penalty here, considering the number of parameters. Add the weight penalty and repeat the training and report the results.

Essentially the same process was for the logistic regression model for the cancer dataset, but this data set had to be imported through the "sklearn.datasets import" rather than importing a raw GitHub file. The cancer data set went through all the splitting, scaling, and standardization. When reporting the accuracy, precision, recall, and F1, both the non-weight penalty and weight penalty(L2 regularization) were plotted and reported. For the logistic regression model without a weight penalty, the accuracy, precision, recall, and F1 score all were around 0.96-0.97. The same trend followed for the weight penalty version, except the precision was 0.95 and the recall was 0.98. Both models were highly effective in classifying malignant and benign tumors.

Not Weighted



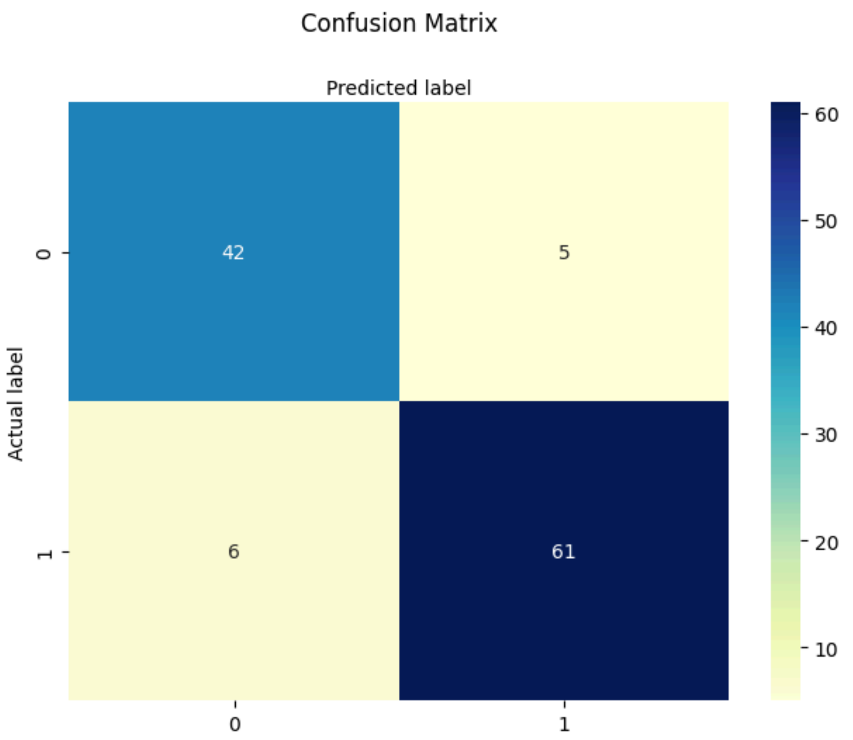
Weighted



Problem 3 (20pts):

Use the cancer dataset to build a naive Bayesian model to classify the type of cancer (Malignant vs. benign). Use 80% and 20% split between training and evaluation (test). Plot your classification accuracy, precision, recall, and F1 score. Explain and elaborate on your results. Can you compare your results against the logistic regression classifier you did in Problem 2.

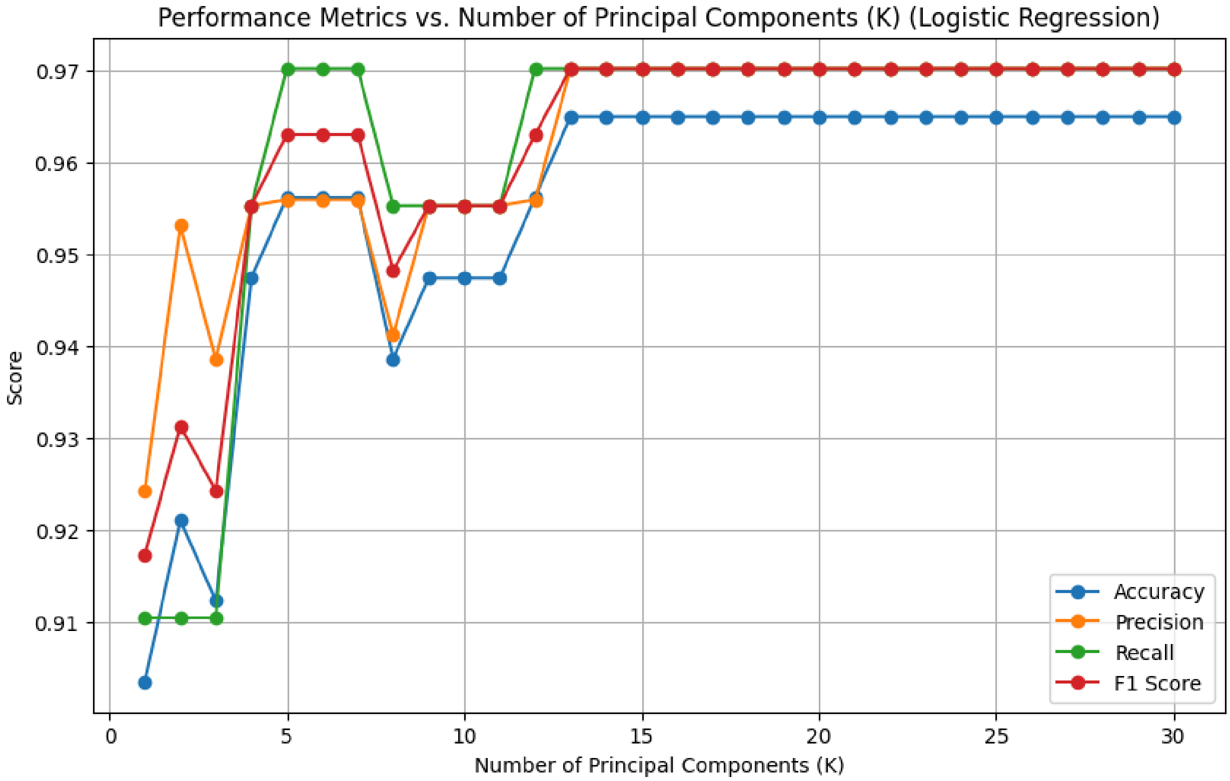
Problem 3 followed the same process as Problem 2, but instead of logistic regression as the model, a Naive Bayesian classifier was applied. This model equated to an accuracy of 0.9, precision of 0.92, recall of 0.91, and F1 score of 0.92. The logistic model from problem 2 had 0.6 more accuracy(0.96) than the Bayesian model. Overall the Bayesian model equated to lower scores in all categories being around 0.4 lower compared to the logistic regression model



Problem 4 (20pts):

Use the cancer dataset to build a logistic regression model to classify the type of cancer (Malignant vs. benign). Use the PCA feature extraction for your training. Perform N number of independent training (N=1, …, K). Identify the optimum number of K, principal components that achieve the highest classification accuracy. Plot your classification accuracy, precision, recall, and F1 score over a different number of Ks. Explain and elaborate on your results and compare it against problems 2 and 3.

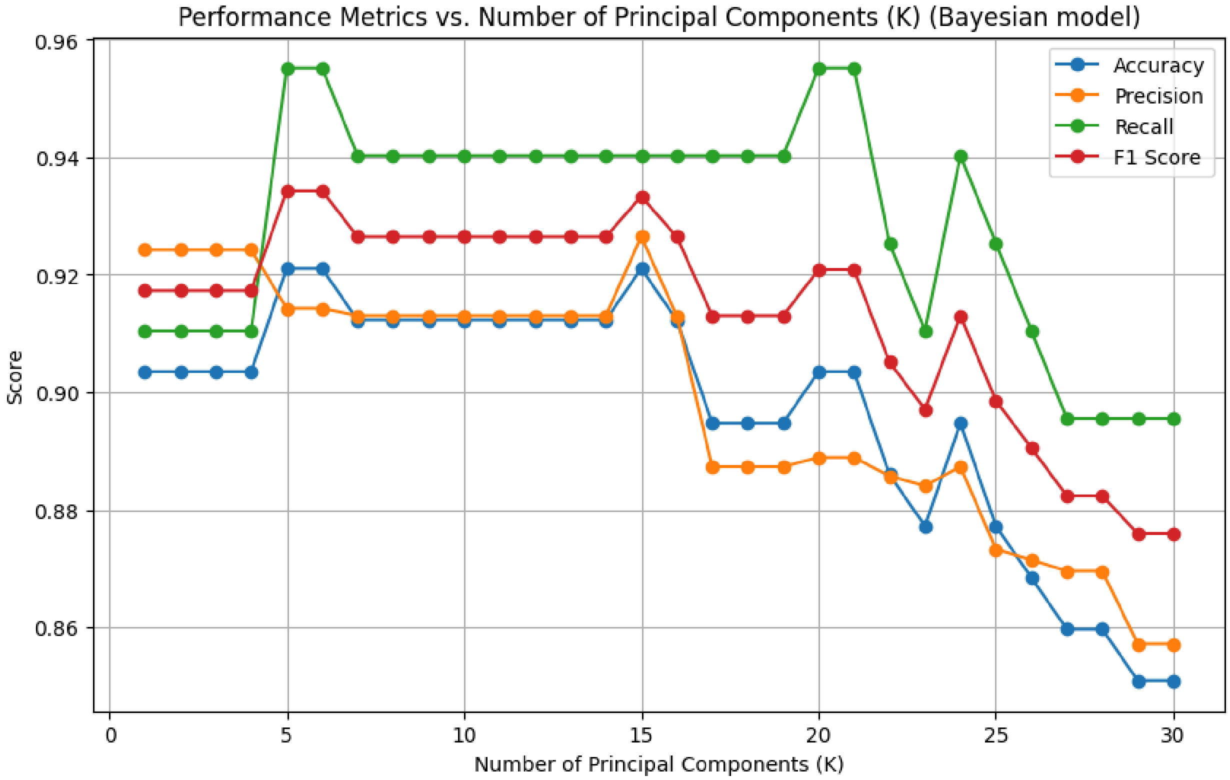
Followed the code structure for Problem 2, except the PCA feature extraction was used for training. K ranged from 1 to 30 to accommodate the whole dataset, and all the classifications were plotted on a graph to showcase the number of principal components. According to the graph, all the scores seemed to flatten out when k is greater than 13. Accuracy precision-recall and F1 score seem to flatten out around 0.97 Compared to problems 2 and 3, when k is greater than 13, the scores seemed to match problem 2's model



Problem 5 (20pts):

Can you repeat problem 4? This time, replace the Bayes classifier with logistic regression. Report your results (classification accuracy, precision, recall and F1 score). Compare your results against problems 2, 3, and 4.

Followed the code structure for Problem 4, except the Bayes classifier was used According to the graph, the scores seemed to be the greatest when k equaled 5. Accuracy precision, recall, and F1 do not seem to flatten out at all compared to 4 and have a downward trend when k approaches 30. Compared to problem 4, the score has a downward trend while problem 4 has an upward trend while approaching 30.


Performance Metrics vs. Number of Principal Components (K) (Bayesian model)
Score
Number of Principal Components (K)
0.96
0.94
0.92
0.90
0.88
0.86
0
5
10
15
20
25
30
Accuracy
Precision
Recall
F1 Score