Name: Long Nguyen
Student ID: 801235507
Homework #: 3

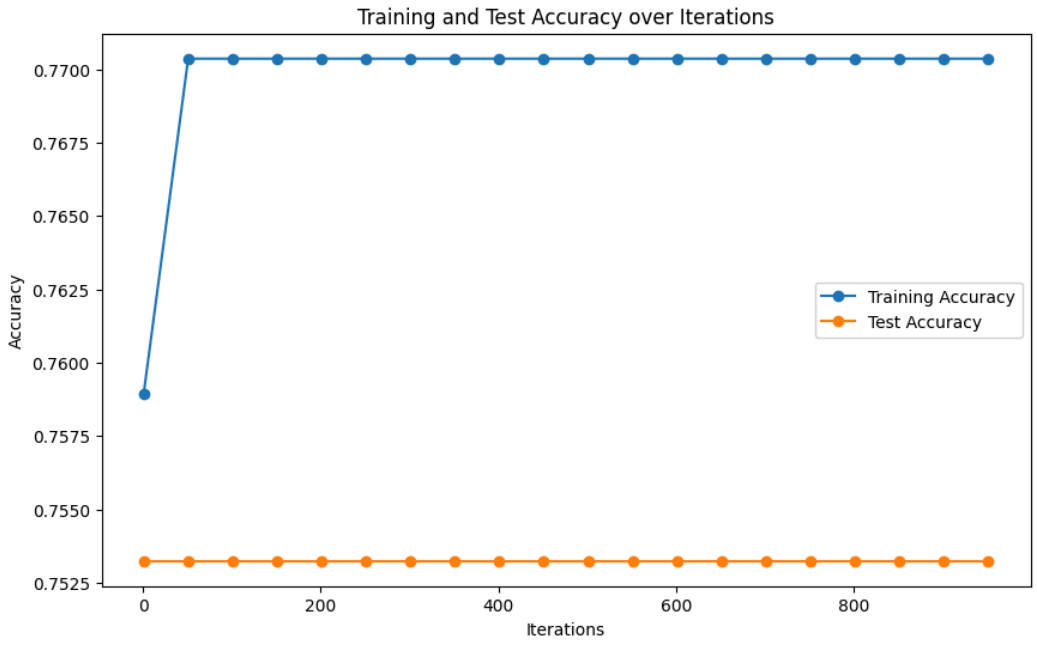
Use the Diabetes dataset and Cancer dataset, use the built-in function from ML libraries for gradient descent, training, and validation to solve the problems.

Link to Github repository: https://github.com/lnguye782/ECGR-4105-Intro-to-ML/tree/main/HW3

Problem 1:

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 the training. Draw the training results, including loss and classification accuracy over iterations. Also, report the results, including accuracy, precision, and recall, F1 score. At the end, plot the confusion matrix representing the binary classifier.

Link to Google Colab: ECGR 4105 - HW3 - Problem 1.ipynb

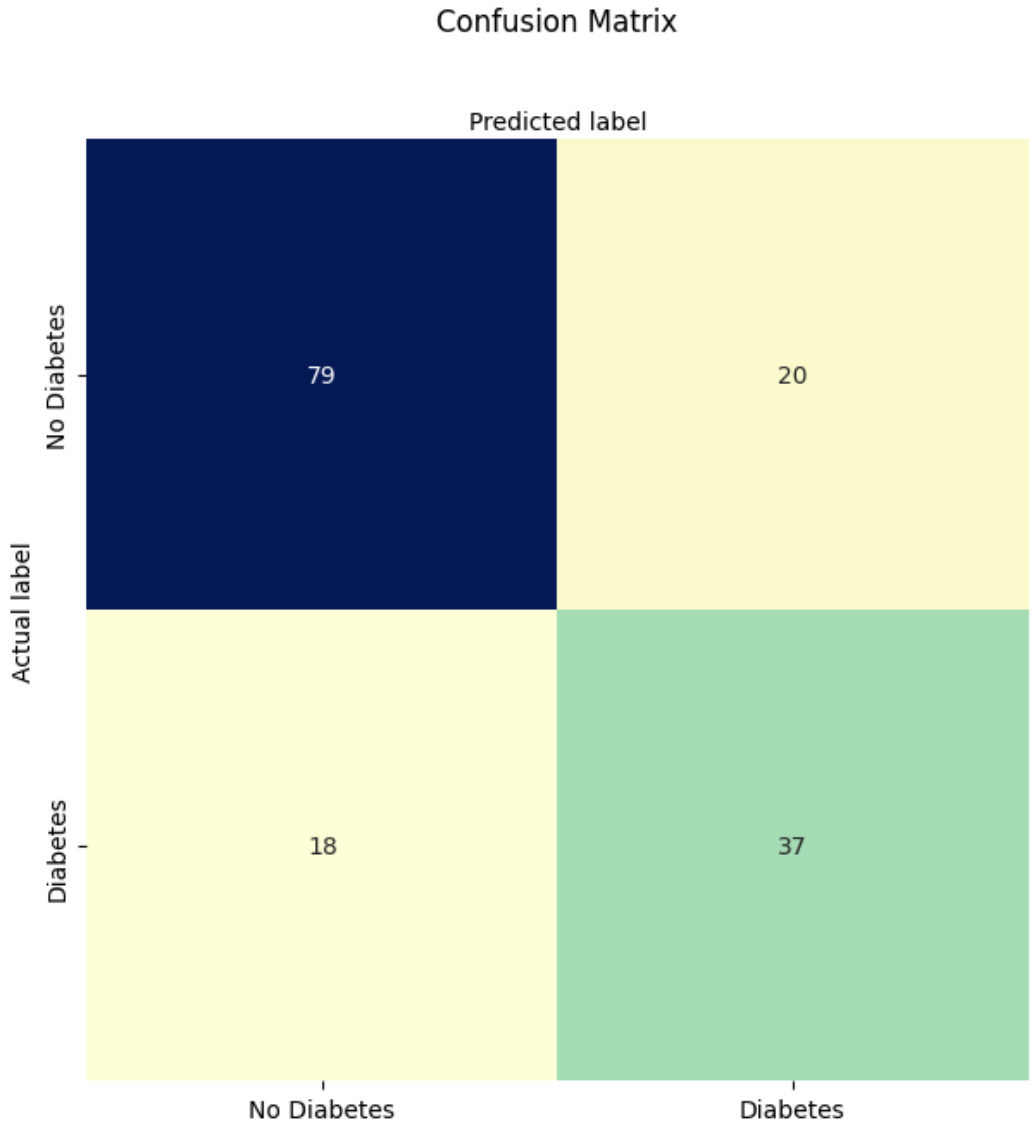



```
Accuracy: 0.7532467532467533
Precision: 0.6491228070175439
Recall: 0.6727272727272727
F1 Score: 0.6607142857142857
```

+ Accuracy: 75.32%
+ Precision: 64.91%
+ Recall: 67.27%
+ F1 Score: 66.07%

Name: Long Nguyen
Student ID: 801235507
Homework #: 3

```
Text(45.72222222222221, 0.5, 'Actual label')
```



Problem 2:

a) 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 the training. Also, report the results, including accuracy, precision, recall and F1 score. At the end, plot the confusion matrix representing your binary classifier.

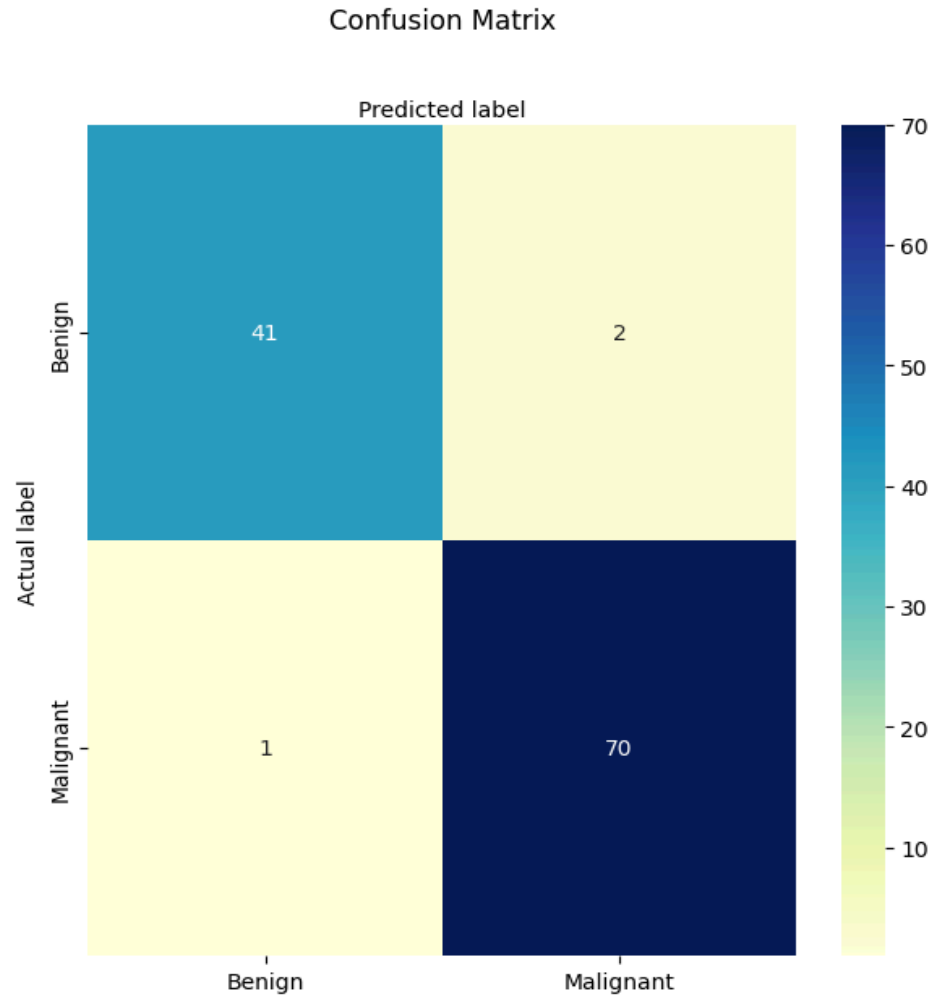
Link to Google Colab: ECGR 4105 - HW3 - Problem 2.ipynb

Name: Long Nguyen
Student ID: 801235507
Homework #: 3

```
Accuracy: 0.9736842105263158
Precision: 0.9722222222222222
Recall: 0.9859154929577465
F1 Score: 0.9790209790209791
```

+ Accuracy: 97.37%
+ Precision: 97.22%
+ Recall: 98.59%
+ F1 Score: 97.90%

```
Text(45.72222222222214, 0.5, 'Actual label')
```



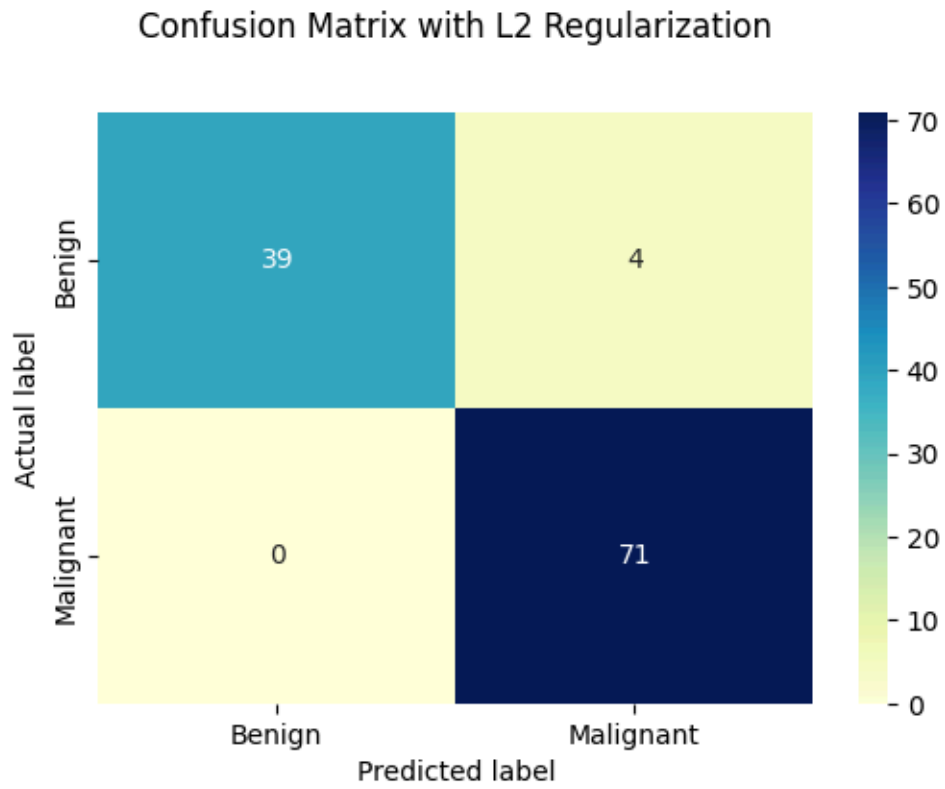
b) How about adding a weight penalty here, considering the number of parameters. Add the weight penalty and repeat the training and report the results.

Name: Long Nguyen
Student ID: 801235507
Homework #: 3

```
Accuracy with L2: 0.9649122807017544
Precision with L2: 0.9466666666666667
Recall with L2: 1.0
F1 Score with L2: 0.9726027397260274
```

+ Accuracy: 96.49%
+ Precision: 94.67%
+ Recall: 100%
+ F1 Score: 97.26%

```
Text(45.722222222222214, 0.5, 'Actual label')
```



Problem 3:

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.

Link to Google Colab: ECGR 4105 - HW3 - Problem 3.ipynb

Name: Long Nguyen
Student ID: 801235507
Homework #: 3

```
Accuracy with Naive Bayes: 0.9649122807017544
Precision with Naive Bayes: 0.958904109589041
Recall with Naive Bayes: 0.9859154929577465
F1 Score with Naive Bayes: 0.9722222222222222
```

+ Accuracy: 96.49%
+ Precision: 95.89%
+ Recall: 98.59%
+ F1 Score: 97.22%

	Accuracy	Precision	Recall	F1 Score
No Weight Penalty	97.37%	97.22%	98.59%	97.90%
L2 Regularization	96.49%	94.67%	100%	97.26%
Naive Bayesian	96.49%	95.89%	98.59%	97.22%

Result Comparison:

+ **Accuracy**: The No Weight Penalty model has the highest accuracy (97.37%) while both L2 Regularization and Naive Bayesian models achieve the same accuracy (96.49%).
+ **Precision**: The No Weight Penalty model outperforms the other two models (97.22%). Naive Bayes has slightly better precision (95.89%) compared to L2 Regularization (94.67%), meaning that it makes fewer false positive predictions.
+ **Recall**: L2 Regularization (100%) outperforms both No Weight Penalty model and Naive Bayes model (98.59%) in recall, meaning that it was better at identifying all malignant cases.
+ **F1 Score**: All three models have nearly identical F1 scores, balancing precision and recall effectively.

Problem 4:

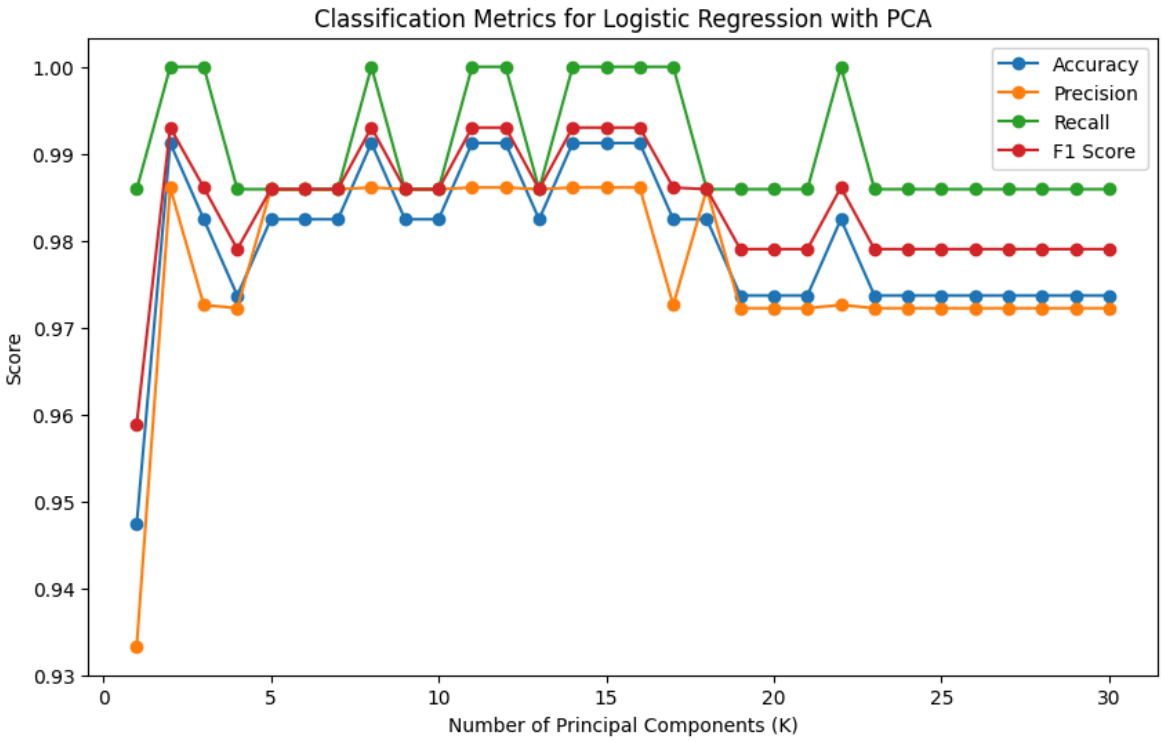
Use the cancer dataset to build a logistic regression model to classify the type of cancer (Malignant vs. Benign). Use the PCA feature extraction for your training. Perform N number of independent training (N=1, …, K). Identify the optimum number of K, principal components that achieve the highest classification 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.

Name: Long Nguyen
Student ID: 801235507
Homework #: 3

Link to Google Colab: ECGR 4105 - HW3 - Problem 4.ipynb

```
(2, 0.9912280701754386)
```

The optimum number of K that achieves the highest classification accuracy is 2, with an accuracy of 99.12%.



By using PCA to reduce the dimensionality of the dataset and training the logistic regression model over different numbers of principal components (K=1 to 30), it shows that the best classification performance is obtained with just 2 principal components (K=2).

After K=2, the additional components did not provide much improvement in classification accuracy.

Result Comparison:

+ The logistic regression with PCA (with K=2) outperforms all three models from Problems 2 and 3 in terms of accuracy (99.12%).

Name: Long Nguyen
Student ID: 801235507
Homework #: 3

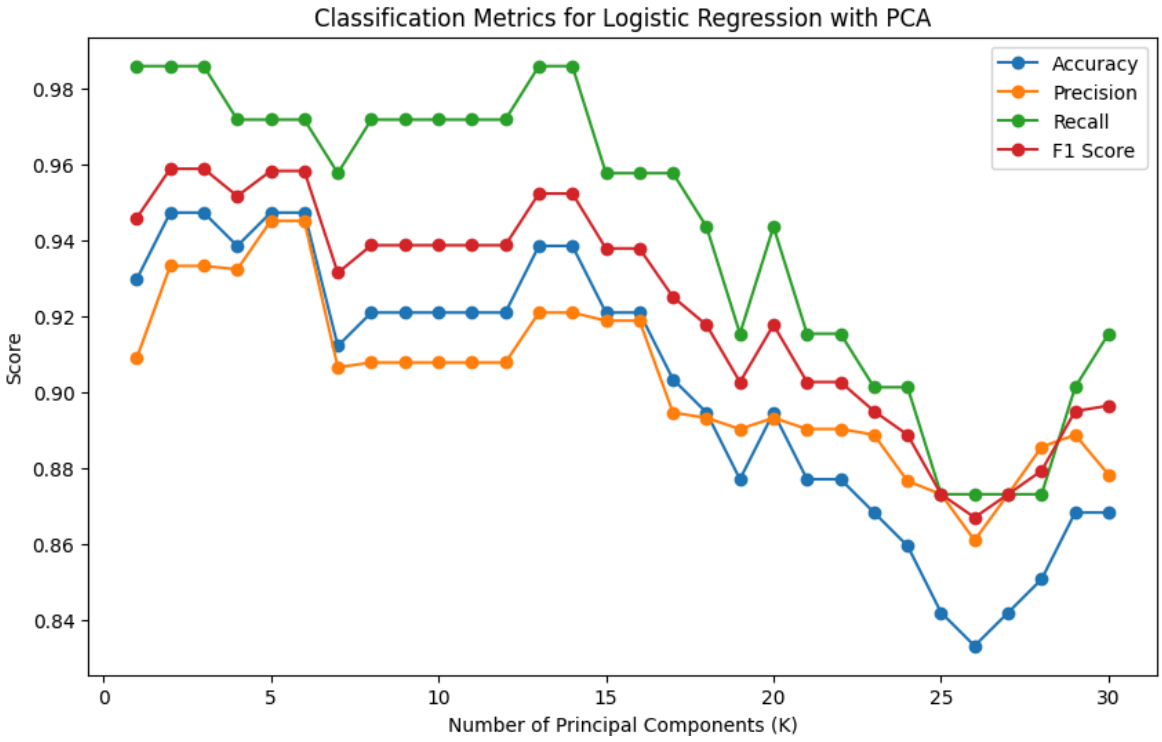
Problem 5:

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

Link to Google Colab: ECGR 4105 - HW3 - Problem 5.ipynb

```
(2, 0.9473684210526315)
```

The optimum number of K that achieves the highest classification accuracy for the Naive Bayes classifier is 2, with an accuracy of 94.74%.



Logistic regression with PCA performed the best (99.12% accuracy), while Naive Bayes with PCA had lower accuracy (94.74%), likely because Naive Bayes assumes independence between features, which can be disrupted by PCA's transformations.

Result Comparison:

+ Problem 2 (Logistic Regression with L2 regularization): Achieved an accuracy of 96.49%.
+ Problem 3 (Naive Bayes without PCA): Achieved an accuracy of 96.49%.

Name: Long Nguyen
Student ID: 801235507
Homework #: 3

+ Problem 4 (Logistic Regression with PCA): Achieved the highest accuracy of 99.12% with 2 principal components.

+ Problem 5 (Naive Bayes with PCA): Achieved a slightly lower accuracy of 94.74% with 2 principal components.