Intro ML Homework 3

Jaskin Kabir
Student Id: 801186717
GitHub:
https://github.com/jaskinkabir/Intro_ML/tree/main/HM2

October 2024

1 Diabetes Classification

For the first problem, the diabetes dataset was fed into my custom logistic classifier class that handles the 80-20 % train test split and feature scaling internally. The training parameter values $\alpha = 0.1$, $\lambda = 0$ were used in a 250 iteration training process that resulted in the plot shown in Figure 1.

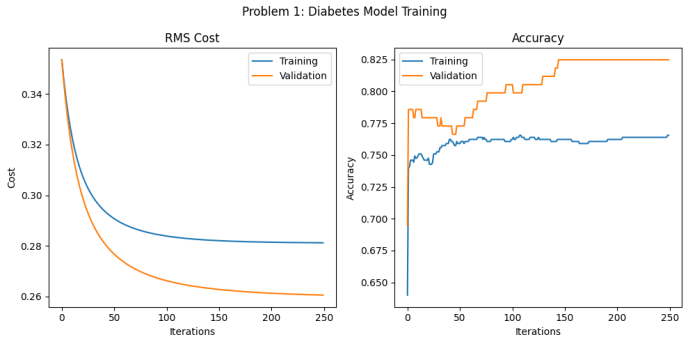

Figure 1: Diabetes Classifier Training Loss and Classification Accuracy

The final model's performance left much to be desired, with an accuracy of only 83 %. This is likely because the dataset only includes eight features, which is not enough dimensionality for a logistic regression to be able to accurately classify the data. The accuracy, precision, recall, and F1 scores can be seen in Figure 2.

Metric	**Value**
Accuracy	0.83
Precision	0.76
Recall	0.62
F1 Score	0.68

Figure 2: Diabetes Classifier Performance Metrics

The recall score is much lower than the precision score, which means the model is more likely to classify false negatives than it is false positives. The confusion matrix in Figure 3 supports this, as it shows that the model classified twice as many false negatives as it did false positives.

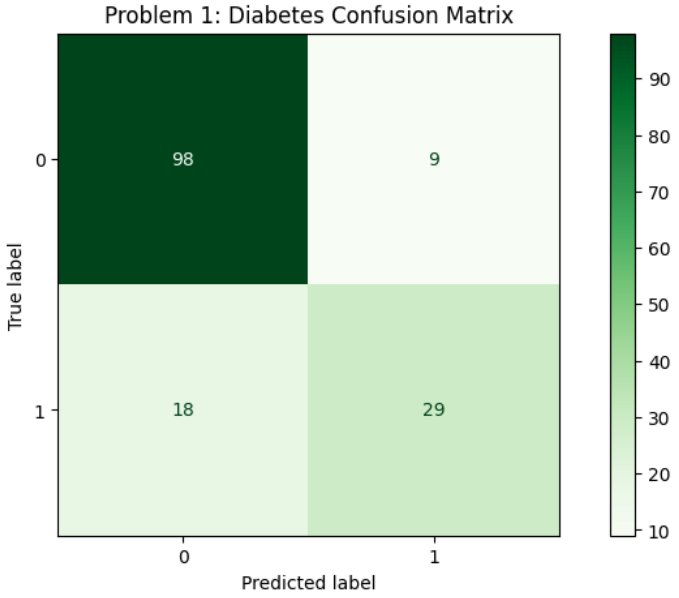

Figure 3: Diabetes Classifier Confusion Matrix

2 Logistic Breast Cancer Classification

1. **No Parameter Penalty** The same custom logistic classifier class was used to train a classifier based on sklearn's breast cancer dataset. The model was trained with the following training parameters: $\alpha = 0.1$, $\lambda = 0$ over 250 iterations. The training process generated the plots shown in Figure 4.

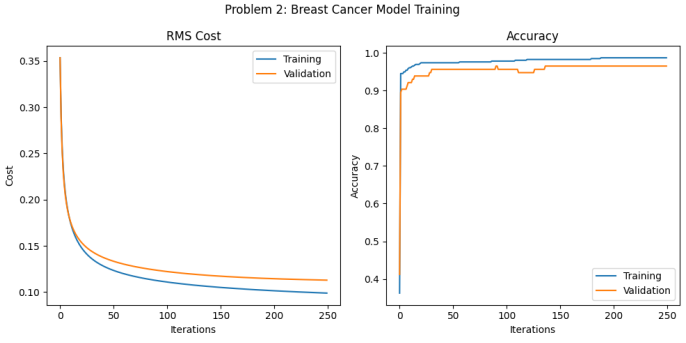

Figure 4: Logistic Breast Cancer Classifier Training

This model, whose performance metrics are shown in Figure 5, performed

much better than the diabetes classifier. This is likely because this dataset had 31 features to describe each datapoint compared to the eight features in the diabetes dataset.

Metric	Value
Accuracy	0.96
Precision	0.96
Recall	0.99
F1 Score	0.97

Figure 5: Logistic Breast Cancer Classifier Performance Metrics

The precision score was slightly lower than the recall score, and this is reflected by the confusion matrix in Figure 6 showing that the model tends to classify false negatives slightly more frequently than it does false positives. However, this difference is so small that it is likely insignificant.

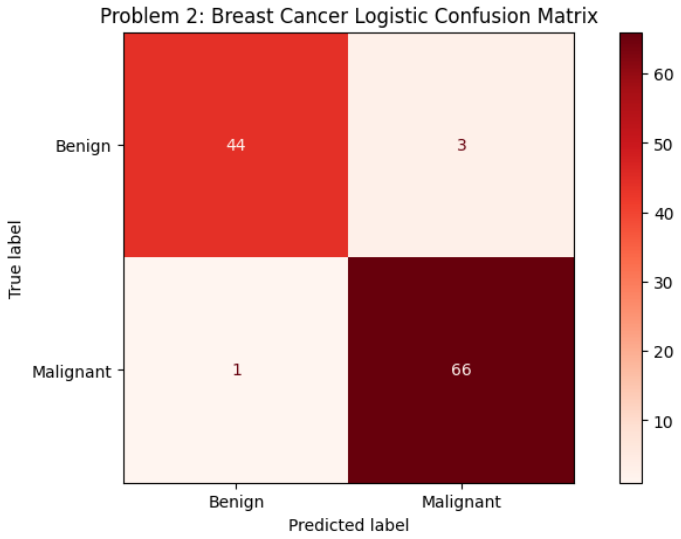

Figure 6: Logistic Breast Cancer Classifier Confusion Matrix

2. **Parameter Penalty**

   By iterating over many different values for the parameter penalty λ and comparing the performance metrics, no value was found that improved any of the metrics when compared to the training done with $\lambda = 0$. While the loss plot in Figure 4 does show slight signs of overfit, this effect is likely not pronounced enough to benefit from a parameter penalty. Thus, the optimal value for the training parameter λ is 0.

3 Naive Bayes Classification

A custom Python class that implements the Gaussian naive Bayes classification technique was developed and trained on this same breast cancer dataset. Its performance scores and their relative difference to the logistic classifier can be seen in Figure 7

Metric	**Value**	**Difference To Logistic Classifier**
Accuracy	0.91	-5.2 %
Precision	0.93	-3.13 %
Recall	0.93	-6.05 %
F1 Score	0.93	-4.12 %

Figure 7: Naive Bayes Vs. Logistic Breast Cancer Classifier Metrics

This model performed slightly worse than the logistic classifier. This is to be expected, as the naive Bayes technique's advantage is in its explainability, not its performance.

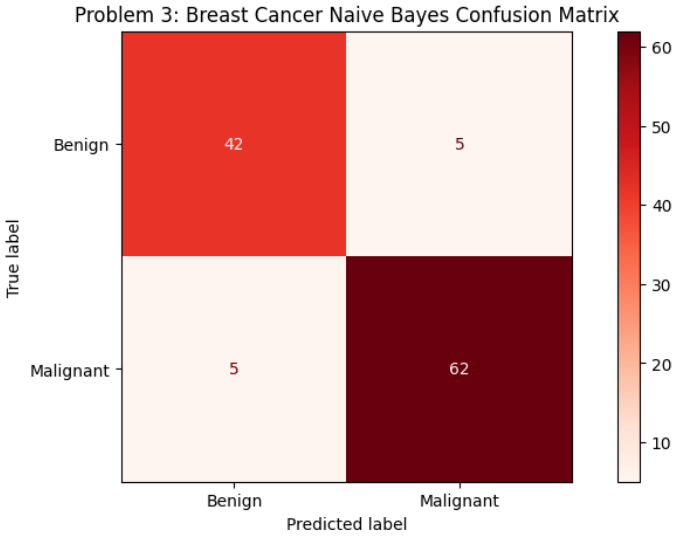

Figure 8: Naive Bayes Confusion Matrix

The precision and recall scores were identical for this model, which is reflected in the confusion matrix shown in Figure 8. The model classified an equal number of false positives and false negatives.

4 PCA For Logistic Classification

By employing the technique of Principal Component Analysis, the 31-dimensional data can be mapped to a k-dimensional space, where $k < 31$, that may be easier to train a model on. To find the optimal dimensionality of this space, the PCA technique was run on the dataset with a varying value of k, and the four performance metrics were measured at each iteration. This process generated the graph shown in Figure 9.

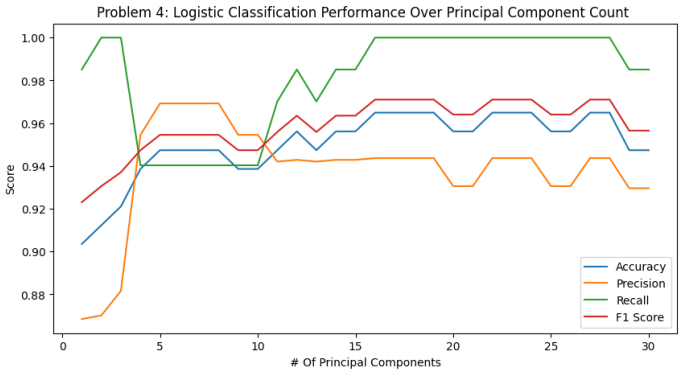

Figure 9: PCA Optimization for Logistic Classifier

Upon analysis of this graph, the optimal number of principal components k is 16. Precision is at its highest at $k = 5$, but the other three metrics reach their peaks at $k = 16$. In the case of breast cancer classification, maximizing the recall score is more important than precision, as maximizing recall equates to minimizing false negatives. A breast cancer test should minimize false negatives so that anyone who has cancer can be treated.

Metric	**Value**	**Difference To Logistic Classifier**	**Difference To Bayes Classifier**
Accuracy	0.96	0.00 %	5.50 %
Precision	0.94	-2.08 %	1.08 %
Recall	1.00	1.01 %	7.53 %
F1 Score	0.97	0.00 %	4.30 %

Figure 10: Performance of Logistic Classifier with PCA

The model trained on the 16-dimensional mapping of the breast cancer data was scored and its performance metrics can be seen in Figure 10 alongside a comparison with the logistic and Bayes techniques without feature extraction. Compared to the logistic classifier, the accuracy and F1 scores were identical. However, the precision was lower and the recall was higher. This may be a worthwhile sacrifice because, as discussed earlier, a maximizing recall is more important than maximizing precision in this case. This model outperformed

the Bayesian classifier in a very similar way as the logistic classifier with no feature extraction. However, the difference in precision was lessened, while the difference in recall was increased.

5 PCA For Naive Bayes Classification

The same technique of iterating over different values of k was repeated for the Naive Bayes classifier. The resulting performance plot can be seen in Figure 11.

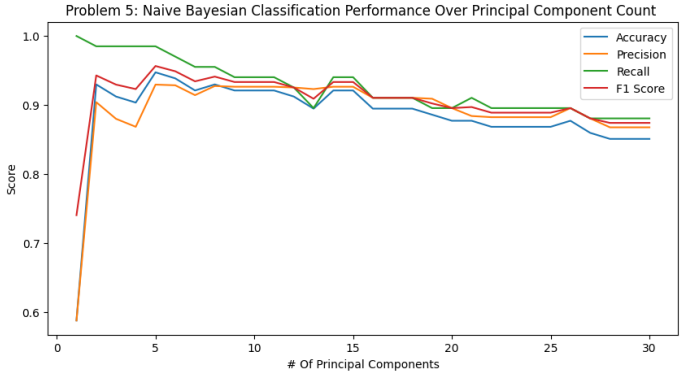

Figure 11: PCA Optimization for Naive Bayes Classifier

This graph shows that the ideal value of k is 5. This is true for all metrics except for recall, which reaches its peak value at $k = 1$. Even though recall is the most important metric for this application, the other metrics are far too low at $k = 1$ for this to be a worthwhile sacrifice.

Metric	**Value**	**Difference To Logistic Classifier**	**Difference To Bayes Classifier**	**Difference To Logistic Classifier With PCA**
Accuracy	0.95	-1.04 %	4.50 %	-1.04 %
Precision	0.93	-3.13 %	0.00 %	-1.06 %
Recall	0.99	0.00 %	6.45 %	-1.00 %
F1 Score	0.96	-1.03 %	3.23 %	-1.03 %

Figure 12: Performance of Bayes Classifier with PCA

The performance metrics for the Bayesian classifier trained on PCA data with $k = 5$ and their comparisons to the previous models can be seen in Figure 12. This model outperforms the standard Bayes classifier in all metrics except for precision. This may be due to the reduction in features limiting the effects of overfit. The model is still outperformed by the standard logistic classifier, but this difference is much lower when compared to the standard Bayes classifier. In fact, it had an identical recall score to the standard logistic classifier. This

model was also outperformed by the logistic classifier with feature extraction, but the difference is even smaller.