SENG 474 Assignment 2

Cross-Validation for Hyperparameter Tuning and Model Selection

Ryan Russell

July 12, 2020

1. Logistic Regression Analysis

Logistic Regression was the first classification method used on the fashion-MNIST dataset. Logistic Regression is a predictive analysis machine learning method used primarily for classification problems [2]. This method uses the sigmoid function to map its predictions to probabilities. The expected result of Logistic Regression, when inputs are pushed through a prediction function, is a set of classes devised from the prediction function's probability scores between 0 and 1.

The specific implementation used was Scikit Learn's Logistic Regression with L2 regularization. Due to the inclusion of L2 regularization, the training objective was penalized according to the sum of squares of the weights. It should be noted that C is known as the regularization parameter, and lower C indicates greater regularization. To reduce the training time required, 6000 training examples from the "Sandal" and "Sneaker" classes (3000 from each) were used to train the model. 2000 test examples from the same classes were used for evaluation. As part of the analysis, the regularization parameter C was varied on a logarithmic scale from 10^{-7} to 10^{7}. In total, 20 different values for C were tested. The test accuracy and training accuracy plotted against the logarithmic scale for the regularization parameter C is displayed in Figures 1, 2, and 3.

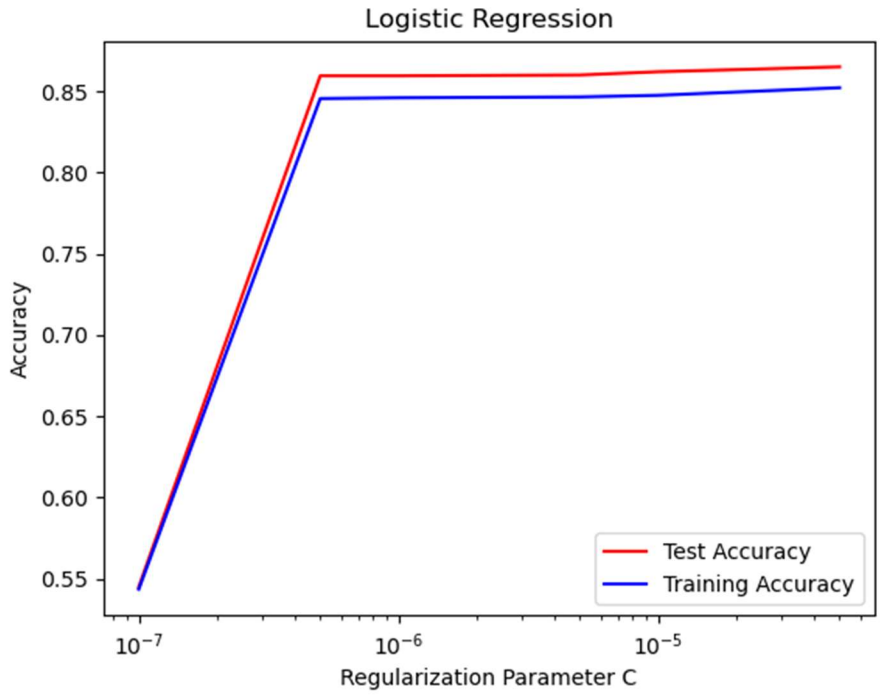

Figure 1: Logistic Regression Accuracy with Low C Values

Figure 1, which includes relatively low C values from 10^{-7} to 10^{-4}, showcases a strong example of underfitting. In this case, underfitting indicates too much regularization. While there is a significant jump in test accuracy from 10^{-7} to 10^{-6}, the maximum accuracy in this range is still just over 0.86, which is considerably lower than the accuracies shown in Figure 2. Interestingly, the test accuracy is slightly higher than the training accuracy for most values. This result can likely be attributed to the over-regularization as well.

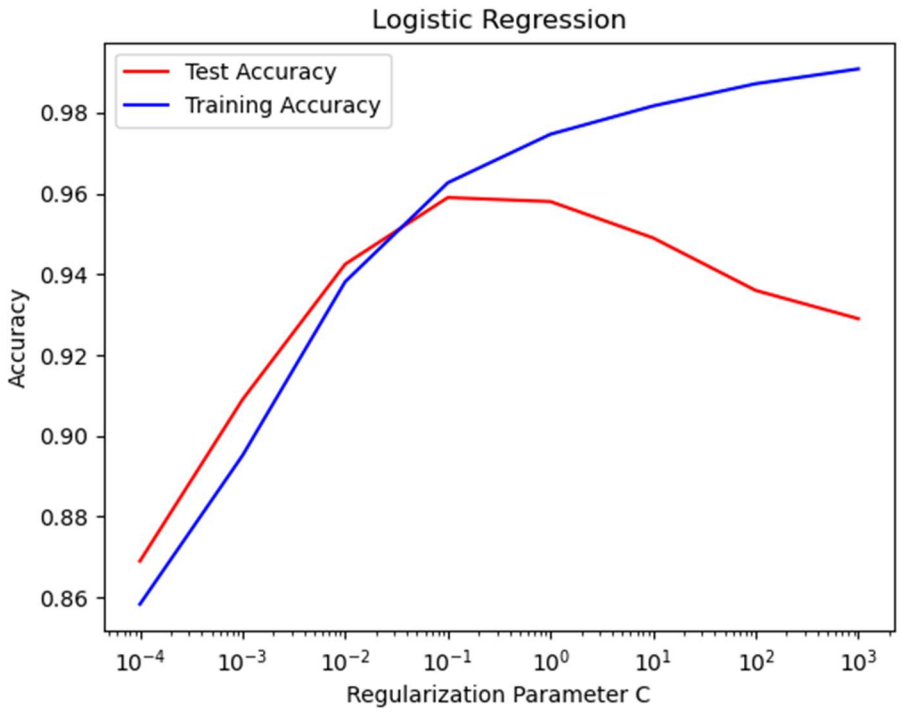

Figure 2: Logistic Regression Accuracy with Midrange C Values

Figure 2, which includes the midrange of tested C values from 10^{-4} to 10^{3}, displays the test accuracy peak as well as the point at which overfitting starts to occur. The test accuracy rises sharply from 10^{-4} to 10^{-2} before peaking at roughly 0.1, where the accuracy is 0.96. From 1 to 10^{3}, the accuracy begins to decline as overfitting occurs, eventually dropping below 0.94. From this, we can observe that a small range of C values surrounding C = 0.1 will produce the most accurate results.

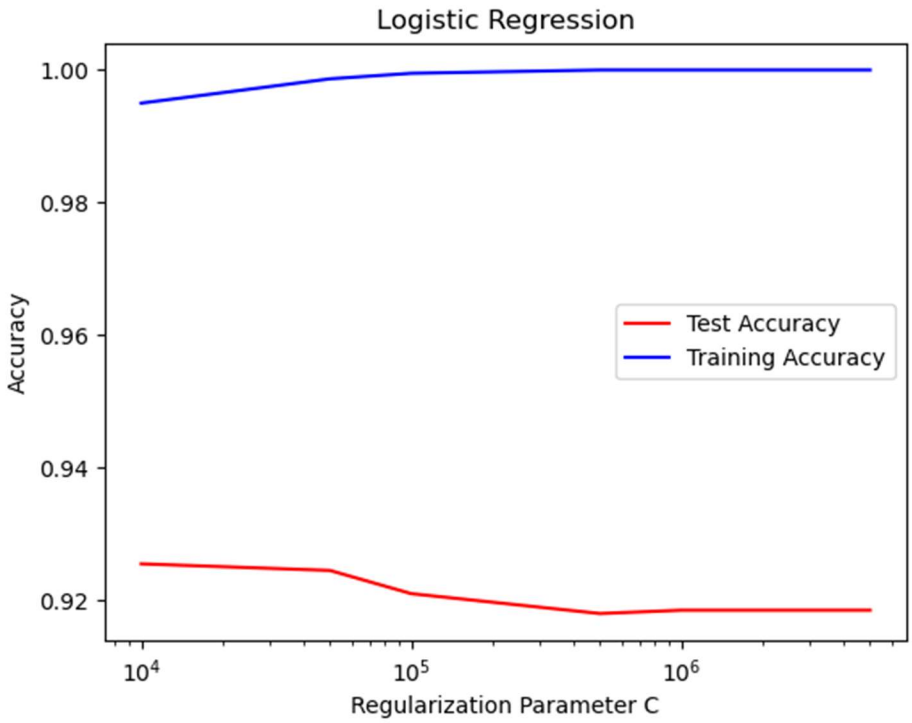

Figure 3: Logistic Regression Accuracy with High C Values

Figure 3 includes relatively high C values from 10^4 to 10^7 and strongly exhibits an example of overfitting. In this case, overfitting indicates too little regularization. C values of 10^5 and higher produce training accuracies of 1.00 while the test accuracy continues in a slightly downward trend, dropping below 0.92. From analysis of Figures 1, 2, and 3, it can be observed that the accuracy of Logistic Regression on the fashion-MNIST dataset benefits from a C value near 0.1. Furthermore, Logistic Regression is not immune to overfitting or underfitting. An interesting follow-up experiment could involve varying other hyperparameters than the regularization value and testing on a multi-class classification problem rather than a binary one, which could be achieved on the same dataset by including all classes, not just those for sandals and sneakers.

2. Linear Support Vector Machine Analysis

A Support Vector Machine with a linear kernel was the second method used on the fashion-MNIST dataset. Support Vector Machines (SVMs) can be used for both classification and regression problems. The purpose of an SVM is to find a hyperplane (a line for two-dimensional data) that classifies the data of an n-dimensional space [3]. This hyperplane should ideally maximize the margin, or the distance between clusters of data for each class. The samples closest to the hyperplane are known as support vectors. The SVM version used in this experiment is known as a soft-margin SVM, which is usable for data that may not be linearly separable.

The specific implementation used was Scikit Learn's Support Vector Classifier with a linear kernel. The regularization parameter C is used for SVMs as well as Logistic Regression, and lower C also indicates greater regularization in this case. To reduce the training time required, 6000 training examples from the "Sandal" and "Sneaker" classes (3000 from each) were used to train the model. 2000 test examples from the same classes were used for evaluation. As part of the analysis, the regularization parameter C was again varied on a logarithmic scale from 10^{-7} to 10^{7}. In total, 20 different values for C were tested. The test accuracy and training accuracy plotted against the logarithmic scale for the regularization parameter C is displayed in Figures 4, 5, and 6.

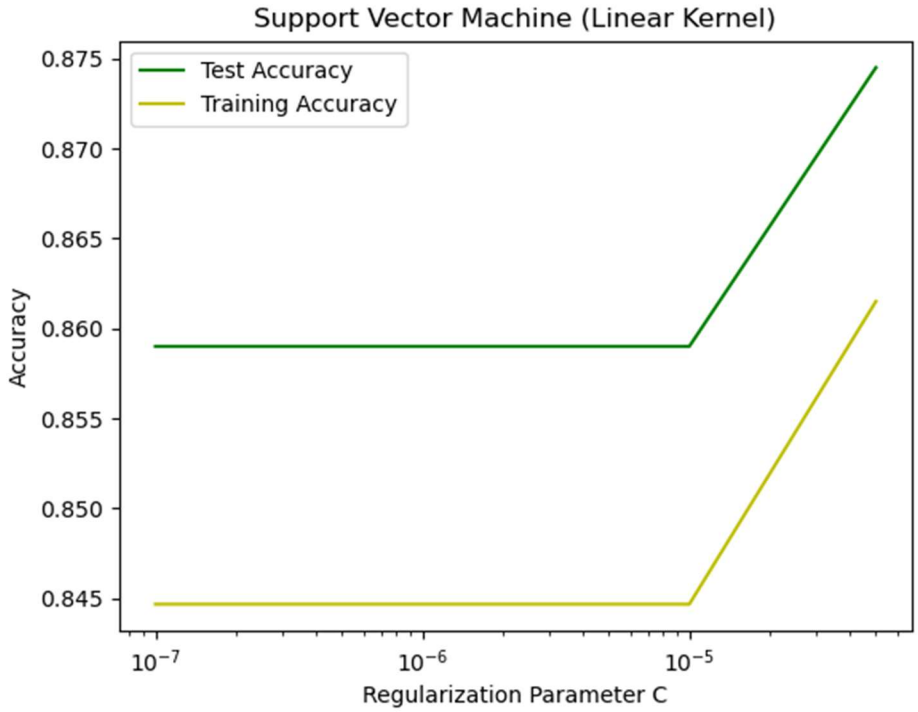

Figure 4: SVM Accuracy with Low C Values

Figure 4, which includes relatively low C values from 10^{-7} to 10^{-4}, showcases a strong example of underfitting. In this case, underfitting indicates too much regularization. While there is a noticeable increase in test accuracy following 10^{-5}, the maximum accuracy in this range is still just over 0.87, which is considerably lower than the accuracies in Figure 5. Similar to the results of Figure 1 for Logistic Regression, the test accuracy is actually higher than the training accuracy for each value in this range. However, we can see these values converge in the first quarter of values displayed in Figure 5.

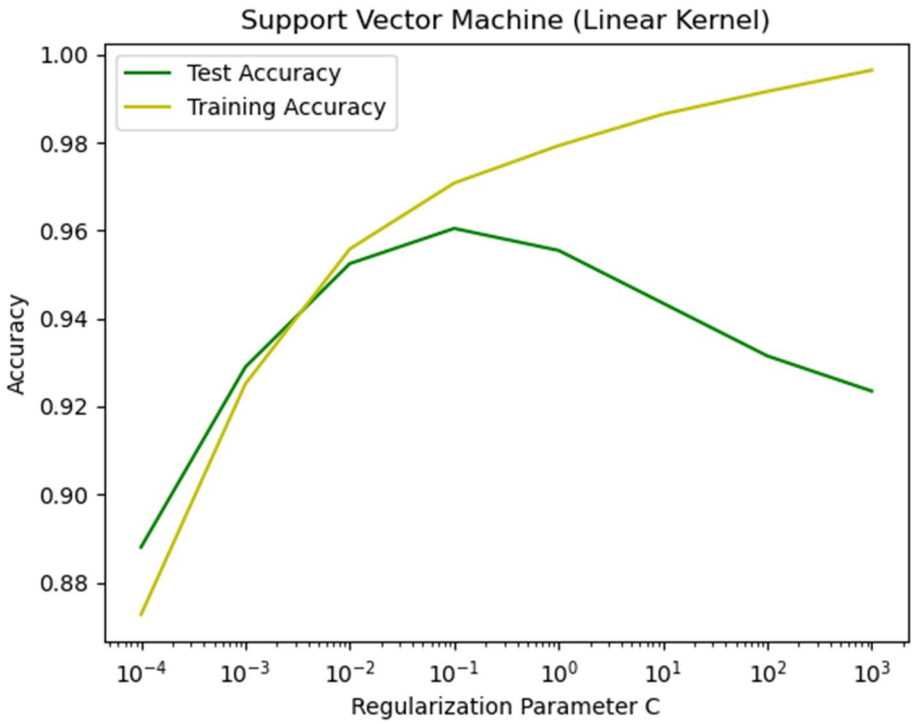

Figure 5: SVM Accuracy with Midrange C Values

Figure 5, which includes the midrange of tested C values from 10^{-4} to 10^{3}, displays the test accuracy peak as well as the point at which overfitting starts to occur. The test accuracy rises sharply from 10^{-4} to 10^{-2} before peaking at roughly 0.1, where the accuracy is just below 0.96. From 1 to 10^{3}, the accuracy begins to decline as overfitting occurs, eventually dropping below 0.93. From this, we can observe that a small range of C values surrounding C = 0.1 will produce the most accurate results. This observation mirrors the insights gained from the midrange of C values for Logistic Regression.

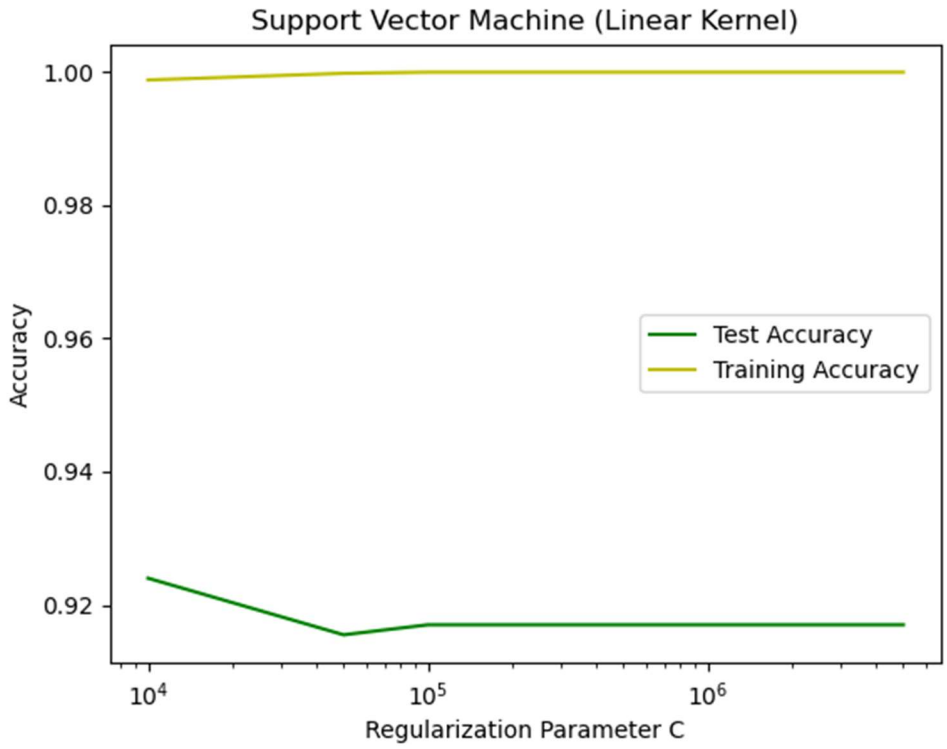

Figure 6: SVM Accuracy with High C Values

Figure 6 includes relatively high C values from 10^4 to 10^7 and strongly exhibits an example of overfitting. In this case, overfitting indicates too little regularization. Nearly all C values shown in Figure 6 produce training accuracies of 1.00 while the test accuracy is below 0.92. From analysis of Figures 4, 5, and 6, it can be observed that the accuracy of Linear Support Vector Machines on the fashion-MNIST dataset benefits from a C value near 0.1. Furthermore, SVMs are not immune to overfitting or underfitting. These results are essentially identical to those observed in the analysis of Logistic Regression. However, one noticeable difference between the two methods was the training time. The SVM implementation used in this experiment required greater than double the training time when compared to the Logistic Regression implementation, and the SVM test accuracy was only higher for very low C values (less than 10^{-6}).

Potential follow-up experiments could involve a test split weighted more towards training data in order to assess the potential accuracy increases. For example, using all 12 000 training examples from the fashion-MNIST dataset for the same binary classification problem used in this experiment. However, it should be noted that this would require significantly more training time and may not lead to a noticeable accuracy increase. Moreover, using a hard-margin SVM instead of a soft-margin SVM has potential to produce higher accuracies, though this strongly depends on the classification problem.

3. Comparison of Models Using K-Fold Cross-Validation

This section analyzes the comparison between optimally regularized forms of the two classification methods described in Sections 1 and 2. This comparison was done using the optimal C values calculated in Sections 1 and 2 along with k-fold cross-validation, a statistical resampling method used to estimate the accuracy of different models. In k-fold cross-validation, the training set is split into k groups or folds and the model is fit on the reduced training set of each fold. Likewise, the test set is split into k folds, and the accuracy of the model is evaluated on the corresponding test fold. The results of a cross-validation run are summarized using the mean (or average) of the accuracy scores for that run.

For the comparison, k-fold cross validation was performed on the Logistic Regression and Linear Support Vector Machine models with the following values of k: 5, 6, 7, 8, 9, and 10. Because the peak accuracy for both Logistic Regression and the SVM (0.96) was observed to occur at a C value of roughly 0.1, a seven-value range surrounding 0.1 was selected for the purposes of this comparison. The difference between the first C value and the seventh C value is 0.03. The average test accuracy plotted against the varied regularization parameter C is displayed in Figures 7 and 8. Each coloured line represents a different value of k. In other words, each coloured line represents cross-validation with a different number of folds.

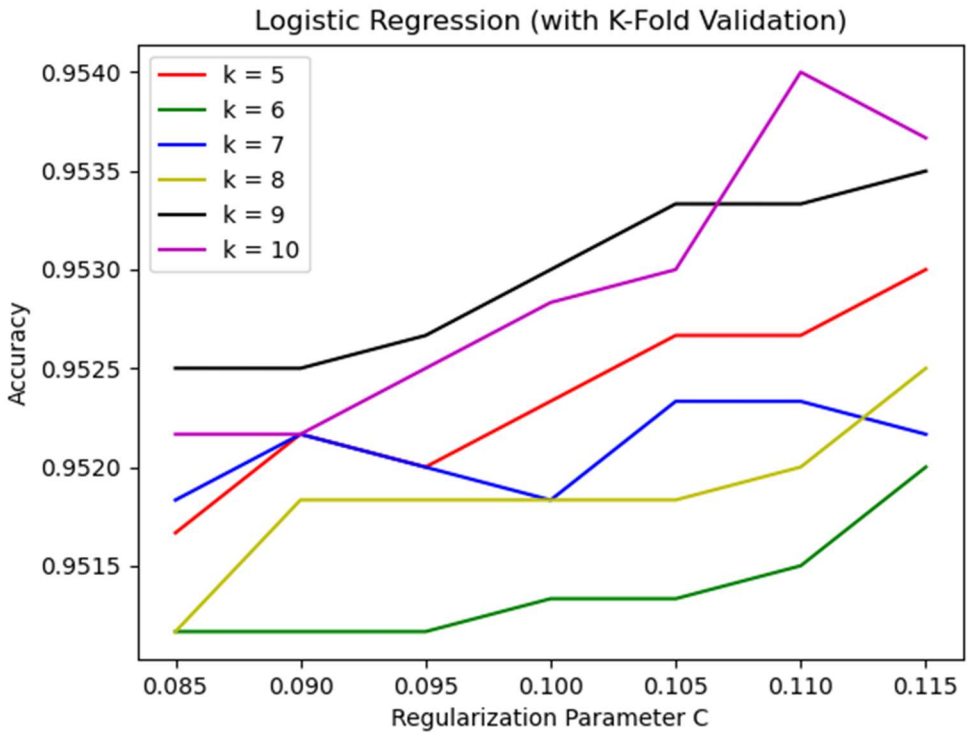

Figure 7: Logistic Regression Accuracy with K-Fold Cross-Validation

Figure 7 shows that C values of 0.110 and 0.115 produced the greatest average accuracy for each value of k. A k value of 10 and C value of 0.110 produced the greatest recorded accuracy of 0.9540. Training the Logistic Regression model on the entire training set with this C value produced an accuracy of 0.9610. The difference between these two accuracies is 0.007. Using a 99% confidence interval with n = 2000 samples (test set size), z = 2.576 (from 99% interval definition), and sample mean = 0.9542 for the observed values, the difference between the optimally regularized accuracy and the greatest accuracy observed in Figure 7 is not significant. From this result, we gain the insight that optimally regularizing the Logistic Regression method does not provide significantly higher test accuracy than simply running the basic test performed in Section 1, where we observed an accuracy of 0.96 at C values near 0.1.

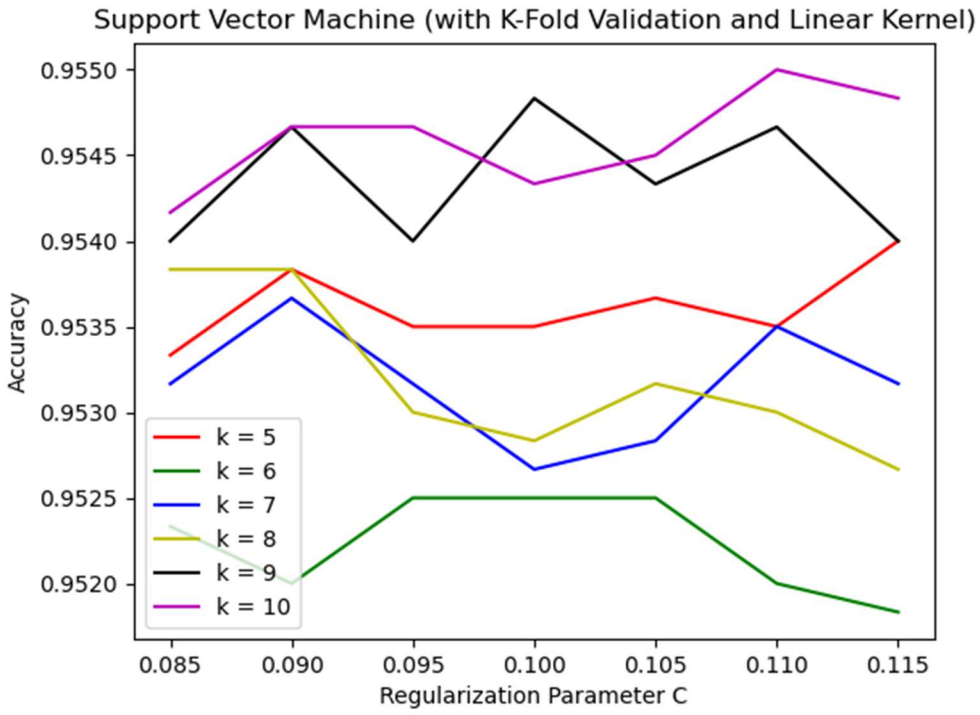

Figure 8: SVM Accuracy with K-Fold Cross-Validation

A greater variety of C values produced high average accuracies for the SVM when compared to Logistic Regression. However, a k value of 10 and C value of 0.110 again produced the greatest recorded accuracy of 0.9550. Training the SVM model on the entire training set with this C value produced an accuracy of 0.9615. The difference between these two accuracies is 0.0065. Using a 95% confidence interval with n = 2000 samples (test set size), z = 1.960 (from 95% interval definition), and sample mean = 0.9536 for the observed values, the difference between the optimally regularized accuracy and the greatest accuracy observed in Figure 8 is not significant.

Comparing Figures 7 and 8, the values for average accuracies are very similar across the board. No recorded accuracy for either method was above 0.955 or below 0.951. The similarities increase when

comparing the maximum accuracy, as each method observes this maximum with a k value of 10 and C value of 0.110. The accuracy of the optimally regularized SVM model was 0.9615, compared to 0.9610 for Linear Regression. Using a 95% confidence interval with n = 2000 samples (test set size), z = 1.960 (from 95% interval definition), and sample mean = 0.9539 for the observed values, the difference between these two optimally regularized accuracies is not significant. Overall, both methods produced consistent results with no significant difference in accuracy. However, Logistic Regression was much more efficient in terms of training time.

4. Gaussian Support Vector Machine Analysis

Now that we have analyzed and compared two linear classification methods, we will use a Support Vector Machine with the Gaussian kernel on the fashion-MNIST dataset. With the Gaussian kernel, there is an additional parameter, gamma. The gamma parameter indicates the influence of a single training example, where low gamma values indicate high influence. For this experiment, k-fold validation will be used as well. Specifically, a k value of 10 will be used for all tests due to its high accuracy in Section 3's results. Five C values near 0.1 were selected due to their high accuracy in all previous tests. Figure 9 displays the test accuracy plotted against a logarithmic scale for the gamma parameter (from 10^{-3} to 10^{0}). Each coloured line represents a different C value used to represent pairs of C and gamma.

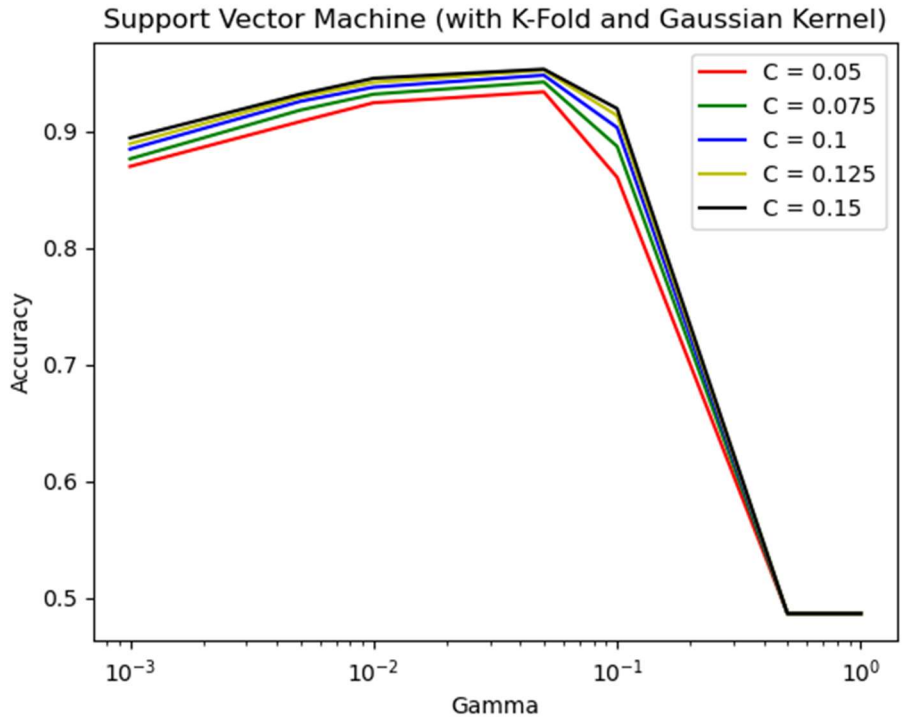

Figure 9: Gaussian SVM Accuracy with All C Values

Figures 10, 11, and 12 show the test and training accuracy plotted against a logarithmic scale for the gamma parameter in a tighter range of 10 values from 0.01 to 0.1. This tighter range was selected as it includes the accuracy peak for each C value and significant drops in accuracy as the value nears 0.1. Only three of the five C values from Figure 9 were selected due to strong similarities in the resultant graphs.

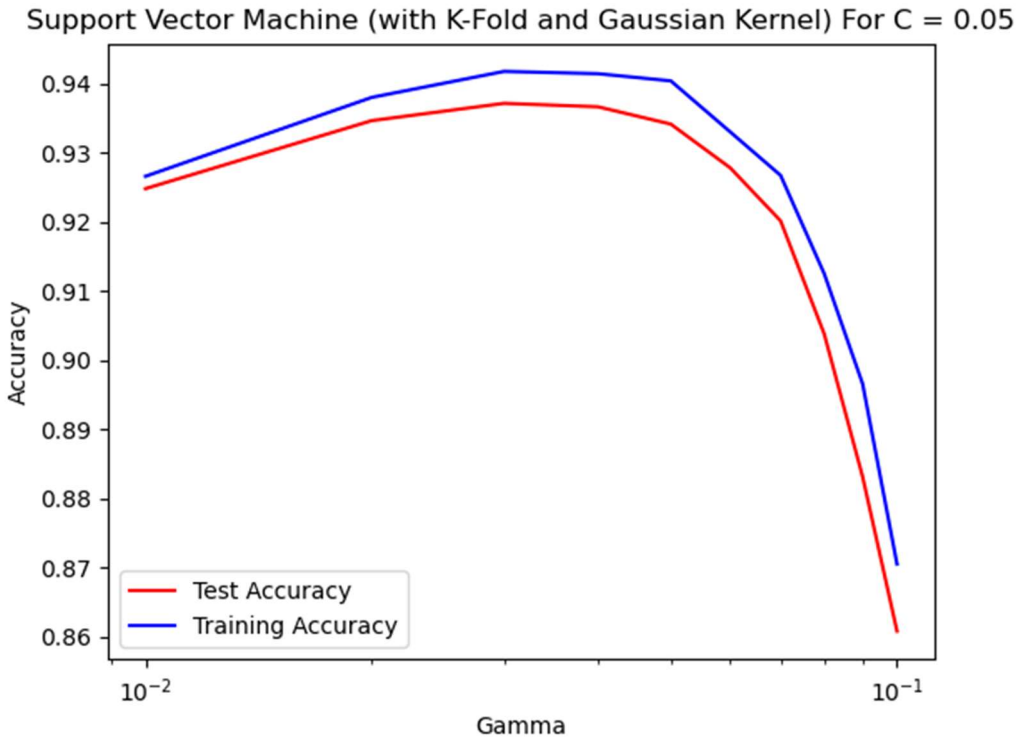

Figure 10: Gaussian SVM Accuracy with C = 0.05

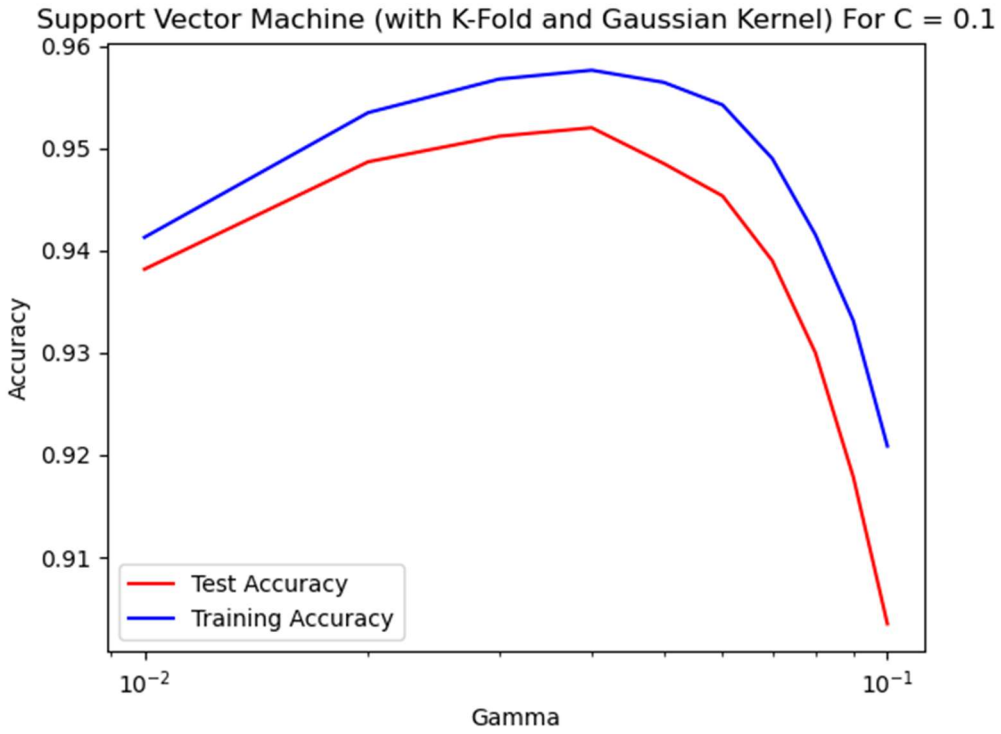

Figure 11: Gaussian SVM Accuracy with C = 0.1

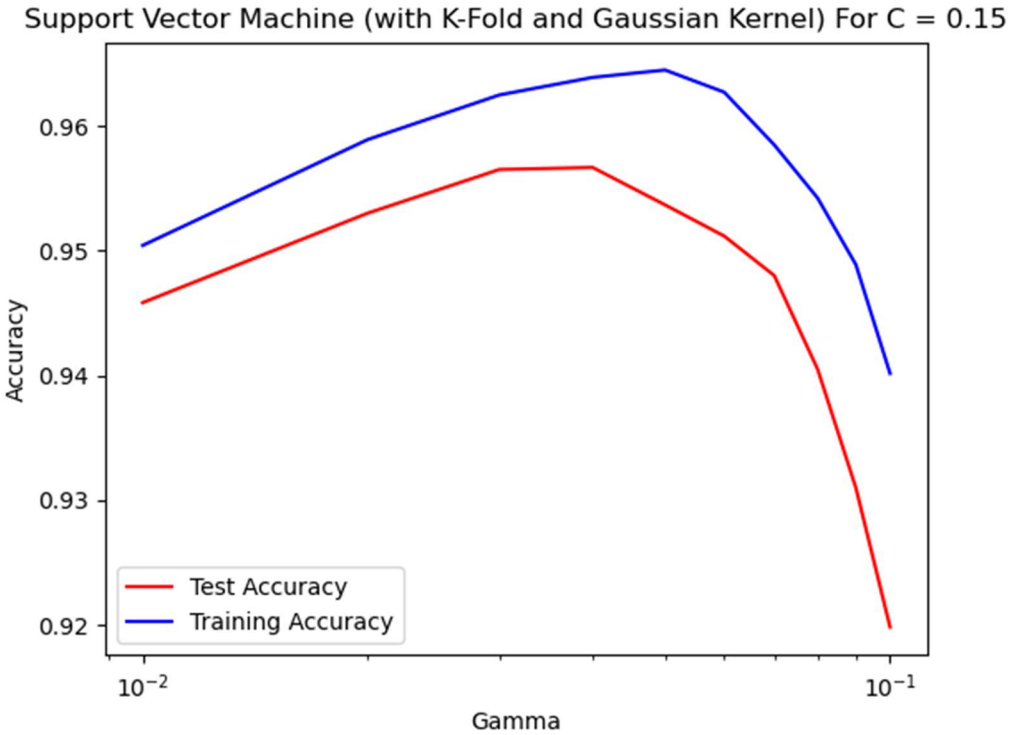

Figure 12: Gaussian SVM Accuracy with C = 0.15

Figure 10 shows a peak in test accuracy (0.935) around a gamma value of 0.05. Figure 11 shows a peak in test accuracy (0.951) around a gamma value of 0.06. Finally, Figure 12 also shows a peak in test accuracy (0.956) around a gamma value of 0.06. Therefore, from this experiment, the most accurate pair of parameters gamma and C was discovered to be gamma = 0.06 and C = 0.15. This pair produced a test accuracy of 0.956 and a training accuracy of 0.963 (other gamma values produced higher training accuracy but lower test accuracy).

Comparing the optimal test accuracy from the Gaussian SVM experiments to the optimal test accuracy from the Linear SVM experiments, where both methods use k-fold cross-validation, it can be observed that the Linear SVM achieved a higher test accuracy, 0.9615, than its Gaussian counterpart (0.956). However, the difference between these values is relatively inappreciable, and both values are similar to the test accuracies produced by the methods in all four sections present in this report. Finally, it should also be noted that the training time was significantly greater for the Gaussian SVM than the Linear SVM.